

Na cestě ke vzdělání

Moderní technologie





Obsah

Obsah	2
1 Vzdělávací program a jeho pojetí	4
1.1 Základní údaje	4
1.2 Anotace programu	6
1.3 Cíl programu	6
1.4 Klíčové kompetence a konkrétní způsob jejich rozvoje v programu	6
1.6 Hodinová dotace	7
1.7 Předpokládaný počet účastníků a upřesnění cílové skupiny	7
1.8 Metody a způsoby realizace	7
1.9 Obsah – přehled tematických bloků, podrobný přehled témat programu a jejich anotac dílčí hodinové dotace	e včetně 7
1.10 Materiální a technické zabezpečení	10
1.11 Plánované místo konání	10
1.12 Způsob vyhodnocení realizace programu v období po ukončení projektu	10
1.13 Kalkulace předpokládaných nákladů na realizaci programu po ukončení projektu	11
1.14 Odkazy, na kterých je program zveřejněn k volnému využití	12
2 Podrobně rozpracovaný obsah programu	
2.1 Roboti LEGO Mindstorms – 21 hodin	
2.1.1 Úvod LEGO Mindstorms – 3 hodiny	
2.1.2 Skládání podvozku robota – 3 hodiny	
2.1.3 Praktická ukázka programování robota – 3 hodiny	20
2.1.4 Dokončení základů programování robota – 3 hodiny	27
2.1.5 Praktická práce s dalšími senzory – 3 hodiny	29
2.1.6 Zahájení prací na vlastním technickém řešení – 3 hodiny	
2.1.7 Dokončování vlastního robota a prezentace VR pro pochopení návaznosti v rán programu – 3 hodiny	ıci plánu 32
2.2 Unity 3D VR – 12 hodin	35
2.2.1 Úvod do Unity – 3 hodiny	35
2.2.2 Částicové efekty v Unity – 3 hodiny	44
2.2.3 Osvětlení a materiály v Unity – 3 hodiny	52
2.2.4 Úprava tvaru scény v Unity – 3 hodiny	60
2.3 Vylepšování – 4 hodiny	66
2.3.1 Finalizace obou částí – 4 hodiny	66
2.4 Turnaj – 3 hodiny	66
2.4.1 Zkoušení výsledků a turnaj – 3 hodiny	





3 Metodická část	68
3.1 Roboti LEGO Mindstorms – 21 hodin	71
3.1.1 Úvod do LEGO Mindstorms – 3 hodiny	71
3.1.2 Skládání podvozku robota – 3 hodiny	74
3.1.3 Praktická ukázka programování robota	76
3.1.4 Dokončení základů programování robota	78
3.1.5 Praktická práce s dalšími senzory	80
3.1.6 Zahájení prací na vlastním technickém řešení – 3 hodiny	81
3.1.7 Dokončování vlastního robota a prezentace VR – 3 hodiny	83
3.2 Unity 3D VR – 12 hodin	84
3.2.2 Částicové efekty v Unity – 3 hodiny	87
3.2.3 Osvětlení a materiály v Unity – 3 hodiny	88
3.2.4 Úprava tvaru scény v Unity – 3 hodiny	89
3.3 Vylepšování – 4 hodiny	91
3.3.1 Finalizace obou částí – 4 hodiny	91
3.4 Turnaj – 3 hodiny	92
3.4.1 Zkoušení výsledků a turnaj – 3 hodiny	92
3.5 Zdroje a doporučená literatura	94
Příloha č. 1 - Seznam metodických příloh – pracovních listů	95
Příloha č. 2 – Závěrečná zpráva o ověření programu v praxi	96





1 Vzdělávací program a jeho pojetí

1.1 Základní údaje

Výzva	Budování kapacit pro rozvoj škol II		
Název a registrační. číslo projektu	Ve škole po škole		
projektu	CZ.02.3.68/0.0/0.0/16_032/0008190		
Název programu	Moderní technologie		
Název vzdělávací instituce	Středisko služeb školám a zařízení pro další vzdělávání pedagogických pracovníků Brno, příspěvková organizace		
	Od 1. 7. 2021: Vzdělávací institut pro Moravu, zařízení pro další vzdělávání pedagogických pracovníků a středisko služeb školám, příspěvková organizace		
Adresa vzdělávací instituce a webová stránka	Hybešova 15, Brno, 602 00		
	http://www.sssbrno.cz		
Kontaktní osoba	Mgr. Hana Knapová		
Datum vzniku finální verze programu	7_2021		
Číslo povinně volitelné aktivity výzvy	02_16_32		
	Aktivita 4 – Propojování formálního a neformálního vzdělávání – rozvoj klíčových kompetencí		
Forma programu	Týmová spolupráce pracovníků formálního a neformálního vzdělávání.		
Cílová skupina	Žáci středoškolských oborů zakončených maturitní zkouškou		
Délka programu	40 hodin		
Zaměření programu (tematická oblast, obor apod.)	Rozvoj talentu dětí a žáků v rámci formálního, zájmového a neformálního vzdělávání, podpora dlouhodobé a systematické práce s talentovanými dětmi a mládeží.		
	Využívání kreativního a inovativního potenciálu dětí a mládeže.		
	Spolupráce škol a školských zařízení s organizacemi a institucemi, které se zabývají pomocí dětem a mládeži s omezenými příležitostmi		





Tvůrci programu	Ing. Miroslav Novotný, Ing. Tomáš Brestič
Odborný garant programu	Ing. Miroslav Novotný, Ing. Ludmila Brestičová
Odborní posuzovatelé	Gymnázium T. G. Masaryka v Zastávce
Specifický program pro žáky se SVP (ano x ne)	ne





1.2 Anotace programu

Program umožňuje středoškolským studentům díky propojení oblasti formálního a neformálního vzdělávání spojit znalosti získané ve fyzice, matematice a informatice s praktickou aplikací při sestavování a programování malých řízených robotů. Program probíhá ve škole za přítomnosti učitelů znalých základů programování a schopných instalace prostředí virtuální reality ze spolupracujících středisek volného času.

Program je určený pro desetičlenné skupiny nadaných studentů středních škol, je však modifikovatelný i pro ostatní žáky.

1.3 Cíl programu

Cílem je umožnit talentovaným a mimořádně nadaným žákům přístup k moderní technice a umožnit jim pracovat pod vedením odborníků na zadaných úkolech. Úkoly, které budou řešit, mohou být dále zpracovány například jako práce středoškolské odborné činnosti, nebo mohou být základem technické soutěže. Účastníci v rámci programu rozvíjejí svoji schopnost řešit komplexnější úlohy vyžadující jak manuální dovednosti, tak aplikaci teoretických znalostí. Dále přispívá k posílení schopnosti týmové práce a komunikace ve skupině a ukazuje žákům, že znalosti nabyté ve fyzice, matematice či informatice mohou mít zajímavá využití v praxi. Program pokrývá značný rozsah obtížnosti jeho cílů, záleží zde na domluvě se žáky a znalostech jak studentů, tak lektorů, aby se nastavila cílová laťka programu vhodně.

1.4 Klíčové kompetence a konkrétní způsob jejich rozvoje v programu

Komunikace v mateřském jazyce

Program není přímo zaměřen na oblast komunikace v mateřském jazyce, ale obohacuje účastníky o nové pojmy z oblasti technických věd.

Komunikace v cizích jazycích

Žáci budou obvykle pracovat s návody a manuály v anglickém jazyce. Budou vedeni k procvičování anglické odborné terminologie.

Matematická schopnost a základní schopnosti v oblasti vědy a technologií

Program je zaměřen na zvyšování této kompetence. Řešením příkladů a navrhováním dalších úloh žáci výrazně posílí své kompetence v oblasti vědy a technologií.

Schopnost práce s digitálními technologiemi

Program je zaměřen na práci s digitálními technologiemi. Výrazně zvyšuje kompetence žáka v této oblasti.

Schopnost učit se

Účastníci se naučí pracovat s různými typy výukových materiálů, chápat smysl zadaných pracovních materiálů a vyhledávat informace podle zadání.





Sociální a občanské schopnosti

Program podporuje spolupráci zapojených žáků. Umožňuje zapojení žáků s omezenými příležitostmi. Umožňuje navázání nových kontaktů.

Kulturní povědomí a vyjádření

Při realizaci tohoto programu budou účastníci řešit i úkoly související s tvorbou digitálního umění.

Smysl pro iniciativu a podnikavost

Výrazná část programu obsahuje samostatnou nebo skupinovou práci a je oceněn inovativní a originální přístupu k problému.

1.5 Forma

Aplikovaná forma tohoto programu je prezenční, z organizačních forem převládá skupinová s aspektem individuálním. Žáci spolupracují ideálně ve skupinách po dvou, výjimečně po třech nebo samostatně. Dále dochází ke spolupráci pracovníků formálního a neformálního vzdělávání. Z oblasti neformálního vzdělávání budou též zapojena střediska volného času.

1.6 Hodinová dotace

40 hodin, rozdělených do více hodinových bloků.

1.7 Předpokládaný počet účastníků a upřesnění cílové skupiny

10 nadaných žáků středoškolských oborů zakončených maturitní zkouškou.

1.8 Metody a způsoby realizace

Mezi konkrétní formy realizace patří workshopy, samostatná řešení zadaných praktických úkolů nebo diskuze, velká část náplně programu je pak tvořena skupinovou nebo samostatnou prací vymezenou společným cílem.

1.9 Obsah – přehled tematických bloků, podrobný přehled témat programu a jejich anotace včetně dílčí hodinové dotace

Tematický blok č. 1: Roboti Lego Mindstorms (21 hodin)

Úvod do robotiky, stavba robota, programování.

Téma č. 1 (Úvod LEGO Mindstorms) – počet hodin 3

- Úvod do robotiky.
- Základního přehled o stavebnici Lego Mindstorms.
- Představení lektorů.
- Uskupení do týmů.
- Směrování programu.





Téma č. 2 (Skládání podvozku robota) – počet hodin 3

- Skládání podvozku robota.
- Způsoby propojení částí a dílů.
- Efektivní použití dílů.
- Vliv způsobů spojení na konstrukční vlastnosti.
- Nutné díly k fungování celku.

Téma č. 3 (Praktická ukázka programování robota) – počet hodin 3

- Možnosti využití řídicího dílu robota Intelligent Brick.
- Základy programování robota, grafický režim.
- Nahrání programu do robota, spouštění.
- Propojení řídicího dílu s programovacím prostředím.
- Efektivní práce s programovými bloky.
- Způsoby řízení robota.

Téma č. 4 (Dokončení základů programování robota) – počet hodin 3

- Pravidla připojení senzorů.
- Vstupní hodnoty senzorů a jejich rozsahy.
- Reakce robota na odlišné vstupy senzorů.
- Zaměření na vzdálenostní, dotykové a barevné čidlo.
- Nezávislé řízení motorů robota.

Téma č. 5 (Praktická práce s dalšími senzory) – počet hodin 3

- Rozšíření stávající funkcionality obohacující roboty o další informace o jejich okolí.
- Kombinace vstupů senzorů.
- Nastavení rychlosti robota.
- Případné strukturální změny z důvodu zvýšení efektivity fungování senzorů.

Téma č. 6 (Zahájení prací na vlastním technickém řešení) – počet hodin 3

- Detailní analýza potřebných vlastností pro efektivní účast v soutěži.
- Samostatná práce na strukturální, případně programové části projektu.
- Vyhodnocení priorit v bodovém hodnocení soutěže.

Téma č. 7 (Dokončování vlastního robota a prezentace VR) – počet hodin 3

- Dokončování vlastního robota a prezentace VR pro pochopení návaznosti v rámci plánu programu.
- Představení hardwarových prvků virtuální reality.
- Využití softwaru s návazností na virtuální realitu.
- Dokončení robota v průběhu testů s hardwarem virtuální reality.
- Ukázkové programy virtuální reality.





Tematický blok (Unity 3D VR) č. 2 – počet hodin 12

Úvod do programu Unity 3D, jednotlivé prvky.

Téma č. 1 (Unity 3D VR) – počet hodin 3

- Úvod do Unity.
- Předvedení nejdůležitějších prvků Unity editoru.
- Vlastnosti a použití nejdůležitějších objektů a komponent v Unity.
- Vytvoření testovací/výukové scény.
- Vytvoření testovacího/výukového skriptu.

Téma č. 2 (Částicové efekty v Unity) – počet hodin 3

- Práce s grafickými objekty v Unity.
- Využití Unity editoru ke grafické úpravě prvků ve scéně.
- Vytvoření efektů ve scéně.
- Představení možností kamery.
- Grafická úprava virtuální scény.

Téma č. 3 (Osvětlení a materiály v Unity) – počet hodin 3

- Nastavení ambientního osvětlení.
- Základní bodové a směrové zdroje světla.
- Možnost promítnutí okolí na objekt a úpravy tohoto obrazu.
- Další grafické vlastnosti osvětlení scény.
- Interakce osvětlení s ostatními prvky a vlastnostmi objektů ve scéně.

Téma č. 4 (Úprava tvaru scény v Unity) – počet hodin 3

- Import textury a tvorba materiálu.
- Určení vizuálních vlastností vykreslovaných objektů.
- Nastavení některých vlastností textur a materiálů.
- Aplikace materiálů na objekty ve scéně.

Tematický blok (Vylepšování řešení) č. 3 – počet hodin 4

Dokončovací práce před závěrečnou soutěží.

Téma č. 1 (Finalizace obou částí) – počet hodin 4

- Úpravy a dokončení robota i VR aplikace, příprava taktiky pro závěrečný turnaj.
- Práce v závislostí s vlastním stavem projektu dané skupiny.
- Testování funkčnosti jednotlivých části projektu.

Tematický blok (Turnaj) č. 4 – počet hodin 3

Doladění řešení a turnaj. Vyhodnocení výsledků.





Téma č. 1 (Zkoušení výsledků a turnaj) – počet hodin 3

- Příprava na provedení turnaje.
- Závěrečné ladění skupinových řešení.
- Závěrečný turnaj s vyhodnocením.

1.10 Materiální a technické zabezpečení

Tablet: 5 ks × 11 000 Kč

PC pro virtuální realitu: 1 ks × 35 000 Kč

Zařízení pro virtuální realitu: 1 ks × 19 000 Kč

stavebnice LEGO Mindstorms + příslušenství: 10 × 17 000 Kč

3D tiskárna: 1 × 30 000 Kč

Materiál pro 3D tisk: 2 × 5 000 Kč

Spotřební materiál: 2 × 10 žáků × 300 Kč

Ceny na soutěž: 1 000 Kč

1.11 Plánované místo konání

Prostory Gymnázia T. G. Masaryka v Zastávce.

1.12 Způsob vyhodnocení realizace programu v období po ukončení projektu

Po ukončení programu se budou zařízení dále využívat v ostatních programech.





1.13 Kalkulace předpokládaných nákladů na realizaci programu po ukončení projektu

Počet realizátorů: 3

Položka		Předpokládané náklady
Personální náklady – 2 × odborný učitel, 1 × asistent učitele		52 000 Kč
z toho	Hodinová odměna pro 1 odborného učitele (výuka + příprava, DPP bez odvodů)	500 Kč
	Hodinová odměna pro 1 asistenta učitele (výuka + příprava, DPP bez odvodů)	300 Kč
Náklady na zajištění prostor		20 000 Kč
Ubytování, stravování a doprava účastníků		0 Kč
z toho	Doprava účastníků	0 Kč
	Stravování a ubytování účastníků	0 Kč
Náklady materiální a technické vybavení		305 000 Kč
z toho	Nákup vybavení (tablet 5 ×, PC pro virtuální realitu 1 ×, zařízení VR 1 ×, stavebnice LEGO MIndstorms 10 ×, 3D tiskárna 1 ×)	300 000 Kč
	Odměny žákům	5 000 Kč
Režijní náklady		23 000 Kč
z toho	Stravné a doprava organizátorů	0 Kč
	Ubytování organizátorů	0 Kč
	Poštovné, telefony	500 Kč
	Doprava a pronájem techniky	0 Kč
	Propagace	500 Kč
	Ostatní náklady	10 000 Kč
	Odměna organizátorům	12 000 Kč
Náklady celkem		400 000 Kč
Poplatek za 1 účastníka		40 000 Kč





1.14 Odkazy, na kterých je program zveřejněn k volnému využití

Pokud není uvedeno jinak, jsou všechny použité obrázky, fotografie a videa autorským dílem organizace Vzdělávací institut pro Moravu, zařízení pro další vzdělávání pedagogických pracovníků a středisko služeb školám, příspěvková organizace. Program je zveřejněn k volnému využití na stránkách:

https://www.sssbrno.cz/projekty/projekt-ve-skole-po-skole



.

Program Moderní technologie podléhá licenci Creative Commons Uveďte původ-Zachovejte licenci 4.0 Mezinárodní. Pro zobrazení licenčních podmínek navštivte





2 Podrobně rozpracovaný obsah programu

2.1 Roboti LEGO Mindstorms – 21 hodin

2.1.1 Úvod LEGO Mindstorms – 3 hodiny

1. hodina – Co vlastně budeme v kroužku dělat? O čem je robotika? Podrobně rozpracovaný obsah

- Dobrý den! Vítáme vás v tomto programu. Jak jistě víte, hlavním tématem našich setkávání budou roboti a robotika obecně. Ale co tyto pojmy vlastně znamenají? Zkusíme si je na začátek společně ujasnit.
 - a. a. Co si představíte pod pojmem robot?
 - b. b. Kde všude se můžeme s roboty setkat a jaký užitek nám mohou přinést?
 - c. c. Jak vlastně označení robot vzniklo? Je to výstižné pojmenování? Zvládli byste vymyslet lepší?
 - d. d. Jsou roboti součástí nějakých významných literárních nebo filmových děl?
 - e. e. Co to je robotika?
- 2. Během příštích hodin si postupně sestavíte vlastního robota a toho následně spojíte s virtuální realitou. Aby mohl váš robot obstát v soutěži, kterou nakonec uspořádáme, bude potřeba splnit některé základní funkční požadavky. Nebojte se, možná vše vypadá na první pohled složitě, ale vedoucí tohoto programu vám se vším pomohou a postupně vás naučí, jak sestavit robota, jak jej ovládat, i to, jak ho propojit do virtuální reality. Prostě máme zvládnuté a naplánované všechny kroky, abyste všichni zdárně prošli celým programem. Uvidíte, že i váš robot se bude schopen vypořádat se všemi překážkami, které ho potkají.
- 3. Ve druhé polovině programu vytvoříte scénu aplikace ve virtuální realitě, která se pak krátce stane domovem vašeho, v té době již dokončeného robota. V programu kombinujeme dohromady několik různých technologii, přičemž některé z nich pochází z dosti odlišného technologického základu. Ale aby toho na vás najednou nebylo příliš, budeme vám tyto nástroje a technologie uvádět postupně.
- 4. Ještě, než se dáme do práce možná už vy sami nějaké osobní zkušenosti s robotikou a virtuální realitou máte. Povídejte! Ukazujte, pochlubte se! Určitě můžete své vlastní výrobky přinést příště ukázat ostatním.
 - a. Setkali jste se s roboty v nějakém jiném kroužku?
 - b. Máte podobné stavebnice a sety sami doma?
- 5. Pojďme se nyní seznámit s pravidly BOZP, s řádem učebny a způsoby nakládání s pomůckami. Je to opravdu nutné. V tomto programu totiž budeme všichni pracovat s technikou. Není levná, je opravdu sofistikovaná a náchylná na pečlivé zacházení. Co se týče lego robotů, upozornění, která se vás skutečně týkají, je minimální počet, jsou mířeny i na podstatně mladší věkovou kategorii. Obvykle jsou pak ve formě: prosím nepolykejte drobné části, neházejte elektroniku do vody a podobně.
 - a. Nenakládejte s částmi stavebnice mimo jejich zamýšlené využití.
 - b. Chovejte se k částem stavebnice opatrně, obzvláště k elektronickým součástkám.
 - c. Nepřesouvejte součásti mezi krabicemi bez povolení lektorů.





- d. Nenechávejte, obzvláště malé části ležet volně na stole nebo na zemi, rády se ztrácejí a mohou se poškodit, pokud na ně někdo šlápne.
- e. Za žádných okolností s díly neházejte.

Reflexe

Jak vás téma robotů zaujalo? Inspirovalo vás dnešní setkání? Chtěli byste se zabývat moderními technologiemi i v budoucnu?

2. hodina – Pořádek ve stavebnici je základ

Podrobně rozpracovaný obsah

Jdeme na to. Dnes poprvé otevřeme stavebnici Lego Mindstorms. Podíváte se, co je vlastně uvnitř, a hlavně se naučíte veškeré součástky (a vlastně veškerý pracovní materiál) správně zorganizovat. Určitě zbyde prostor i na první malé stavění – postavíme podvozek vašeho robota.

Důležité upozornění: Nejlepší způsob, jak se naučíte pracovat s jednotlivými díly, je vaše vlastní experimentování, nenechte se lektory a pracovními listy odradit od vaší vlastní vize! To, že něco je v návodu, ještě neznamená, že to sami nemůžete vylepšit.

- 1. Jak jistě víte, dvě hlavy ví více než jedna. A ve skupině se robot staví lépe. I proto budete v tomto programu spolupracovat s vašimi spolužáky a budete mít prostor projevit vaše zájmy a nadání. Skupiny utvořte ideálně dvojčlenné.
- 2. Z tohoto projektu si sice robota neodnesete, ale zato získáte nejen nové zážitky, ale i zkušenosti a znalosti. Když na něco přijdete nebo něco vymyslíte, nezapomeňte si o tom popovídat se zbytkem skupiny a podělit se tak o své nápady!
- 3. V programu budete používat větší množství vybavení. Každá stavebnice zároveň obsahuje velké množství malých částí, v mnoha totožných kopiích, rozdělených do několika krabic. V průběhu práce proto může nedopatřením docházet k záměně některých částí nebo neúmyslnému přesunu některých dílů mezi krabicemi. Bude lepší, pokud se toho pokusíte vyvarovat. U malých součástek, které existují ve větším množství kopií, tyto přesuny obvykle nezpůsobují zásadní problémy, mohlo by se ale stát, že dojde k záměně některých z elektronických částí. Abychom tomu zamezili, na začátku programu, než se pustíš ve skupině do prvních pokusů se stavebnicemi, důsledně označ všechny důležité prvky. Stavebnice s tímto naštěstí počítá a disponuje několika nálepkami, na které je možné vhodnou fixou napsat čísla, která jednoznačně určí, ke které stavebnici daný důležitý díl patří. Stejné číslo je pak napsáno na podložku pro díly a samotné krabice stavebnic. Napište na všechny krabice číslo zepředu i z vrchu, aby se daly později snáze najít. Pokud nejste první skupina v tomto programu, stačí pouze zkontrolovat, jestli jsou popisky dobře čitelné na každé součástce.
- 4. Těch součástek je opravdu hodně, že?! Kdo se v tom má vyznat! Naštěstí máte k dispozici podrobnou uživatelskou příručku od autorů stavebnice obsahující podrobné informace o všech jejích částech:
- . Jestli se ti některý díl nedaří identifikovat, využij manuál.
 - 5. Jakmile budete mít ve stavebnici nějaký řád, můžete se rovnou vrhnout do tvorby svého robota. Nejlepší je začít hlavní kostrou a podvozkem, na které budete ostatní díly připevňovat.





Ke stavebnici jsou na internetu dostupné i návody na stavbu různých robotů, silně vám však doporučuji pustit se do stavby po vlastní ose, získáte tak nejvíce zkušeností.

- 6. Při stavbě konstrukce robota mějte v povědomí, co bude potřeba nést, určitě budete potřebovat řídicí blok a dva motory, k tomu pravděpodobně nějaká kola nebo pásy. Pamatujte, že řídicí blok musí být kabelem napojen na motory, je potřeba jej nabíjet a má na horní straně ovládací tlačítka, je velmi vhodné jej umístit tak, abyste na něj později dosáhli.
- 7. Dalším dílem, který budete později přimontovávat na robota, bude VR tracker, který slouží k označení polohy vašeho robota, aby se mohl hýbat i ve virtuální scéně (detaily se dozvíte později), tento díl je však pouze jeden pro celý kurz, budete si jej tedy půjčovat od lektorů a jeho umístění nemusíte řešit.

Reflexe

Pracujete rádi ve skupinách nebo preferujete individuální činnost? Jaká byla vaše dnešní spolupráce? Nakolik jste využili manuál k popisu dílů stavebnice? Je manuál zpracovaný přehledně? Byl pro vás pochopitelný? Podařilo se vám navrhnout vlastní funkční řešení?

3. hodina – Rozhýbeme robota

Podrobně rozpracovaný obsah

Minulé hodiny vám mohly připadat trochu suché, nyní se program doslova rozhýbe. Naučíme totiž vašeho prvního robota pár základních pohybů. Co kdyby uměl reagovat na povely vpřed, vzad, otoč se? Možná vám to připadá složité, ale opak je pravdou. Zvládneme to za pár minut s pomocí pár bloků kódu a vašeho mobilu či tabletu. Pojďme na to.

- V minulého hodině jste začali skládat podvozek vašeho robota, nyní mu přidáme pohyb. Jediné, co k řízení potřebujete, je dálkového ovládání z mobilu nebo tabletu (aplikace Lego Mindstorms Commander). Aplikace je dostupná v distribucích aplikací jako je google play pro Android nebo app store pro iOS. Najdete ji pod jménem LEGO® MINDSTORMS® Commander.
- Tak co, poslouchá vás robot? Funguje ovládání mobilem? Tak to nyní posuneme trochu dál. Vytvořte jednoduchý program pro robota (v aplikaci Lego Mindstorms Programmer). Opět vám dám na výběr lehčí a pokročilejší variantu:
 - a. Aplikace pro řízení nám dávají možnost samostatného vyzkoušení již postavených robotů. V podstatě spouštíme předem sestavený program.
 - b. A nyní se posuneme ještě dál krom možnosti spustit program nahraný na robotovi, lze také jednoduše sestavit na tabletu (nebo mobilním telefonu) řídicí schéma pro robota. Toto schéma v sobě má základní ovládací prvky jako vpřed a vzad nebo zatáčení a dovolí vám řídit robota přímo bez potřeby spouštět program. Pojďte si to vyzkoušet. Pokud si nejste jistí, vedoucí programu vám určitě poradí.
- 3. Pokud jste z minulé hodiny ještě nestihli postavit celý podvozek, nezoufejte, pokračujte dále a na program přejděte až budete připraveni.





Reflexe

Složili jste úspěšně všechny části robota? Co vás nejvíce překvapilo? Zasekli jste se na nějaké části stavby? Ohodnoťte na škále 1–5, nakolik bylo pro vás obtížné přidat první pohyb.

2.1.2 Skládání podvozku robota – 3 hodiny

1. hodina – Vylepšujeme podvozek

Podrobně rozpracovaný obsah

Z předchozích lekcí máte hotový základní podvozek. Určitě jde vylepšit, co říkáte? Takže v této lekci budeme všichni experimentovat s rozložením dílů a možná navrhnete vlastní neotřelé řešení.

- 1. V této hodině programu navážete na předchozí představení stavebnice, tentokrát však zajděte do většího detailu, zejména se zaměřte na možné zapojení jednotlivých částí do sebe.
 - a. Jak ovlivní pohyb robota, pokud bude kostra robota příliš volná nebo bude jeho pohon nesymetrický?
 - b. Může se stát, že robot nepojede správně rovně nebo se bude za jízdy třást. Jaké řešení tohoto problému se vám nejvíce osvědčilo?
 - c. Pozor, je velmi obtížné postavit podvozek s koly jako auto, tak aby robot správně a hladce zatáčel. Pásy nebo dvě kola a kovová kulička představují mnohem snazší cestu. Každé technické řešení však s sebou nese i své nevýhody.

Poznámka – Nejlepší cestou je pak experimentování s různým umístěním a připojením hnací nápravy. Robot by ideálně měl mít pohon symetrický, nic by nemělo drhnout a nemělo by hrozit, že se robot za jízdy rozpadne.

Určitě věnujte čas samostatné práci a vlastnímu zkoušení, jak jednotlivé díly stavebnice společně fungují. Lektoři vám poslouží jako poradci a pomůžou vám řešit vzniklé problémy. Tuto hodinu zakončete složením podvozku robota, můžete použít i instrukce pro skládání podvozku některého ze základních robotů. Lepší je však postavit vlastní řešení.

Reflexe

Dnes jsme vybočili z běžných postupů. Baví vás experimentování? Co nového jste se naučili?





Pracovní List 1: Postup stavby robota

Vaším cílem bude navrhnout a postavit robota, který ponese *Vive Tracker* a bude schopen vyhýbat se fyzickým překážkám a zůstávat v oblasti vymezené barevnou páskou na podložce. Pokud jste se dosud nesetkali se stavebnicí *LEGO Technic*, nebojte se strávit nějaký čas zkoumáním možností spojování dílků dohromady, než se pustíte do vlastní konstrukce robota. Vyzkoušejte si secvakávání dílků pomocí kulatých spojek i nasouvání na tyče s křížovým průřezem, prozkoumejte různé tvary kostiček a možnosti jejich kombinace.

Pokuste se navrhnout vlastní konstrukční řešení, ale pokud cítíte, že potřebujete inspiraci, můžete k tomu využít některý z oficiálních návodů, například:

Nebojte se v průběhu práce svoje řešení částečně rozebírat a upravovat.

Pokud nemáte mnoho zkušeností se stavebnicí tohoto typu, držte se spíše rovnoběžných a kolmých konstrukcí, využití jiných úhlů je obvykle náročnější.

Myslete na to, že některé prvky *"programovatelné kostky"* (řídicí jednotka robota, viz obrázek) by měly zůstat přístupné (nezakryté jinými částmi robota) – je to především zdířka napájecího konektoru, konektor USB a vstupní a výstupní porty. Může být výhodné umožnit výměnu celé programovatelné kostky bez nutnosti rozebrání rozsáhlejší části robota (např. pak lze vyměnit akumulátorový blok nebo celou jednotku v případě vybití při testování nebo při poruše), ale není to nezbytně nutné.



K pohybu robota je možné využít dva *velké motory* a jeden *střední* (viz obrázek). Velké motory lze teoreticky použít i jako vstupní senzor, například pro snímání rozsahu provedené rotace.







Robot by měl nést senzory, které mu pomohou snímat prostředí okolo něj (typicky senzor barvy, ultrazvukový senzor vzdálenosti, tlačítko), proto je do konstrukce vhodně zakomponujte – např. senzor barvy musí být pro zajištění správné funkce umístěn poměrně nízko nad podložkou.

Dále bude nutné, aby bylo možné na robota připevnit *Vive Tracker* (viz obrázek) tak, aby nebyl nijak výrazně zakrytý a aby bylo možné jej volně sundávat a připevňovat.



Nezoufejte, pokud robot nebude fungovat dokonale ve všech situacích (např. nezaznamená všechny překážky, do nichž může najet). I opravdu kvalitní řešení pravděpodobně nebude stoprocentně úspěšné.

Reflexe

Dělá váš robot to, co má? Splňuje vaše stavba vaše očekávání? Nakolik zvládl váš robot požadované úkoly?





18

2. hodina – Pokračujeme ve stavbě robota

Podrobně rozpracovaný obsah

Tak co, jak jste daleko? V minulé hodině jsme vylepšovali podvozky. Vznikla mnohá velmi zajímavá řešení. Každý z vás ten problém pojal a vyřešil trochu jinak. Jak jste daleko? S čím potřebujete pomoct? Než se pustíme do dalších kroků ve stavbě, je potřeba, abychom se sjednotili.

- Nyní byste již měli mít z větší části složený podvozek robota. Pokud jste už v minulého hodině robota opakovaně přestavovali nebo jste se někde zdrželi, nedělejte si s tím starosti, je zde dost prostoru pro dokončení nebo vylepšování. Počáteční požadavky na stavbu nejsou vysoké a jde spíše o získané zkušenosti, nebojte se robota celého přestavět, pokud budete cítit, že je to vhodné.
- 2. S funkčním základním podvozkem společně se zapojením řídicího bloku vyzkoušejte, jestli je robot schopen jet rovně dopředu a zatočit do požadovaného směru. K tomuto testování použijte řídicí schéma vytvořené na mobilním zařízení. O tom jsme se bavili už v 2. hodině.
- 3. Později v programu budete na robota připevňovat zařízení umožňující jeho zahrnutí do světa ve virtuální realitě. Mějte tedy na paměti, že váš robot musí mít na sobě dost široký prostor, na který se tento, takzvaný VR tracker, dá připevnit.

Reflexe

Skládá se vám se stavebnicí dobře? Povedlo se vám na robotovi vytvořit dostatečně široký prostor pro připojení VR trackeru?

3. hodina – Programovací prostředí – první seznámení

Podrobně rozpracovaný obsah

Tak, jdeme konečně programovat. Naštěstí k tomu nepotřebujeme žádný vyšší programovací jazyk a vše zvládneme pomocí bloků programů. Pokud znáte code.org nebo jste někdy programovali ve Scratchi, bude vám ten systém práce blízký. Dnes se poprvé podíváme do programovacího prostředí a podrobně jej prozkoumáme.

Co se naučíte?

- 1. Jak vytvořit, ukládat a načítat projekt do "programovatelné kostky" (řídicí jednotky robota).
- 2. Manipulaci s rozhraním programovacího prostředí, základní skládání funkčních bloků, jejich propojování.
- 3. Vyzkoušíte si program umožňující pomocí grafického programování vytvořit základní řídicí program pro jednoduchého robota.

Poznámka – Grafické programovací prostředí je dostupné ve dvou variantách – mobilní verze a PC verze. Zatímco PC verze nabízí mocnější nástroje a lepší ovládání, mobilní usnadňuje přístup k robotovi skrze bezdrátové napojení. Na začátek doporučujeme začít na mobilních platformách s případným pozdějším přechodem na vybavenější počítačové rozhraní.





Reflexe

Je váš programovací blok na robotovi dostatečně přístupný? Fungovalo vám propojení vašeho robota s mobilem dobře?

2.1.3 Praktická ukázka programování robota – 3 hodiny

1. hodina – Experimentujeme s programováním

Podrobně rozpracovaný obsah

Ve vývoji našeho robota jsme opravdu pokročili. Vyzkoušíme nové techniky, a právě proto vás čeká další pracovní list s konkrétními úkoly. Víte, co je to cyklus (loop)? I takový úkol na vás v pracovním listu čeká. Postupujte podle něj.

V pokročilých úkolech dokonce vyzkoušíte senzory. To je jedna z funkcí, kterou jsme dosud neřešili. Co kdyby váš robot dokázal sám reagovat na překážku?

Vyvíjejte dále vašeho robota, ve stavbě i v programu, a tyto úpravy opakovaně testujte nahráváním a spouštěním programu.

- 1. V této hodině máte prostor k dalšímu experimentování. Tentokrát se zaměřte na programování robota, prostředí vám umožňuje z bloků vytvořit jednoduché chování. Co které bloky dělají, můžete vyzkoušet, ale jejich popisky opět najdete v příručce.
- 2. Vyvíjejte dále vašeho robota, ve stavbě i v programu, a tyto úpravy opakovaně testujte nahráváním a spouštěním programu. V momentě, kdy budou vytvořeny základní funkce, můžete přestoupit k vylepšování a ladění programu a k jeho případnému rozšiřování. Tento proces bude dále probíhat i v průběhu dalších hodin tohoto bloku programu.

Reflexe

Je pro vás prostředí přehledné? Nakolik dělá váš robot to, co po něm vyžadujete?





Pracovní List 2: Úvod k programovatelné kostce robota

Úkol 1: Příprava a ověření funkčnosti základních komponent

V krabici s dílky najděte "programovatelnou kostku" (řídicí jednotku robota), oba velké motory, senzor barvy (viz obrázek) a tři kabely.



Prohlédněte si *programovatelnou kostku*. V horní části se nacházejí *výstupní porty* označené písmeny (A až D), sem se zapojují motory umožňující pohyb robota. Ve spodní části jsou *vstupní porty* označené čísly (1 až 4), do nichž je možné připojit senzory, prostřednictvím kterých robot vnímá své okolí. Pomocí kabelů zapojte motory do výstupních portů B a C a senzor barvy do vstupního portu 3.

Zapněte programovatelnou kostku a prozkoumejte menu. Pokuste se najít položku pro přímé ovládání motorů a vyzkoušejte, zda se otáčejí. Poté najděte položku pro čtení hodnot senzoru a otestujte, jaké hodnoty získáte na různě barevných či lesklých površích a v jaké vzdálenosti funguje senzor nejlépe.





Úkol 2: Seznámení s programovacím prostředím

Otevřete program *LEGO MINDSTORMS Education EV3*. Pro vytvoření nového programu klikněte na tlačítko se symbolem "+" v levé horní části obrazovky (na obrázku označeno šipkou 1). Zobrazí se prázdná plocha s jediným *instrukčním blokem*, který označuje začátek programu pro robota (viz šipka 2). V dolní části obrazovky se nacházejí jednotlivé instrukční bloky, z nichž lze poskládat program pro robota (viz šipka 3).



Instrukční bloky jsou rozděleny do barevně odlišených kategorií. Zleva doprava jsou to tyto kategorie:

- akční bloky (zelená)
- bloky pro řízení programu (oranžová)
- senzorové bloky (žlutá)
- bloky pro operace s daty (červená)
- pokročilé bloky (tmavě modrá)
- vlastní bloky (azurová)

Z těchto kategorií vás budou momentálně zajímat zejména první dvě – budete je potřebovat, aby se váš robot pohyboval a reagoval na svoje okolí. Se zbylými kategoriemi můžete samostatně experimentovat, pokud budete chtít vytvořit ještě komplexnější chování robota.





A*kční bloky* slouží k tomu, aby robot provedl nějakou fyzickou akci. Spadá sem otočení jednoho nebo dvou motorů současně, změna obsahu zobrazeného na displeji, přehrání zvuku a podsvícení tlačítek.



Bloky pro řízení programu dovolují vytváření složitějších programových konstrukcí, než je pouhá posloupnost (sekvence) příkazů – můžete pomocí nich vytvořit podmíněné větve a cykly. Je zde i blok pro čekání.



Senzorové bloky umožňují čtení hodnot z jednotlivých senzorů robota. Pomocí bloků pro operace s daty můžete s těmito hodnotami provádět jednoduché matematické a logické operace, jako je sčítání nebo porovnání. Můžete si vygenerovat i náhodnou hodnotu. *Pokročilé bloky* zpřístupňují pokročilé funkce robota, jako je přístup k souborům, posílání či přijímání zpráv a podobně. Je možné si dotvořit i *vlastní bloky* obsahující například často používané sekvence příkazů.

Vytváření programu pro robota probíhá prostřednictvím přetahování *instrukčních bloků* do hlavní plochy okna a jejich skládáním a zapojováním za sebe. K propojování bloků slouží šedé "konektory" nacházející se na levé a pravé straně každého bloku. Jednotlivé bloky pak budou prováděny právě v pořadí, v jakém jsou za sebe zapojeny. Bloky, které nejsou zapojeny do tohoto *instrukčního řetězce* (nenavazují na počáteční blok programu), jsou zašednuté a nebudou prováděny. Toho lze využít v případě, kdy budete chtít, aby se určitá část programu prozatím neprováděla, ale budete si ji chtít nechat uchovat pro budoucí použití – stačí tak "obejít" danou část pomocí přepojení předchozích bloků k následujícím, aniž byste museli danou část mazat.

Vyzkoušejte si přetahování a zapojování bloků za sebe.

Reflexe

Jede váš robot dopředu rovně a zatáčí správně? Zkoušeli jste nastavit různou rychlost a sílu motorů? Jak toto nastavení změnilo jízdní vlastnosti?

2. hodina – Vylepšujeme pohyb robota

Podrobně rozpracovaný obsah





Základ robota máte postavený, podvozek je hotový a také jste se naučili ovládat robota pomocí povelů – buď automaticky nebo pomocí programového prostředí. Teď se detailně vrhnete na cykly a na programové bloky.

- Postupně se připravujete na složitější a složitější činnosti. K tomu potřebujete jednoduchý řídící prvek – cyklus. Ten vám umožní opakování nějaké činnosti v kódu. Znát a ovládat cykly je opravdu užitečné, využijete to i v jiných oborech a programovacích jazycích. Také v dalších hodinách budete cykly používat.
- 2. V programovacím prostředí máte k dispozici velké množství bloků, které navíc můžete různě nastavovat. Až se s nimi lépe seznámíte, budete si moci vybrat takové, které se vám hodí nejvíce. Po robotovi budete chtít, aby uměl odlišné věci, takže uvnitř programu nakonec použijete větší počet těchto bloků zároveň.

Úkol 3: Vytvoření jednoduchého programu

Vytvořte jednoduchý program, který bude točit oběma motory, dokud senzor barvy nezaznamená černý povrch.

Je vhodné k tomu použít následující bloky:

- blok pro spuštění dvou motorů (Move Tank):
 - nastavte správně výstupní porty, ke kterým jsou motory připojeny (viz šipka 1)
 - můžete zvolit mód spuštění (viz šipka 2); na výběr jsou prostá změna stavu motorů (jejich neomezené zapnutí nebo vypnutí) nebo spuštění na předem stanovený rozsah pohybu (časový nebo míra otočení ve stupních či počtu otáček)
 - můžete nastavit rychlost otáčení motorů (viz šipka 3); při nastavení jako na obrázku se budou oba motory otáčet stejně rychle, ale každý na opačnou stranu



- blok pro cyklus (Loop):
 - nastavte typ/zdroj podmínky ukončení cyklu (viz šipka 1) pokud dojde ke splnění definovaného stavu, cyklus je ukončen a pokračuje se vykonáváním následujícího bloku





(pokud žádný blok nenásleduje, program skončí); ve vašem případě je konec cyklu určen barvou detekovanou senzorem barvy, proto zvolíte možnost *Colour Sensor* – *Colour*

- vstupní port, k němuž je připojen daný senzor, musíte také správně nastavit (viz šipka 2)
- zadejte správnou hodnotu, při jejíž detekci má cyklus skončit (viz šipka 3); v našem případě je to černá, které odpovídá číselná hodnota 1



Výsledný program může vypadat následovně:



Výše znázorněný program nejprve spustí motory a poté v cyklu čeká (konkrétně počká setinu vteřiny, pak provede měření barvy a tento postup opakuje), dokud senzor barvy nedetekuje černou. Po ukončení cyklu (tj. detekci černé barvy) vypne (zastaví) motory.

Alternativní a pravděpodobně intuitivnější přístup k témuž úkolu by byl uvnitř cyklu spouštět motory na předem definovanou dobu. Toto řešení je také možné, jeho nevýhoda je však ta, že samotné vyhodnocení podmínky (měření barvy) trvá nějakou dobu a robot by tak v praxi jezdil přerušovaně,





cukal by se. Zároveň by se mohlo stát, že by robot během doby vykonávání jedné iterace cyklu mohl přejet hledanou černou plochu, neboť podmínka se nevyhodnocuje uvnitř těla cyklu, ale až po jeho skončení. Tento přístup si můžete také vyzkoušet a budete mít možnost si jeho uvedené vlastnosti ověřit, jakmile postavíte funkčního robota.

Jakmile se budete domnívat, že jste vytvořili funkční řešení, můžete si jej uložit standardním způsobem (na liště vlevo nahoře zvolte *File→Save Project As…* a vyberte vhodné umístění souboru).

Reflexe

Jak se vám dařila práce v programovacím prostředí? Je prostředí dostatečně přehledné?

3. hodina – pracujeme se senzory

Podrobně rozpracovaný obsah

Máte základ stroje postavený, umíte cykly a bloky kódu. Teď přišel čas posunout vše na ještě vyšší úroveň a naučit vašeho robota pracovat se senzory. Proč? Podobně, jako vy máte své oči a uši, i roboti mohou vnímat okolní svět pomocí svých vlastních smyslů (čidel, senzorů). Senzorů existuje několik typů a všechny si je projdete a vyzkoušíte.

- Velmi důležitou částí stavebnice, a tedy i robotů, jsou senzory. Ty se vám budou hodit, aby váš robot "vnímal", reálný svět okolo. Výstup senzorů můžete vidět i přímo v řídicím bloku robota, můžete se proto podívat, jak který senzor v různých situacích funguje a lépe prozkoumat, jak je s nimi možné zacházet.
- 2. Tip pro ty nejrychlejší z vás možná už během této hodiny budete schopni navázat informace senzorů na programové příkazy z předchozích hodin, například pohyb motorů nebo zvukové upozornění, takže může třeba robot couvat od překážky nebo se před ní zastavit. Ostatní se nebojte, určitě na to bude čas a příležitost v příštích hodinách.

Úkol 4: Nahrání a spuštění programu na robotovi

Připojte *programovatelnou kostku* k počítači pomocí USB kabelu. V panelu v pravé dolní části okna programovacího prostředí byste měli vidět záznam o připojené jednotce. V záložkách můžete přepínat druh zobrazených informací – můžete si zobrazit například hodnoty detekované jednotlivými senzory, což může být užitečné pro zjišťování rozsahů, na které má robot reagovat, až budete vytvářet složitější program. Pomocí tří tlačítek umístěných svisle pod sebou úplně vpravo můžete nahrávat program (nebo jeho části) do robota a spouštět jej. Můžete sledovat, který *instrukční blok* je právě prováděn – jeho horní lišta se bude pohybovat. Vyzkoušejte si spuštění vytvořeného programu a ověřte, že ukončení funguje správně pomocí přiložení senzoru barvy na černý povrch. Pokud program nefunguje správně, pokuste se najít chybu a opravit ji. Pokud funguje správně, můžete si vyzkoušet různé režimy spouštění a zastavování motorů, nastavení rychlosti otáčení, případně doplnit akce robota o zobrazení obrázku na displeji nebo přehrání zvuku. Můžete také vytvořit složitější chování, například nastavovat různou rychlost motorů podle detekované barvy za použití bloku *Switch*.

Úkol 5: Seznámení se senzory

Stavebnice obsahuje kromě již zmíněného *senzoru barvy* i *ultrazvukový senzor vzdálenosti, gyroskop* a dva *senzory dotyku* (tlačítkové senzory). Připojte je ke *vstupním portům* a pomocí okna v pravé dolní části programovacího prostředí ověřte jejich funkčnost a prozkoumejte, jaké módy měření nabízejí, a jak tyto módy fungují.





Reflexe Zamyslete se, k čemu by získané hodnoty mohl robot ještě využít?

2.1.4 Dokončení základů programování robota – 3 hodiny

1. hodina – Senzor barvy

Podrobně rozpracovaný obsah

Co kdyby váš robot dokázal číst barvy na semaforu? Myslíte, že je to možné? Uměl by, podobně jako lidé, zastavit na červenou a rozjet se na zelený signál? Právě proto tu máme čidla, která dokážou vnímat barvy, a o tom je tato hodina.

Senzor, který každá z vašich skupin využije, je barevný senzor. Tento zde má výsadní postavení, a proto mu věnujeme speciální péči, jak po stránce stavby robota, tak po stránce propojení vstupu dodávaného senzorem s programem.

- 1. Zakomponování senzorů do robota může na první pohled vypadat jako přímočará záležitost, může však nastat několik komplikací. Ale nebojte, vedoucí programu vám vše jasně předvede a vysvětlí.
- 2. V závislosti na kvalitě osvětlení a barvě podlahy či podložky je nutné dát barevný senzor do vhodné výšky nad zemí, obvykle platí, že čím blíže země tím kvalitnější vstup. Přesunutím čidla blíže k zemi se také částečně omezuje vliv barvy osvětlení na jeho funkci.
- 3. Zde se vám může opět stát, že nejvhodnější cestou bude přestavět vašeho robota, protože jinak nepůjde přimontovat senzor dostatečně nízko.
- 4. Je důležité také vyzkoušet, jestli senzory a jejich zpracování správně fungují v takové situaci, která bude podobná finálnímu tvaru prostředí, ve kterém budou roboti jezdit. Dejte si například pozor na barvu podlahy a osvětlení v místnosti.

Tato činnost patří opravdu už mezi ty pokročilé. Také máte radost, kolik jsme se toho od začátku programu společně naučili? Dostáváme se dostatečně daleko, co se týče postupu programem, a vaše získané zkušeností vás krok za krokem směřují k finální podobě robota.

Reflexe

Lišilo se nastavení senzorů robota v odlišných světelných podmínkách? A co lidské oko? Myslíte, že roboti vnímají barvy lépe než lidé? Jak se naše vnímání liší?

2. hodina – Jízda po čáře

Podrobně rozpracovaný obsah

Naučíte vašeho robota jezdit po čáře. Bude ji automaticky sledovat. Abyste toho dosáhli, musíme si my sami vysvětlit, co je to podmíněné vykonávání bloků programu. A také to později vysvětlit vašemu robotovi.

Proto se budete v této lekci soustředit na rozbor vlastností jednotlivých senzorů a jejich kombinování s podmíněným vykonáváním bloků programu – detekce čáry na podložce.





- Správně umístěný senzor dodává robotu informaci o barvě, která se nachází právě pod ním. Této informace pak použijte k vytvoření požadovaného chování – je v dosahu čidla čára dané barvy? Sleduj ji.
- Aby mohl robot řádně reagovat na vstupy ze senzorů, musí v programu využít bloky umožňující větvení. Představte si více cest, které vedou k cíli. Na každé křižovatce se musíte rozhodnout. Přesně o tom je větvení. A co funguje v životě, funguje i v kódu programu.
- 3. Jaká budou konkrétní nastavení podmíněných příkazů, už samozřejmě záleží na potřebě daného případu. Robot může například obrátit směr jízdy po tom, co ztratí na senzoru snímanou barvu. Pokud žádnou barvu nevidí, může ji zkusit najít popojížděním a zatáčením, dokud zase barvu neuvidí.
- 4. Testování je nutné. Testování je základ práce programátora. Aby mohl robot fungovat v požadovaném prostředí, musí vaše skupina vyladit jeho programování a odstranit chyby a nedostatky, které se při jeho tvorbě dozajista objeví. Nejlepší cesta k odstranění takových chyb je pak opakované testování v podmínkách s výskytem různých komplikací, které mohou za běhu nastat. V průběhu celého testování budete pravděpodobně nacházet další problémy způsobené nedostatky v řešení předchozích úkolů a současně s prací se senzory se budou tyto problémy odstraňovat.

Reflexe

Kolikrát jste během této hodiny otestovali své řešení? V čem vám to pomohlo?

3. hodina – Finalizujeme pohyb po čáře

Podrobně rozpracovaný obsah

Samostatná tvorba komplexního programu pro pohyb – sledování čáry na podložce.

- Pro lepší pochopení práce se senzory a jejich propojení s programem zde zahrneme vytvoření cvičného programu, na kterém si osvojíte kvalitní postupy a získáte nějaké další zkušenosti. Toto cvičení nebudeme přímo využívat v závěrečném programu, ale pomůže vám s jeho implementací. Tento úkol je: Naprogramovat robota tak, aby jel po barevné čáře.
- 2. Plnění tohoto úkolu zahrnuje opět skupinovou práci, vymýšlení řešení, tvorbu programu a testování. V ideálním případě by zde již nemělo docházet k zásadním přestavbám robota.
- 3. Nějaký čas v tomto bloku je vhodné využít k vzájemné komunikaci mezi skupinami, předávání získaných zkušeností a porovnání kvality vybraných řešení.

Reflexe

Zamyslete se a řekněte, v čem vám vaše nynější stavba robota zkomplikovala programování.





2.1.5 Praktická práce s dalšími senzory – 3 hodiny

1. hodina – Senzor s tlačítkem

Podrobně rozpracovaný obsah

Víte, jak pracuje skutečný programátor? Nikdy nenapíše hotový kód na první pokus. Ale testuje, testuje a stále testuje. Pak vše vylepšuje podle výsledků předchozího testování. A přesně to budeme dělat i my.

Předpokládám, že v předchozích hodinách jste neměli úplně dokonalé výsledky s pohybem po čáře. To je v pořádku, na úrovni primitivního kódu to ani jinak fungovat nemohlo. Právě proto nyní vše ještě vylepšíte (téměř jako skuteční programátoři, lidé z praxe) a přidáte vašemu robotovi tlačítko! Ano, slyšíte správně – skutečně přidáte tlačítko, které bude fungovat jako automatické čidlo a reagovat na podněty. Samo bez vašeho zásahu. O čidlech jsme se už bavili v předchozích lekcích a nyní to spojíte s bloky a větvením.

- Chování na základě vstupu barevného čidla je zde mírně odlišné od předchozích lekcí. Robot totiž nesmí odjet z ohraničeného prostoru (vyjet ven). Uvědomte si prosím, že s nasbíranými zkušenostmi z předchozích bloků je čas zahrnout funkcionalitu senzorů do hlavního programu. Váš program ovládající robota je pak komplexnější než jednotlivé díly a spojuje předchozí úkoly dohromady. Snadno tedy dosáhnete požadované funkce.
- 2. Až to zvládnete, posunete náš projekt ještě dál. Co když se na trase objeví překážky? Může reálný robot v reálném skladu vrážet do regálů a stěn? Jak by se měl v takovém případě zachovat? Proto vyrobíte malý nárazník na přední část robota, který bude tento účel plnit.
- 3. Další překážkou ve splnění finálního úkolu budou objekty, které budou fyzicky bránit robotovi ve volné jízdě po ohraničené ploše. Tyto objekty budou příliš těžké na to, aby je robot odtlačil, ten proto musí tento případ rozpoznat a kvalitně na něj reagovat. Na tento podúkol je vhodné



použít dotykové čidlo také fungující jako tlačítko.





Reflexe

Co by se se vám hodilo k tomu, abyste mohli lépe a snáze testovat?

2. hodina – Senzory vzdálenosti

Podrobně rozpracovaný obsah

Určitě ale nechcete, aby váš robot bezcílně narážel do překážek. Už jste jej sice naučili se po nárazu otočit, ale mnohem lepší by bylo, kdyby překážku detekoval z větší vzdálenosti, nenarazil do ní a úspěšně se jí vyhnul. Vidíte, tady se už více blížíme reálnému světu reálných robotů. Přesně tak se totiž naši průmysloví roboti v reálném provozu chovají. Pojďme na to.

- 1. Další senzor, který přichází na řadu, je čidlo snímající vzdálenost k nejbližší překážce, tímto senzorem vylepšíte navigaci robota a doplníte nárazník z předchozí hodiny.
- 2. Využitím tohoto senzoru je pak možné nejen reagovat na náraz do překážky, ale předejít takové situaci úplně. Nárazník je vhodné stále zachovat, čidlo nemá perfektní zorné pole.
- 3. Efektivní využití obzvláště tohoto senzoru bude vyžadovat testování. Proto touto hodinou nekončíme a s čidly vzdálenosti budeme pokračovat i příště
- 4. Funkční program využívající všech senzorů budete postupně sestavovat v průběhu této a dalších hodin. Změny ideálně otestujte.

Reflexe

Čidlo vzdálenosti je v praxi velmi důležité. Jakou souvislost má tato aktivita s reálnými roboty v průmyslu? Zhodnoť, nakolik nyní lépe rozumíš problematice řízení robota. Posunulo se nějak tvé vnímání skutečných technických problémů návrhářů průmyslových robotů?

3. hodina – Využití senzoru na měření vzdálenosti

Podrobně rozpracovaný obsah

Ani senzor na vzdálenost není perfektním řešením v každé situaci, v této hodině se zaměřte na vylepšení vašeho algoritmu a spojení funkcí senzorů do jednoho funkčního programu.

- 1. Zde opět přichází na řadu testování. Budete testovat fungování robota v různých situacích, odhalovat slabiny řešení a zamýšlet se nad jejich případným odstraněním.
- V programu si můžete nastavit různé chování v závislosti na různé vzdálenosti, které senzor detekuje. Například pokud senzor vidí překážku 30 cm daleko, může zkusit zatočit, pokud 10 cm daleko, bude brzdit nebo couvat.
- 3. Při programování chování mějte na vědomí, že co je třeba udělat, závisí také na rychlosti vašeho robota, jak ostře dovede zatáčet, a na pozici samotného senzoru.

Reflexe

Který senzor vám přišel nejužitečnější? Který vám způsobil nejvíce potíží?





2.1.6 Zahájení prací na vlastním technickém řešení – 3 hodiny

1. hodina – Bude turnaj!

Podrobně rozpracovaný obsah

Pozor, pozor, velký turnaj automatických robotů se blíží. Hledáme soutěžní kandidáty. V nejbližších hodinách navrhnete, sestrojíte, naprogramujete a otestujete vašeho vlastního robota.

1. Představíme si pravidla závěrečného turnaje, která musí každý zúčastněný robot respektovat.

Pravidla jsou následující:

- a. Robot ponese Vive tracker, jehož pozice bude reprezentovat pozici cíle ve virtuální realitě. Na tento cíl bude člen konkurenčního týmu střílet virtuální šípy. Bude sledován počet zásahů za jednu hru (hra trvá pevnou dobu). Tým tvůrců robota, který obdrží nejméně zásahů, vyhrává. Úkolem robota tedy bude vyhýbat se virtuálním šípům, tj. pohybovat se (resp. pohybovat trackerem) způsobem, který činí zásah co možná nejobtížnějším rychle, nepředvídatelně atd.
- b. Tracker musí být umístěn na dobře viditelném, odkrytém místě. Robot nesmí v rámci své funkcionality záměrně zakrývat tracker žádným neprůhledným objektem.
- c. Vyjede-li robot během hry celým objemem z vytyčené oblasti (černá čára na bílé podložce), bude tým diskvalifikován.
- d. V oblasti pro pohyb robota budou umístěny překážky, kterým by se měl robot vyhnout. Při vytlačení překážky mimo vytyčenou oblast bude týmu udělena penalizace v podobě přičtení zásahu navíc.
- e. Povoleno je využití všech dostupných senzorů, avšak lze používat pouze jedno tlačítko a jeden kanál na dálkovém ovladači robota.
- f. Žádný ze žáků nesmí během hry robota nijak fyzicky posouvat nebo mu naopak bránit v pohybu.
- g. Startovní pozici robota i překážek určuje tým, který bude střílet, robot i překážky však musí být celým objemem uvnitř vytyčené plochy a robot musí být umístěn podvozkem na podložku (tj. nesmí být položen "vzhůru nohama" nebo na některou z překážek). Robot také nesmí být překážkami zcela obklopen.
- Využijte vašeho robota, prototypování a získávání základních zkušeností se chýlí ke konci. Nebo ještě lépe – postavte nového robota úplně od začátku. Během minulých lekcí jste se toho hodně naučili, využijte to nyní při návrhu soutěžního stroje.

Reflexe

Co by mohly být přednosti vašeho robota? Máte již vymyšlenou týmovou taktiku? Jaká strategie se vám osvědčila v rámci skupinové domluvy? Měli jste nějaké problémy se shodnout? Jak jste je případně vyřešili?

2. hodina – Co musí robot umět?

 Dobře si promyslete vaši taktiku, rychlejší robot bude snáze uhýbat, ale pomalejší robot je jistější volba.





- 2. Můžete zvolit riskantnější, ale rychlejší pohyb pouze vpřed, nebo můžete například couvat od překážek.
- 3. Můžete se také, pokud budete cítit potřebu, vrátit k nějakému pracovnímu listu z dřívějších hodin a zopakovat si nebo dokončit některý z předchozích kroků. Pracovní listy můžete použít i jako jakousi referenci k používání některých prvků.
- 4. Je vhodné nepodceňovat skupinovou domluvu, na chvíli se zastavte od tvoření a promluvte si o tom, jak dále pokračovat, jak se nyní moderně říká, není nad pořádný brainstorming.
- 5. Nyní nezbývá než popřát, ať vám jde stavba dobře od ruky. Když si s něčím nebudete vědět rady, obraťte se na lektory, určitě vám rádi poradí.

3. hodina – Pokračujeme ve stavbě soutěžního robota

Podrobně rozpracovaný obsah

- V této a následujících několika hodinách nebudeme přidávat nové prvky ani představovat nové postupy a příklady. Budete pokračovat ve stavbě soutěžních robotů a jejich průběžném testování s cílem vytvoření unikátního řešení, které bude splňovat soutěžní požadavky. Cílem by měla být realizace konstrukčního i programového řešení robota na základě návrhu požadované funkcionality.
- 2. Lektoři v těchto hodinách nebudou aktivně vést, ale budou vám k dispozici jako poradci. Nebojte se o radu kdykoliv požádat! Při stavbě robota můžete narazit na spoustu různých komplikací, nejčastějším problémem bývá chybička v konstrukci podvozku, která způsobuje, že robot nedokáže jet rovně, špatně zatáčí nebo se zadrhává při couvání. Zkontrolujte si, jestli jsou vaše motory symetricky umístěné a hmotnost vašeho robota není vychýlená. (Například pozor na programovací blok, je těžký!)

Reflexe

Máte všechny komponenty, které potřebujete? Zkontrolovali jste správné zapojení všech součástek? Jaký způsob práce se vám v této více samostatné aktivitě osvědčil? Nakolik vnímáte přínosné moci si zvolit vlastní postup a tempo práce?

2.1.7 Dokončování vlastního robota a prezentace VR pro pochopení návaznosti v rámci plánu programu – 3 hodiny

1. hodina – finální tvar

Podrobně rozpracovaný obsah

- 1. Nyní byste měli mít všechny potřebné základy k sestavení vašeho robota. Pusťte se tedy do práce.
- 2. Při stavbě myslete na to, co jste se již naučili a nezapomeňte na cíle, které váš robot musí splňovat.
- 3. Když si nebudete jistí, požádejte lektory o zapůjčení VR trackeru a ozkoušejte si, zdali vám jde nasadit na vaši konstrukci.







2. hodina – ověřujeme a testujeme

Podrobně rozpracovaný obsah

Znovu vám doporučuji nezapomínat na testování. Zde máte některé důležité body, které by vám robot měl splňovat a které můžete snadno otestovat, tyto body nejsou přímo vašimi cíli, ale jsou instrumentální k jejich efektivnímu dosažení.

- Robot by se neměl převracet v zatáčkách.
- Robot by se neměl příliš zasekávat při couvání.
- Robot by neměl přímo, bez snahy o brzdění nebo vyhnutí, najíždět do překážek.
- Robot by měl být schopen poznat, že najel na barevnou čáru.

Reflexe

Jste spokojeni s tím, jak váš robot jezdí? Líbí se vám virtuální realita?

3. hodina – robot bude hotov

Podrobně rozpracovaný obsah

- 1. V tomto bloku se nadále věnujte samostatné práci s případnou asistencí ze strany lektorů.
- 2. Souběžně si mohou jednotliví účastníci programu vyzkoušet používání virtuální reality na aplikaci The Lab (dostupné zdarma na platformě Steam). S vyzkoušením VR aplikace bude zároveň třeba předvést základní ovládání VR a základní bezpečností požadavky kolem VR.
- 3. Podstatné je, že máte nyní možnost se zde seznámit s technologií, ke které většina studentů nejspíš nemá přístup. Můžete si hrát, zkoušet možnosti ovládání a odpočívat v prostředí virtuální reality, což je klíčové k vašemu dalšímu úspěšnému postupu.

Váš robot pak může vypadat například takto:







Reflexe

Vyzkoušeli jste si nasazení VR trackeru? Ověřili jste, že váš robot splňujete všechny zadané podmínky? Bylo nutné vašeho robota vícekrát přestavovat – pokud ano, tak v čem byl problém? Jak jste spokojeni se svou výslednou prací? Jak se vám pracuje s aplikací The Lab?





2.2 Unity 3D VR - 12 hodin

Další část programu je založena na využití virtuální reality. Naučíte se v ní, z čeho se skládá scéna a jak nějakou vytvořit. Zde budeme využívat aplikaci Unity. Tento primárně herní engine je pro výukové účely (a také nízkopříjmové komerční účely) zcela zdarma a dostupný na internetu ke stažení pro kohokoliv.

Můžete se podívat na stránky . Pro školní účely si možná ani nebudete muset zřizovat účet, ale pozor, Unity se hodně vyvíjí, takže se věci rychle mění a určitě se vám bude hodit se porozhlédnout, co je nového.

2.2.1 Úvod do Unity – 3 hodiny

1. hodina – digitální scéna

Podrobně rozpracovaný obsah

Stejně jako ve skutečnosti, na to, aby bylo něco ve scéně vidět, potřebujete tři věci. Věc, na kterou se koukáte, nějaké světlo, které ji nasvítí, a nakonec nějakého pozorovatele, na počítači zastoupeného kamerou, která nám poskytuje okno do digitálního světa s pomocí grafické karty, monitoru a velké spousty matematiky.





Pracovní list 3: Unity – základní manipulace s objekty – část. 1

Po zapnutí Unity bude okno editoru vypadat přibližně takto:



Rozprostření oken nemusí být totožné, ale všimněte si několika hlavních bodů.

Označená okna jsou zleva: Scene, Hierarchy a Inspector.






Pro vytvoření jednoduchého objektu kliknete pravým tlačítkem kamkoliv v oblasti okna *Hierarchy*. V otevřené nabídce dále vyberete *3D Object* a z nabídky objektů pak *Cube*.



Všimněte si poté nově vytvořeného objektu ve *Scene*, jeho ekvivalentu "Cube" v *Hierarchy* a nově otevřeného detailní náhledu objektu v *Inspectoru*.



Náhled scény v okně Scene umožňuje výběr objektu kliknutím a přímé úpravy.

Okno *Hierarchy* funguje jako zjednodušený seznam objektů ve scéně. Je zde také možné označovat objekty, měnit jejich název a upravovat hierarchii objektů ve scéně. Položka ve scéně se nazývá





GameObject. Inspector v pravé části obrazovky pak představuje detailní náhled vybraného *GameObjectu* ve scéně. Ten slouží k nastavení vlastností objektu skrze takzvané *Komponenty*.

Reflexe

Překvapilo vás, z čeho se skládá virtuální scéna? Líbí se vám prostředí Unity?

2. hodina – editor virtuálního světa

Podrobně rozpracovaný obsah

- 1. Další část programu je založena na využití virtuální reality. Abychom vytvořili virtuální scénu a také s ní pracovali, potřebujeme nějaké rozhraní a sadu vhodných nástrojů. Tyto prostředky poskytuje herní engine Unity, jehož editor je ve vhodné pozici snadné učební křivky a velkého množství užitečných nástrojů. V první části tohoto bloku představíme Unity a jeho nejdůležitější části. Výuka Unity je zahrnuta v sérii pracovních listů zaměřených na Unity.
- V první etapě výuky se zorientujete v oknech editoru a poznáte základních sady nástrojů (Scene, Game, Project, Hierarchy, Inspector), které poskytují přístup k hlavním částem projektu. Pro tuto ukázku je připraven pracovní list.
- Další na řadě je pak vyzkoušení ukázkových scén. Některé příklady jsou dodány od vývojářů nástrojů, se kterými se v programu pracuje, další ukázkovou scénu tvoří základ pro virtuální scénu tohoto kurzu.





Unity – základní manipulace s objekty – část. 2

Po následování kroků z předchozího listu by mělo okno Unity vypadat přibližně takto:



Na výše uvedeném obrázku pak vlevo nahoře vidíte panel pro manipulaci s objektem a pohyb po scéně. Těmito nástroji je možné přímo ze scény měnit vlastnosti *Komponenty Transform* označené vpravo. Tato komponenta odpovídá pozici, natočení a měřítku objektu ve scéně. Každý *GameObject* má právě jedno *Transform*.

Image: Cube Image: Static Tag Untagged # Layer Default	•
Tag Untagged ‡ Layer Default	•
	<u>.</u>
🔻 🙏 Transform 🛛 🗐 न	¢,
Position X 2.942581 Y -1.23 Z -0.12378	6
Rotation X 20.404 Y 16.752 Z -14.354	
Scale X 2.2322 Y 1.5798 Z 2.6341	
🔻 🧾 Cube (Mesh Filter) 🛛 🗐 🕂	¢,
Mesh 🔡 Cube	0
▼ 🛃 🗹 Mesh Renderer 🛛 🔯 🚽	\$,
Light Probes Blend Probes	•
Reflection Probes Simple	\$
Anchor Override None (Transform)	0
In Deferred Shading, all objects receive shadows and get per-pixel reflection probes.	
Cast Shadows On	•
Motion Vectors Per Object Motion	+
Lightmap Static	
To enable generation of lightmaps for this Mesh Renderer, please enable the 'Lightmap Static' property.	e
Rendering Layer Mask Everything	+
▶ Materials	
Dynamic Occluded 🗹	
🔻 🤪 🗹 Box Collider 🛛 🔯 🖓	¢,
🔥 Edit Collider	
Is Trigger	
Material None (Physic Material)	0
Center X 0 Y 0 Z 0	
Size X 1 Y 1 Z 1	
DefaultHDMaterial	\$,
Shader HDRenderPipeline/Lit	•
Add Component	



EVROPSKÁ UNIE Evropské strukturální a investiční fondy Operační program Výzkum, vývoj a vzdělávání



🚝 Hierarchy 🗈 Project	
Create * Q*All	
🔻 🚭 work text scene*	*≡
Main Camera	
Directional Light	
Cube	



Z panelu nacházejícího se vlevo nahoře nad oknem scény nás budou zajímat pouze první čtyři tlačítka. Ikona nejvíce vlevo

(Obrázek bílé ruky) umožňuje pohyb po scéně se stisknutím levého tlačítka myši. Následující tlačítko, druhé v pořadí (na obrázku označené), slouží ke změně pozice objektu ve scéně. Viz Ilustrace 1. Další tlačítko poté k úpravě natočení objektu ve scéně (Ilustrace 2) a čtvrté pak ke změně velikosti (Ilustrace 3). Při každém nastavení se kolem vybraného objektu objeví manipulační značky. Tyto značky se dají pomocí levého tlačítka myši zachytit a změnou jejich pozice se pak upraví vlastnosti objektu.



V Komponentě Transform je pak možné sledovat změny v odpovídajících hodnotách.

Position odpovídá pozici objektu, Rotation jeho natočení a Scale poté jeho velikosti.





▼人 Transform						💽 🕸 🌣,
Position	Х	2.942581	Y	-1.23	Z	-0.123786
Rotation	Х	20.404	Y	16.752	Z	-14.354
Scale	Х	2.2322	Y	1.5798	Z	2.6341

Reflexe

Přijde vám manipulace s objekty intuitivní? Která část byla pro vás nejsnazší? Se kterou částí jste měli největší problémy? Kterou aktivitu byste případně potřebovali příště zopakovat?

3. hodina – skládačka

Unity – základní manipulace s objekty – část. 3 – Vytváření hierarchie objektů



Ilustrace 1: Unity – objekty v okně hierarchie

Po dokončení předchozích dvou částí dokážete nyní v Unity vytvářet objekty různých tvarů a ty následně upravovat pomocí editace jejich transformace (*Transform*). Dalším velmi důležitým nástrojem je pak skládání objektů do hierarchií. Hierarchie objektů tvoří strom, ve kterém objekt níže v hierarchii (potomek, v Unity *child*) dědí (přejímá) transformaci objektu v hierarchii výše (rodiče, v Unity *parent*).

Dědění transformace v praxi znamená, že pokud změníte některou z vlastností transformace objektu (pozice, rotace, velikost), tato změna se aplikuje i na všechny objekty v hierarchii níže (potomky).

V příkladu hierarchie v Ilustraci 1 vidíte jako nejvyšší objekt se jménem "SpaceShip", tento objekt je předkem všech ostatních objektů, které jsou v hierarchii zařazeny pod něj (Unity nám rodičovské objekty označuje s šipkou a jejich potomky odsazuje lehce doprava). Nejvyššímu rodiči se také říká kořen (*root*) hierarchie. V závěru se tedy při pohnutí s objektem "SpaceShip" pohnou i objekty "Model", "Body", "Engines", "Engine-Top" a "Engine-Bottom". Pokud byste změnili transformaci objektu "Engines", změnili byste tím krom něj ještě objekty "Engine-Top" a "Engine-Bottom".

Tuto strukturu vytvoříte jednoduchým přetažením objektu přes jiný objekt v okně hierarchie.

Poznámka: Unity nazývá okno "hierarchie" to okno, které obsahuje všechny objekty ve scéně, tyto objekty mohou samy tvořit vlastní hierarchie. Tento název je zde, protože samotnou scénu můžeme považovat za kořen hierarchie, ve které jsou všechny objekty ve scéně.

Poznámka: 2: Jak udělat objekt barevně, se naučíte později.





Příklady úprav transformací v různých stupních hierarchie:

I	TE Hierarchy Project 2 1	Project	a 1	O Inspector Project 🔒 I
🛠 💵 🔻 Gizmos 🖙 🔍 All	+ * [9: All	+ • 9	🐉 🛷 🜟 👳 11	
ý ·	▼	V + Favorites	Assets > Scenes	Tag Untagged • Layer Default •
₩ ₩	⑦ Directional Light ♥ ⑦ SpaceShip ♥ ۩ Model	Q. All Models Q. All Prefabs Q. All Scripts	# ProjectileBehav	▼ <mark>↓ Transform</mark> ● ‡ ! Position × 0 Y 0 Z 0
	Engines Engine-Top	v In Assets	 ShipBehaviour SpaceShipBody SpaceShipMoto 	Rotation X 0 Y 0 Z 0 Scale X 1 Y 1 Z 1
	Body	► IIIn Test ► IIIn Packages	≪J Test Scene	Add Component

Ilustrace 2: Výchozí stav



Ilustrace 3: Otočení s objektem "Engines" způsobí otočení s "Engine-Top" a "Engine-Bottom"







Ilustrace 5: Create Empty – vytvoření prázdného uzlu v hierarchii



Ilustrace 4: Otočení pouze s "Engine-Top" otáčí pouze tento objekt

Při vývoji aplikace v Unity je obvykle vhodné uvažovat nad strukturou jednotlivých objektů podle toho, jaké bude jejich využití. Unity pak umožňuje vytváření prázdných uzlů (*Empty* viz Ilustrace 2), které při vytvoření obsahují pouze komponentu *Transform* (je samozřejmě možné přidat do nich další komponenty). Příklady *Empty* v ukázce jsou objekty "SpaceShip", "Model" a "Engines".





Další pracovní listy rozebírající například jak nabarvit objekty ve scéně, najdete v příloze pracovních listů.

Reflexe

Jaký objekt ve scéně jste vytvořili? Šlo vám skládání objektů, jak jste chtěli?

2.2.2 Částicové efekty v Unity – 3 hodiny

1. hodina – vizuální efekty

Podrobně rozpracovaný obsah

- První a základní způsob úpravy scény, je její grafická úprava. Unity nabízí velké množství nástrojů k tomuto určených. Dále do aplikace také můžete načíst grafické soubory (tzv. asety) z jiných programů. Tato část projektu je navíc nejvíce závislá na vámi zvoleném postupu výuky a vybraných částech úpravy scény (viz metodická část).
- Jeden z nejdůležitějších stavebních kamenů počítačových efektů jsou částicové efekty. Unity nabízí editor takovýchto efektů (od 2020 dokonce dva různé). Ukážeme si, jak s editorem pracovat. Můžete zde také postupovat dle pracovního listu zaměřeného na toto téma.
- 3. Nejlepší způsob, jak se naučit vytvářet tyto efekty, je pak samozřejmě vyzkoušet si nějaké vyrobit a věnovat nějaký čas experimentům a metodě pokusu a omylu.
- 4. Tato a následujících několik hodin vás naučí pracovat s částicovými efekty v Unity. Po projití jednotlivých podkapitol a ovládnutí základních mechanizmů částicových efektu, si zvolte sami, jaký efekt byste chtěli vytvořit, a pak jeho tvorbu realizujte. Zajímavé efekty jsou například ohniště, mraky, kouř, déšť a další.





Unity – částicové efekty – část 1.

Pro vytvoření částicového efektu klikněte pravým tlačítkem kamkoliv v oblasti okna *Hierarchy*. V otevřené nabídce dále vybereme *Effects* a z nabídky objektů pak *Particle System*.



Po vytvoření částicového efektu by mělo okno Unity vypadat přibližně takto:



V levé části vidíte samotný částicový efekt v okně scény. Vpravo dole v okně scény pak okénko s nadpisem *Particle Effect*, které slouží k základnímu ovládání ukázky efektu.





V pravé části obrazovky se nachází samotná komponenta částicového efektu. Většina ovládání je provedena právě skrze tuto komponentu.

▼人 Transform			🔯 🕸 🖗
Position	X 3.81026	5 Y -3.458	354 Z -1.05113
Rotation	X -90	Y 0	Z 0
Scale	X 1	Y 1	Z 1
V V Particle Syst	tem		🔯 🎝 🔅
			Open Editor
Particle System			
Duration	5.00		
Looping	Z		
Prewarm	-		
Start Delay	0		•
Start Lifetime	5		•
Start Speed	5		•
3D Start Size			
Start Size	1		•
3D Start Rotation			
Start Rotation	0		•
Flip Rotation	0		
Start Color			•
Gravity Modifier	0		•
Simulation Space	Local		\$
Simulation Speed	1		
Delta Time	Scaled		\$
Scaling Mode	Local		\$
Play On Awake*	×.		
Emitter Velocity	Rigidbody		\$
Max Particles	1000		
Auto Random Seed	×.		
Stop Action	None		\$
✓ Emission			
✓ Shape			
Velocity over Lifetime	2		
Limit Velocity over Li	fetime		
Inherit Velocity			
Force over Lifetime			
Color over Lifetime			
Color by Speed			
Size over Lifetime			
Size by Speed			





*	S 🔀 🖾 🏵 🛛 🕬 Pivot	@Local		► II	M			Collab •	Account • Layers • La	ayout -
C +		©Lecil BAset Stor Droller	© Annate	Public Street	Contractory (Diffusion) Contractory (Diffusion) Contractory (Diffusion) Contractory Contreatory Contractory Contractory Contractory Contr	Core : Core	Asch - S cont Gamplesone Genetication Conten	Collabor Col	Account • Layers • Le Project • 0 Insector virten • Layer (Perfect × 3 # 000 × 10 × 2000 ×	yout • • • • • • • • • • • • • • • • • • •
Frame Debu	Console			Show Only Selected					Kodimizen I	Remove
Contract Contract										

V následujícím obrázku je pak předvedeno, jakým způsobem je možné změnit barvu částice v efektu. Kliknutí do, ve výchozím stavu bílého, rámečku otevře nabídku pro nastavení barvy. Tato barva pak bude aplikována na každou částici v efektu.

Ve spodní části *Particle System* se nachází položka *Renderer*, která vám umožní nastavit vlastní obrázek na místo výchozí částice, která je jen rozmazaný kruh. Toto se hodí při vytváření složitějších efektů.

Reflexe

Jaký efekt byste chtěli nejraději vytvořit? Chovaly se částice ve vaší scéně, jak jste chtěli?

2. hodina – chování částic

Podrobně rozpracovaný obsah

- Částicové efekty vám umožňují vytvářet velkou škálu vizuálních grafických efektů, dejte si pozor, ať při přidávání částic a na další různá místa nenapíšete příliš velké číslo, počítače nemají neomezený výkon, takže doporučuji držet množství generovaných částic maximálně kolem stovek až tisíců za vteřinu a maximální počet existujících částic někde kolem desítek tisíc. (I tak bude váš efekt náročný na zdroje počítače.)
- Pokud jste vy (a vaši lektoři) dostatečně odvážní, můžete vyzkoušet i novější metodu pro vytváření vizuálních efektů nazývanou Visual Effect Graph, tyto nástroje jsou citelně složitější, a proto pro ně v tomto kurzu nezbyl prostor.





Unity – částicové efekty – část 2.

POSICION	A	2.010202		-9.490994	4	-1.09119
Rotation	X	-90	Y	0	Z	0
Scale	x	1	Y	1	z	1
Particle	e System				pen	Editor
Particle Sys	tem					+
Duration	5.00	Ê.				
Looping	V					
Drawarm						

Předchozí list končil nastavením barvy částic efektu, další možnosti úpravy jsou možné po kliknutí na tlačítko *Open Editor*, v pravé horní části komponenty.

Po kliknutí na toto tlačítko se otevře nové okno s nadpisem *Particle Effect*. V tomto okně vidíme náhled komponenty a můžeme zde její hodnoty upravovat. (Upravovat hodnoty efektu lze i bez otevření editoru v samostatném okně.)







Zde můžeme změnit například dobu života jedné částice, její počáteční rychlost anebo její velikost.

Particle Effect		Particle
Particle System	. 0 ^	 Optimize Remove Transfor
Duration Social Laping S.00 Laping S.00 Start Delay O Start Dieme Start Speed O.79 JO Start Gielen JO Start Rotation BJO Start Rotation BJO Start Rotation Dieme Start Rotation Start Celer Gavarty Medifer Gawarty Medifer Data Time Scaled Scaling Mede Local Play On Avake V Emitter Volocity Rigidb Mac Particles 1000	edy e	Position Rotation Biole Porticle Porticle Shader
✓ Emission ✓ Shape		
Velocity over Lifetime		

Zde vidíme příklad takovýchto úprav, jednotlivé částice jsou mnohem větší a pomalejší, ale doba jejich trvání byla zkrácena.

Reflexe

Zkusili jste vytvořit více částicových efektů? Jaké možnosti nastavení vám nejvíce schází?

3. hodina – generujeme

Podrobně rozpracovaný obsah

Unity – částicové efekty – generátor částic

Tento list slouží k ukázce další dvou důležitých podkomponent částicového efektu. *Emission* a *Shape*, odpovídají množství a načasování generovaných částic a tvaru, ze kterého jsou generovány. Tato nastavení najdeme pod *Particle System* z předchozího listu.





Pause Stop		Resimulate Show Bounds
Delta Time	Scaled +	Obtimize Remov
Scaling Mode	Local	
Play On Awake*	V V	
Emitter Velocity	Rigidbody ‡	
Max Particles	1000	
Auto Kandom Seed Stop Action		
VEmission	-one +	
/ Shane		
Valacity over Lifet		
Limit Velocity over	r l ifatima	
Inherit Velocity		
Force over Lifetim		
Color over Lifetim		
Color by Speed		
Size over Lifetime		
Size by Speed		
Rotation over Life	ime	
Rotation by Speed		
External Forces		
Noise		
Collision		
Triggers		
Sub Emitters		
Texture Sheet Ani	nation	
Lights		
🔿 Trails		
Custom Data		
🗸 Renderer		
• ())*	

Rate over Time	10			*
Rate over Distance	0			
Bursts				
Tim	Count	Cycles	Oterval	
List is npty				
			1	_

V části *Emission* najdeme dva způsoby nastavení množství generování částic, *Rate over Time* a *Rate over Distance* umožňují nastavit generování částic v závislosti na nějaké jiné veličině. *Rate over Time* umožňuje nastavit, kolik částic se vytvoří za vteřinu. Pro pohyblivé efekty je pak možné nastavit množství vygenerovaných částic za jednotku vzdálenosti.

Další možností je nastavit generování částic pomocí metody *Burst*. Tento způsob generování částic slouží k vytvoření "dávek" nebo "vln" částic. Idea tohoto způsobu není kontinuální vytváření částic, ale nárazové vytvoření většího množství v jeden okamžik. Tlačítkem s ikonou "+" na spodní straně okna s nadpisem *Bursts* vytvoříte novou vlnu částic.

Time	Count	Cycles	Interval	
0.000	30	1	0.010	





V nově vzniklém řádku pak můžeme nastavit několik atributů. První z hodnot, *Time*, je zpoždění vlny od stratu efektu, druhá, *Count*, pak určuje samotné množství vygenerovaných částic v jedné dávce. Třetí parametr, *Cycles*, slouží k nastavení počtu dávek, které budou vytvořeny během trvání efektu, tento parametr se dá dále změnit z čísla na hodnotu *Infinite*, neboli nekonečno, při kliknutí na malou šipku napravo od řádku s číslem. Čtvrtou hodnotou je pak *Interval* a ten určuje dobu mezi jednotlivými dávkami.

V podokně *Shape* je pak možné nastavit například atribut *Shape*, který určuje, v jakém tvaru budou částice na počátku generovány. Následující obrázek představuje možné nastavení s tvarem *Circle*, kde jsou částice generovány po obvodu kružnice.



Při vytváření částicového efektu může být také užitečné změnit hodnotu *Max Particles*, která se nachází v *Particle System*, pro upravení maximálního počtu částic, které mohou existovat v daný moment. (Žádná další částice nebude vytvořena generátorem, dokud bude počet částic stejný jako je nastavené číslo v *Max Particles*.)

Reflexe

Přežilo Unity vaše experimentování? Jaký vizuálně působivý efekt se vám podařilo vyrobit?





2.2.3 Osvětlení a materiály v Unity – 3 hodiny

1. hodina – zjednodušené Slunce

Podrobně rozpracovaný obsah

- Jedna z klíčových částí scény je osvětlení. V těchto hodinách si společně ukážeme, jaké typy světla v Unity existují a jak se dají modifikovat. Tato část je opět pokryta v jednom z pracovních listů.
- Osvětlení může výrazným způsobem ovlivnit již existující scénu, efekt nebo jiný grafický aset. Proto je důležité, abyste si vyzkoušeli, jaké možnosti v tomto ohledu Unity poskytuje, což je opět nejlépe provedeno metodou vlastních experimentů.
- 3. Nejjednodušším typem světla, které můžete použít ve scéně, je světlo směrové, jehož pozice nemá žádný efekt na jeho působení. Tento typ světla použijte tehdy, pokud chcete scénu osvětlenou stejně jako Sluncem. Tento typ najdete obvykle v Unity ve scéně již při jejím vytvoření pod názvem Directional Light.

Reflexe

Rozumíte, proč stačí informace o směru, a nikoliv o pozici zdroje světla? (Tip: Je to daleko.). Pobavte se o tomto problému ve skupině. Je to všem dostatečně jasné? Dostavil se "aha efekt"? ^(C)

2. hodina – přidáme žárovky

Podrobně rozpracovaný obsah

Unity – bodové světlo

Na tuto úlohu je vhodné mít připravenou scénu, na které bude vidět změna osvětlení. Příklad může vypadat například takto:







Pro lepší představu je možné na začátku smazat světlo, které se vytvořilo se scénou. Toho je možné dosáhnout vybráním položky *Directional Light v Hierarchy*. Není to však nutné. (V následujících obrázcích je pro lepší ilustraci světlo smazáno.)



Začnete vytvořením jednoduchého bodového světla. Vytvořený objekt bude, jako každý *GameObject* v Unity, tvořen komponenty. U *Transform* bodového světla má význam pouze pozice, na ostatních hodnotách nezáleží. Nastavení světla pak naleznete v komponentě *Light* a jejích pod-komponentách.

▼人 Transform						(n 1	¢,
Position	Х	2.394531	Y	2.97	Z	-3.6	59238	3
Rotation	Х	0	Y	0	z	0		
Scale	Х	1	Y	1	Z	1		
🔻 💡 🗹 Light						(<u>ا:</u> []	¢,
▶ Features								
▶ Shape								
▶ Light								
▶ Shadows								
🔻 🏟 🗹 HD Additiona	l Ligl	ht Data (S	crij	ot)		(<u>ا:</u> [\$,
🔻 📾 🛛 Additional Sh	adov	w Data (Sc	rip	t)		(김 규	\$,
	Ad	d Compon	ent					

Po vytvoření světla může scéna vypadat nějak takto:







Z nastavení v komponentě Light vás budou zajímat především pod-komponenty Shape a Light.

♥ 🖗 🗹 Light ▶ Features 🛛 🦯		🗐 🕸 🌣
🔻 Shape 🦯		
Туре	Point	+)
Range	10	
Max Smoothness		01
🔻 Light 🧹		
Use color tempera	ten-	
Color		19
Intensity	600 Lumen	+
Indirect Multiplier	1	
Mode	Mixed	•
Cookie	None (Texture)	0

Shape ovlivňuje, jak již název napovídá, tvar světla, je zde možné změnit typ světla, jeho dosah a jemnost.

V *Light* budete měnit parametry *Color* a *Intensity*. *Color* ovlivňuje barvu světla a je možné je nastavovat jako klasickou barvu nebo jako teplotu světla. *Intensity* pak představuje sílu světla.





Výsledek pak může být například takovýto:



Reflexe

Líbí se vám více směrové nebo bodové světlo? V jakých situacích byste raději použili jedno než druhé?

3. hodina – hrubost a lesk

Podrobně rozpracovaný obsah

- Jeden z důležitých faktorů, který ovlivňuje, jakým způsobem je grafický objekt vykreslen na obrazovku, je materiál. V této hodině představíme způsob importu textury a vytvoření materiálu pro určení vizuálních vlastností vykreslovaných objektů. Tím umožníme studentům nastavovat vlastnosti objektů, jako jsou barva, lesk nebo hrubost povrchu.
- Existuje také zobrazování na fyzikálním základu (Physically based rendering), které je novější a perspektivní způsob, jak vykreslovat scénu a doplnit ji o efekt způsobený reálnými vlastnostmi světla.
- 3. Dále si vyzkoušejte manipulaci se světlem v Unity scéně.

Unity – vytvoření materiálu – část 4.



Ilustrace 1: Objekt vytvořený nastavením pozice a velikosti kostek (cube), složený do jedné hierarchie.





Začínáme s jednoduchým objektem, jehož výchozí barva bude šedá. (Unity *Default-Material*). Materiály jsou velmi důležitým aspektem počítačové grafiky, základní typy materiálů ovlivňují vzhled povrchu objektů, například barva (*albedo*) nebo odrazy světla (*normal map*). Často jsou součástí materiálů textury (*texture map*), které umožňují načíst a namapovat obrázky na modely (*mesh*).

Folder		Create	1
C# Script		Show in Explorer	
Shader	>	Open	
Testing	>	Delete	
Playables	>	Rename	
Assembly Definition		Copy Path Alt+Ctrl+C	2
Assembly Definition Reference		Open Scene Additive	
TextMeshPro	>	View in Dackage Manager	
Scene		view in Fackage manager	
Prefab Variant		Import New Asset	
Audio Mixer		Import Package	
		Export Package	
Material		Find References In Scene	
Lens Flare		Select Dependencies	
Render Texture		Refresh Ctrl+F	R
Lightmap Parameters		Reimport	
Custom Render Texture		Reimport All	
Sprite Atlas		Fates at From Deefah	
Sprites	>		
Tile		Run API Updater	
Animator Controller		Update UIElements Schema	
Animation		Open C# Project	
Animator Override Controller	1		i.

Materiál vytvoříme v Unity snadno, pravým kliknutím v okně Assets => Create => Material.

Ilustrace 2: Vytvoření nového materiálu

Nově vzniklý materiál můžeme okamžitě pojmenovat. Jako například v ilustraci 3 materiál "SpaceShipBody".







Ilustrace 3: Materiál a jeho nastavení

Kliknutí na barvu u možnosti *Albedo* umožní nastavení barvy materiálu. Tato barva ovlivňuje texturu, pokud nějaká je vložená. (Texturu je možné nahrát do Unity přetažením obrázku přímo do asetů (*Assets*) v Unity a pak přetažením do okýnka vlevo vedle *Albedo* v materiálu.)







Ilustrace 4: Výběr barvy (albedo)

Po nastavení barvy ještě zbývá aplikovat materiál na objekt, nejjednodušší způsob je přetáhnout materiál z okna *Assets* přímo na model ve Scéně (*Scene*).







Ilustrace 5: Aplikace materiálu na objekt ve scéně

Seznam materiálů na modelu (*mesh*) vidíme v inspektoru v komponentě *Mesh Renderer* v seznamu *Materials*. Okno inspektoru zobrazuje detaily právě vybraného objektu. Na tomto místě je také možné materiál změnit nebo odstranit kliknutím na kolečko napravo vedle jeho názvu.



Ilustrace 6: Detail označeného objektu "Body", vpravo je možné vidět přidaný materiál "SpaceShipBody"

Reflexe

Jaké barvy jste použili? Jak barvy interagovaly s vámi nastaveným světlem? Myslíte, že vše můžeme ještě více vylepšit? Odhadněte čas nutný k případnému vylepšení systému.





2.2.4 Úprava tvaru scény v Unity – 3 hodiny

1. hodina – tvar scény

Podrobně rozpracovaný obsah

- V této hodině představíme (v napojení na pracovní listy) možnosti úprav tvaru virtuální scény dle libosti, uvážení a kreativních schopností každé skupiny, za pomocí jednoduchého 3D modelování, které Unity nabízí prostřednictvím stažitelného modulu ProBuilder. Dále také můžete, pokud nějaký znáte, použít externí modelovací program.
- 2. Pokud vám bude postup z této hodiny připadat příliš složitý, můžete scénu vyrobit stejným způsobem, jak jste skládali dohromady scénu v jedné z předchozích hodin v pracovním listě.
- 3. Zde bude nutné stáhnout si do Unity rozšíření ProBuilder, toto rozšíření je zdarma dostupné přímo z Unity editoru. Rozklikněte v menu tlačítko Window, poté Package Manager.

Win	dow	Help	
	Pane	els	>
	Next Window Previous Window		Ctrl+Tab
			Ctrl+Shift+Tab
	Layo	outs	>
	Unit	y IAP	>
	Coll	aborate	
	Asse	t Store	
	Pack	age Manager	

- 4. V nově otevřeném okně package manageru vpravo najděte vyhledávač a do něj vpište jméno použitého balíčku "ProBuilder", balíček poté nainstalujte kliknutím na tlačítko dole.
- 5. Pozor, pokud nebudete moci najít balíček, zkontrolujte si, zda jste připojeni k internetu a zda máte vlevo nahoře v nabídce Packages vybranou volbu Unity Registery.
- 6. Vaše uživatelské rozhraní může vypadat odlišně kvůli aktualizacím verzí Unity.



Reflexe

Jakou verzi Unity používáte? Způsobila instalace nových balíčků nějaké potíže?





2. hodina – skládání nebo modelování

Podrobně rozpracovaný obsah

Unity – Pro Builder

ProBuilder je nástroj Unity sloužící ke snadnému vytváření jednoduchých 3D objektů přímo ve scéně. Nový objekt editovatelný *ProBuilder*em vytvoříme následovně:



🛛 🕼 Pb_Object (Script)					💽 🗟 🕯				
Open ProBuilder									
Object Size (read only X	1	Y	1	Z	1				





V komponentách nově vzniklého *GameObject*u poté najdeme jednu s názvem *Pb_Object*, editaci zahájíme kliknutím na tlačítko *Open ProBuilder*. Scéna poté by měla vypadat přibližně takto:



Panel v levé části obrazovky nás zatím nebude zajímat. Na místo toho se soustředíme na menší panel v horní části. V tomto panelu jsou čtyři tlačítka ovlivňující způsoby výběru. První tlačítko odpovídá vybrání celého objektu, druhé vybírá body, třetí hrany a čtvrté stěny objektu.









Po vybrání čtvrtého tlačítka, představujícího stěnu, vybereme kliknutím přední stěnu objektu (pravděpodobně bude nejprve potřeba vybrat objekt, a proto bude potřeba dvojí kliknutí).

Jakmile budeme mít stěnu označenou, změníme její pozici ve směru červené šipky. Tímto dojde k protažení objektu ve vybraném směru.



Dále vybereme levou stranu a znovu ji posuneme, tentokrát ovšem budeme při posouvání držet tlačítko *shift* na klávesnici. Výsledek této úpravy pak bude vytvoření nového bloku napojeného na předchozí. Všimněte si nově vzniklé čáry rozdělující dva bloky přibližně uprostřed objektu.







Tento proces ještě jednou zopakujeme, až získáme blok rozdělený na tři části.







EVROPSKÁ UNIE Evropské strukturální a investiční fondy Operační program Výzkum, vývoj a vzdělávání



Pokračujeme předchozím způsobem jako dříve, vybereme stěnu ve středu objektu a pomocí změny pozice ji přesuneme podle červené šipky.

Když pak budeme měnit pozice levé a pravé stěny na čelní straně tohoto objektu, budeme opět držet klávesu *shift*. Výsledek by měl být přibližně takovýto:



Reflexe

Vymodelovali jste i nějaký vlastní objekt? Nakolik vás bavilo 3D modelování? Který tvar považujete za nejkomplikovanější?

3. hodina – jak to vypadá

Podrobně rozpracovaný obsah

- 1. Základní prvky pro úpravu grafické scény jsme předvedli, a proto je v tomto bloku věnován čas vaší vlastní práci. Stejně jako v závěru stavby robota zde přejdeme do pasivnější pozice a budeme vám pomáhat pouze tehdy, když to bude potřeba.
- V případě zájmu si můžete vyzkoušet ještě další pracovní listy týkající se Unity zaměřené na rozšíření práce s částicovými efekty a sérii zaměřenou na základní využití Unity skriptování (programovací nástroje).

Reflexe

Jakým způsobem jste vylepšili vaši scénu? Jaký objekt bylo nejtěžší vytvořit?





2.3 Vylepšování – 4 hodiny

2.3.1 Finalizace obou částí – 4 hodiny

1. hodina – blíží se finále

Podrobně rozpracovaný obsah

- 1. Cílem tohoto bloku je pak po všech stránkách dokončení vašeho projektu, ať už vám schází jakákoliv část.
- Tyto hodiny již neobsahují fixní předpis tematického rozsahu jako hodiny předchozí, v jejich časovém horizontu byste již měli dokončit své projekty a další chybějící věci a připravit se na turnaj.
- 3. Lektoři by vám měli být k dispozici, ať už co se týče posledních úprav na robotech, tak při práci v Unity.

2. hodina – poslední velké myšlenky

Podrobně rozpracovaný obsah

- 1. Jako v předchozí hodině, pokračujte v rozvíjení vašich děl, opravujte případné nedostatky. Nebojte se obrátit na lektory o pomoc.
- 2. Dejte si pozor na čas, zbývají vám poslední tři hodiny.

3. hodina – závěrečné úpravy

Podrobně rozpracovaný obsah

- 1. Nezapomínejte si vaši práci na počítači zálohovat, málo co zamrzí tak jako dílo, které vám zmizí těsně před dokončením.
- 2. Zamyslete se například také nad vizuální stránkou vašeho robota a jak působí na pohled.

4. hodina – konec, hotovo

Podrobně rozpracovaný obsah

- 1. Zbývá poslední hodina vaší kreativní práce, poté bude již následovat turnaj.
- 2. V případě, že vás ještě napadne nějaké vylepšení, dobře si rozmyslete, jestli jej stihnete celé implementovat. Může se vám také stát, že vám dojde čas uprostřed práce a váš robot skončí nedokončený!

Reflexe

Jste hrdí na svůj dokončený projekt? Co byste vylepšili? Co byste udělali znova?

2.4 Turnaj – 3 hodiny

2.4.1 Zkoušení výsledků a turnaj – 3 hodiny

1. hodina – připravte se na turnaj





Podrobně rozpracovaný obsah

- 1. V závěrečných lekcích budete soutěžit v samotném turnaji. S pravidly turnaje jste už byli seznámeni v předchozích hodinách. Po tomto krátkém úvodu se postupně vystřídáte a dostanete body za kvalitu vašich řešení v různých aspektech. V případě menšího počtu skupin se dá využít způsob turnaje "každý s každým", v případě větších skupin se pak dá sestavit klasický turnajový žebříček, ve kterém jdou vždy dvě skupiny proti sobě a vítěz se posouvá do dalšího kola, kde jde proti vítězům z ostatních párů, a tak dále až k vítězi posledního kola, a tedy i celého turnaje. Následující hodiny se od sebe obsahově nebudou lišit, protože se během nich utkáte v turnaji.
- Pokud budete mít lichý počet skupin v kole, je vhodnější nechat jednu skupinu se střetnout se dvěma různými protivníky než nechat ji přeskakovat kola, dokud nebude skupinek licho bez ní, ale je možné, že na takovou akci nebudete mít dostatek času.

Reflexe

Probíhá turnaj spíše v kompetetivním nebo přátelském duchu?

2. hodina – v přátelském duchu

Podrobně rozpracovaný obsah

Pokračujeme v turnaji. Ještě, než se do toho dáme, je potřeba připomenout několik důležitých zásad.

- 1. Vězte, že lektoři mají právo pravidla turnaje dle jejich uznání upravit.
- 2. Nezapomínejte, že v tomto programu je důležitější se něco naučit a bavit se, před zvítězením v turnaji!
- 3. Postupně se prostřídáte ve Virtuální realitě, pozor ať nezašlápnete nějakého robota.
- 4. Lektoři vám budou stopovat čas, dbejte jejich pokynů. Počítat body by měli nejen lektoři, ale také diváci, a nezapomeňte fandit!
- 5. Zbytečně neprotahujte aktivitu mezi jednotlivými koly turnaje, času je málo.
- 6. Není od věci si občas zkontrolovat, jestli jsou ovladače dobře zkalibrované.
- 7. Pozor, ať vám během kola nedojde baterka u robota a žádné podvádění s dálkovým ovládáním!

Reflexe

Jak dobře vám šlo ovládat virtuální realitu? Pomohli jste si designem vaší scény?

3. hodina – nejdůležitější co si odnesete, jsou znalosti

Podrobně rozpracovaný obsah

Opět pokračujeme v turnaji. Toto je poslední lekce našeho kroužku, dokončíme turnaj, vyhodnotíme a vyhlásíme vítěze. Diplomy pro vítěze a jejich slavnostní předávání je zcela jistě na místě, stejně jako drobné odměny pro všechny zúčastněné. Těšíte se?

1. Dejte si pozor, ať neztratíte nějaké součástky, na konci programu dobře prohledejte místnost, aby nezůstaly žádné neuklizené části.





2. Na závěr se ještě posaďte a pohovořte mezi týmy o tom, co byly nejlepší nápady, co jste třeba ještě měli v plánu a nestihli jste realizovat a podobně.

Reflexe

Líbily se ti technologie použité v tomto programu? Spolupracovalo se ti dobře ve skupině? Bavilo tě tvořit scénu a skládat robota?

3 Metodická část

Cíl celého programu a uvedení do tématu

Obecným cílem programu je nastínit účastníkům rozsah praktického využití programování ve skutečném světě. Obvyklé školní příklady mohou být občas příliš samoúčelné (např. zpracovávání spousty čísel bez zřejmé návaznosti na konkrétní praktický problém) nebo typově příliš podobné (např. jen konzolové aplikace – vyčtení ze souboru, zpracování dat a výpis výsledku). Žáci tak mohou pochybovat o jejich reálném významu a ztrácet motivaci zlepšovat se v programování.

V tomto programu budou mít naopak možnost aplikovat své dosavadní programovací schopnosti ve dvou dosti různých doménách, které svým způsobem definují celé spektrum použití programování v praxi: na jednom konci programování jednoduchých počítačů s nízkou výpočetní silou pro provádění činností ve fyzickém světě, na druhém vytváření náročných grafických aplikací při vysoké úrovní abstrakce od hardware.

Demonstrujeme, že prostřednictvím programování je možné předepisovat strojům, jak interagovat s reálným světem, a naopak přenášet uživatele do světa virtuálního, přičemž obojí lze navíc propojit do jednoho spolupracujícího celku. Díky tomuto širokému záběru má projekt potenciál zaujmout jak nadšence do mechanické techniky, tak i například milovníky počítačových her.

Dalším cílem je použít a vylepšit schopnosti účastníků k řešení problémů i mimo oblast programování (při konstrukci robota) a také prohloubit jejich zájem o méně obvyklé a interaktivnější formy uživatelského rozhraní, které může nabídnout virtuální realita a které se zřejmě budou v budoucnosti uplatňovat stále více.

Nezbytná teorie

Realizátor by měl ovládat základní koncepty programování. Měl by mít základní přehled o programovacích prostředích využívaných v rámci projektu (Lego Mindstorms a Unity) a měl by být schopen v nich samostatně vytvořit všechny programy, které budou mít za úkol vytvářet účastníci, případně vytvořit či upravit kostru projektu pro virtuální realitu v Unity podle konkrétních aspektů realizace tohoto programu (tj. podle aktuální verze programového vybavení, použitých hardwarových zařízení a podobně).

Je žádoucí, aby realizátorovi nezpůsobovalo obtíže používání virtuální reality, tj. aby nebyl náchylný k nevolnostem, neměl související zrakové či neurologické problémy apod. Cennou výhodou je též zkušenost se stavebnicí Lego Technic a související schopnost identifikovat nedostatky v konstrukci robota a navrhnout jejich řešení, pokud se účastníci ocitnou v nesnázích.

Schopnost práce v nástrojích pro 3D modelování a praktická znalost problematiky 3D tisku je další výhodou, která umožňuje vyhotovování speciálních konstrukčních prvků pro robota.





Vstupní předpoklady programu

Tento program je zaměřen spíše na nadanější žáky, nadšené do technologií, počítačů, robotů a podobně, s čímž se váže jistý zájem o získávání nových znalostí a objevování nových oborů. Vhodné jsou dále počáteční znalosti, není nutné, aby žák měl všechny tyto předpoklady, nicméně je velmi vhodné, aby měl alespoň některé z nich.

- 1. Z důvodu silného technického základu teorie a praxe tohoto kurzu budou moci žáci se základními programovacími dovednostmi snáze přijímat nové znalosti zaměřené na oblast robotiky a virtuální reality.
- 2. Vzhledem k velkému rozsahu oborů, které spadají do tohoto programu, je velmi vhodné, aby žáci disponovali jistou dávkou nadšení, trpělivosti a schopnosti učit se.
- 3. V rámci práce na úkolech bude nutné prokázat schopnosti jak samostatné, tak i skupinové práce.
- 4. Z důvodů vyšších obecných znalostních požadavků se předpokládá taktéž zaměření na vyšší věkovou skupinu účastníků. Primárně střední školy a vyšší stupně víceletých gymnázií.

Stejně tak jako na účastníky nese tento kurz zvýšené požadavky na technické znalosti a dovednosti realizátorů a lektorů.

- 1. Bez základních znalostí a orientace v problematice nebudou schopni připravit požadované prostředí a vést jednotlivé části kurzu.
- 2. V rámci kurzu bude nutná jak hromadná, tak individuální forma výuky, a s tím svázané schopnosti poskytnout radu, či žáky navést k řešení, avšak nevodit za ručičku, aby organicky vznikala nová zajímavá originální řešení.
- 3. Bude také zapotřebí dopředu prostudovat materiály a vyzkoušet spojený software.

S kurzem jsou také spojeny nároky na vnější prostředky, které program vyžaduje k hladkému běhu. Těmi jsou například možnost pravidelného setkávání v delších časových úsecích, což může způsobit problémy jak na straně infrastruktury školy, tak na straně volného času žáků.

Dalším požadavkem je přítomnost a funkčnost vybavení potřebného pro program. (PC, Vive, Mindstorms, tablet, dostatek prostoru a další.)

Přínos neformálního vzdělávání

Program využívá pokročilého technologického vybavení k účinnějšímu propojení teoretické znalosti s aplikovatelnými dovednostmi a také usnadňuje cestu k jejich nabytí. Současně představuje významnou podporu v oblasti motivace. Tím se mění potřebné postoje k pochopení významu moderní techniky při utváření současného světa i možností, které každý z nás má, aby se do toho procesu nějak zapojil.

Dalším rozměrem přínosu neformálního vzdělávání je uplatnění formativního hodnocení, které průběžně provází účastníky, opírá se v prvé řadě o sebehodnocení a koučovací přístup lektorů. Program probíhá za příznivé, podporující atmosféry, zahrnuje různé formy spolupráce.

Možné komplikace a problémy, řešení nestandardních situací

Úspěšné opakovatelnosti realizace projektu neprospívá fakt, že trh s elektronikou je dosti dynamický a nabídka zařízení se poměrně často mění. Program tak v době návrhu počítal s využitím headsetu pro virtuální realitu HTC Vive, který byl v dané době zdaleka nejdostupnějším technickým řešením, které





podporovalo práci s externími trackery. V době dokončování ověřování programu však již tento headset není v produkci. Místo něj lze použít headsety HTC Vive Pro nebo Valve Index, které nabízejí obdobné technické možnosti včetně využití Vive Trackeru (resp. jeho modernizované verze s podporou sledování na bázi systému Lighthouse 2.0). Lze předpokládat, že v budoucnosti se bude dostupnost a cena takovýchto zařízení dále vyvíjet, a je obtížné odhadnout, do jaké míry a v jaké cenové kategorii bude podpora trackerů v budoucnosti rozšířená.

Značná část budoucích řešení však bude téměř jistě podporovat alespoň sledování pozice a rotace dvou ovladačů, přičemž v takové situaci je možné případnou nedostupnost externího trackeru kompenzovat připevněním jednoho z ovladačů na robota a ovládáním VR aplikace pomocí jediného ovladače. To s sebou však nese nutnost úprav projektu v Unity, např. luk a šípy by bylo nutné nahradit nějakým vrhacím projektilem. Může být taktéž potřeba upravit 3D model držáku trackeru pro robota podle konkrétního zařízení.

Obdobnou komplikací je proměnlivost softwarového vybavení – formou aktualizací se zásadně mění jak Unity a jednotlivá VR API (např. SteamVR/OpenVR), tak i samotný operační systém atd., přičemž ne všechny tyto komponenty mohou být v budoucnu dostupné v té (starší) verzi, na které proběhlo ověření programu, a budoucí verze nemusí nutně fungovat společně bez problémů. Je tedy nezbytné, aby byl realizátor dostatečně technicky zdatný a schopný samostatně zprovoznit nutnou kostru projektu v Unity a další potřebné součásti s přihlédnutím k těmto změnám.

Program v Unity

Je pravděpodobné, že zprovoznění a napojení virtuální reality a Unity bude vyžadovat jistou dávku adaptace, technologie se rychle mění a již v rámci psaní této metodiky vzniklo několik odlišností, které bylo třeba upravit. Předchystané postupy a scéna nebudou nutně na první pokus fungovat, protože se již mohly stát zastaralými. Obecné postupy, jak pracovat v Unity, by měly vydržet aktuální ještě řadu let, takže většina adaptace oproti programu bude ležet na bedrech lektorů.

V případě nefunkčnosti doporučuji následovný, víceméně univerzální postup (platný, dokud bude používaná stejná knihovna).

- 1. Stáhněte si nejnovější stabilní verzi Unity.
- 2. Poté v Unity knihovnu SteamVR, kterou naleznete na této adrese nebo ji vyhledáte na internetu:
- 3. Po instalaci knihovny a zapojení hardwaru VR najdete v projektu složku příkladů a tam najdete podsložku Examples.
- 4. Ve složce Examples budou různé scény, některá z nich by měla typově odpovídat cílové scéně.
- 5. Nakonec je třeba doplnit objekt ve scéně, který bude sledovat pozici VR trackeru, toto může být netriviální a je náročné předvídat, jak se workflow změní do budoucna.

Další informace

Ukázkové projekty s roboty Lego Mindstorms, návody: ev3lessons.com

Ukázkové projekty, návody, výuková videa vysvětlující jednotlivé koncepty v Unity: https://learn.unity.com/tutorials

Problematika 3D uživatelských rozhraní: 3D User Interfaces: Theory and Practice (Bowman et al., 2017)





Technické pokyny k vytváření specifických vstupních zařízení pro VR s použitím Vive Trackeru: https://dl.vive.com/Tracker/Guideline/HTC_Vive_Tracker(2018)_Developer+Guidelines_v1.0.pdf

3.1 Roboti LEGO Mindstorms – 21 hodin

V prvním celku se žáci naučí pracovat s roboty Lego Mindstorms. Budou schopni sestavit robota a budou schopni aplikovat základní prostředky programovacího prostředí. Pro programování budou používat stolní počítače a také aplikace v tabletech (APLIKACE EV3 PROGRAMMER).

Pro stavbu robota budou využívat základní i rozšířenou sadu EV3. Pro vytvoření speciálních kostek a součástek potřebných pro připevnění trackeru použijí 3D tiskárnu. Tvorba 3D modelů v grafickém programu není předmětem kroužku. Grafický model je součástí připravených dat a účastníci kroužku dostanou k dispozici již hotový výsledek a budou provádět jednoduché úpravy pro potřeby svého robota a následné tisky. Vzhledem k časové náročnosti bude na průběh tisku dohlížet pomocný lektor i v době mimo konání kroužku.

Součástí stavebnice je i několik materiálů dodaných přímo od výrobců Lego, je vhodné mít tyto materiály dostupné v nějaké formě během aktivity. Materiály je možné mít jak v tištěné, tak v digitální podobě.

Příručku je možné stáhnout na tomto odkazu:

Zde je celá série návodů pro stavbu různých specifických robotů, (použijte dle uvážení):

3.1.1 Úvod do LEGO Mindstorms – 3 hodiny

Forma a bližší popis realizace

Hromadná forma. Představení učitelů a vzdělávacího programu samotného.

Metody

Prezentace a hromadná debata.

Pomůcky

LEGO Mindstorms, HTC Vive s počítačem, dataprojektor

1. hodina

Žáci se dozví o cíli tohoto programu a budou jim představeni učitelé a jejich role. Tím je sestavení a naprogramování robota LEGO Mindstorms a vytvoření aplikace pro virtuální realitu v enginu Unity. Uživatel se v ní bude snažit ve virtuálním prostředí sestřelovat cíl lukem ovládaným pomocí ovladačů HTC Vive. Terčem bude ve skutečné realitě robot LEGO Mindstorms, na němž bude umístěn Vive tracker pro sledování jeho skutečné pozice a jejího promítnutí do virtuální reality.

Na tvorbě scény se budou podílet všichni žáci a programování robota bude úkolem ideálně dvoučlenných žákovských týmů. Tyto týmy budou na konci kroužku postaveny proti sobě, kdy jeden bude mít za úkol v časovém limitu co nejvíckrát zasáhnout terč na robotovi ovládaném programem druhého týmu a naopak.

Důležitou částí úvodu programu je získání základního přehledu o stavebnici Lego Mindstorms. Vzhledem k tomu, že v následujících několika hodinách budete postupně rozšiřovat vaše znalosti





ohledně možností této stavebnice, nebojte se vlastního experimentování. To vám pomůže k získání vlastního pochopení jednotlivých dílčích mechanismů stavebnice.

Samotná analýza jednotlivých dílů a jejich případy využití jsou pak aspektem celého průběhu této části programu.

Definice robota:

Definice robotiky:

Žáci pak budou seznámeni s řádem učebny, základními bezpečnostními pravidly, organizační strukturou a časovým plánem kroužku. Samotná upozornění a upřesnění postupů manipulace s vybavením budou provedeny až v momentě, kdy bude dané vybavení poprvé využito v programu. Důležitější pak bude bezpečnost týkající se virtuální reality, která vám bude lektory představena, až přijde VR na řadu v programu. Bezpečnost samotné lego stavebnice je obdobná jako každé jiné stavebnice. Studenty ve věkové kategorii, na kterou je tento program mířen, by nemělo být nutné upozorňovat, aby omylem nevdechnuly malé části a podobně. Nicméně by měli zachovávat opatrnost, delší kusy se mohou zlomit, pokud na ně někdo šlápne, a motory a senzory ve stavebnici nejsou odolné proti nárazu.

Je vhodné napsat na všechny krabice číslo zepředu a navrchu, u všech krabic na stejné místo, protože hledání krabice s nějakým číslem je poměrně častý proces, obzvláště na začátcích hodin. Dále je vhodné číslo krabice zahrnout i do názvu příkazového bloku robota, tento krok se dá případně doplnit až v pozdější hodině, kdy dojde na práci s tímto dílem.

2. hodina

Forma a bližší popis realizace

Skupinová forma. Studenti se vzájemně seznámí a poté utvoří dvoučlenné skupiny.

Metody

Představování a hromadná debata

Pomůcky

LEGO Mindstorms, HTC Vive s počítačem

Po představení cílů projektu může následovat vzájemné seznámení a představení žáků, což bude první částí této hodiny. Druhou částí potom bude sestavení soutěžních týmů. Týmy by ideálně měly být dvoučlenné. Samostatná práce je sice možná, ale rozhodně není preferovaná.

3. hodina

Forma a bližší popis realizace

Hromadná forma. Představení robotů LEGO Mindstorms a jejich možností.

Metody

Aktivizační metody, předvádění lektorem.




Pomůcky

LEGO Mindstorms, HTC Vive s počítačem

Cílem této hodiny je nadchnout žáky pro práci s roboty. Ukázat jim, co je možné s nimi dělat a jaké mají limity. Žáci se budou moci podívat na skutečného robota plnící naprogramovaný úkol, nebo alespoň videa zaznamenávající vykonávání složitých programů. Příkladem takovýchto úkolů je například navigace v reálném prostoru, sběr a třídění objektů nebo skládání Rubikovy kostky.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Jak vás téma robotů zaujalo? Inspirovalo vás dnešní setkání? Chtěli byste se zabývat moderními technologiemi i v budoucnu? Pracujete rádi ve skupinách nebo preferujete individuální činnost? Jaká byla vaše dnešní spolupráce? Nakolik jste využili manuál k popisu dílů stavebnice? Je manuál zpracovaný přehledně? Byl pro vás pochopitelný? Podařilo se vám navrhnout vlastní funkční řešení? Složili jste úspěšně všechny části robota? Co vás nejvíce překvapilo? Zasekli jste se na nějaké části stavby? Ohodnoťte na škále 1–5, nakolik bylo pro vás obtížné přidat první pohyb.





3.1.2 Skládání podvozku robota – 3 hodiny

1. hodina

Forma a bližší popis realizace

Skupinová forma. Získávání praktických zkušeností a znalostí se stavebnicí a metod základní stavby robota – studenti si vyzkouší práci se stavebnicí LEGO Technic a započnou konstrukci podvozku robota.

Metody

Prezentace, samostatná práce, konzultace a dotazy

Pomůcky

LEGO Mindstorms

Žákům budou představeny základní typy dílků stavebnice LEGO Technic a možnosti jejich spojování, přičemž si je samostatně vyzkouší. Jakmile budou schopni tyto principy aplikovat v dostatečné míře, mohou začít skládat základ budoucího robota – podvozek s pohonem (motory a kola nebo pásy), který ponese "programovatelnou kostičku" LEGO Mindstorms, jako jednu z možností usnadnit žákům tvorbu představ. Co se možností stavebnice týče, je také představení již sestavených robotů, ideálně i s předchystaným programem řídícím daného robota. Tento robot může být žáků z předchozího běhu programu nebo sestaven lektory.

Není nutné, aby programování ukázkového robota přesně odpovídalo cílům tohoto programu, a to obzvláště v bodech, kde program závisí na vlastním rozhodnutí žáků. V takovém případě by mohlo předvedení takového řešení omezit snahu žáků pokusit se vymyslet jejich vlastní. Začátečníkům je doporučeno postupovat dle instrukcí pro skládání podvozku některého ze základních robotů, vhodný je zejm. Driving Base (Education) nebo Track3r či Gripp3r (Standard); účastníci disponující výraznějšími předchozími zkušenostmi se systémem LEGO Technic mohou navrhnout a realizovat vlastní řešení podvozku robota. V každém případě je vhodné aby:

- Podvozek umožňoval řízený pohyb robota pomocí motorů, typicky jízdu s možností zatáčení.

- Bylo možné snadno vyjmout "programovatelnou kostičku" nebo byl dostatečný přístup k její zadní straně za účelem výměny baterií.

- Byl umožněn pohodlný přístup ke konektorům "programovatelné kostičky".

- Vytvořený základ robota konstrukčně nabízel dostatek možností pro smysluplné uchycení dalších součástí, zejm. senzorů a VR trackeru.

Učitelé by měli průběžně sledovat práci jednotlivých skupin, nabízet jim rady nebo pomoc, případně upozorňovat na možné konstrukční nedostatky zvoleného řešení.

2. hodina

Forma a bližší popis realizace

Skupinová forma. Dokončení základu robota – studenti pokračují v konstrukci podvozku robota.

Metody

Samostatná práce, konzultace a dotazy





Pomůcky

LEGO Mindstorms

Žáci pokračují v započaté práci a dokončují základ budoucího robota. Stále přitom mají možnost klást otázky učitelům kroužku a využívat jejich asistence. K ověření rámcové funkčnosti pohybového mechanismu je možné použít elementární programy vestavěné v "programovatelné kostičce", příkazy z dálkového ovládání apod. Vlastní programování pohybu a chování robota bude předmětem až následujícího tematického bloku. Po této hodině by tedy všechny skupiny měly mít postavený funkční podvozek umožňující pohyb robota a splňující podmínky uvedené v popisu předchozí hodiny.

3. hodina

Forma a bližší popis realizace

Hromadná forma. Teoretické představení programovacího prostředí.

Metody

Prezentace, ukázky

Pomůcky

LEGO Mindstorms, počítače nebo tablety s potřebným SW, dataprojektor

Učitel bude procházet a komentovat základní prvky vývojového prostředí LEGO Mindstorms, přičemž grafický výstup (obsah obrazovky) bude žákům promítán. Tím bude představen způsob vytvoření, ukládání a načítání projektu, základní nabídka pro výběr programových bloků a grafické rozhraní editoru pro vytváření programu prostřednictvím propojování funkčních bloků. Není zde třeba procházet konkrétní vlastnosti jednotlivých bloků apod., toto představení slouží spíše pro vytvoření obecné představy o práci s vývojovým prostředím, tak aby některé aspekty stačilo v příští hodině pouze připomenout a aby prostředí žákům při následné práci připadalo povědomé a známé.

Žáci mohou tyto kroky následovat na vlastních počítačích a trénovat tak vyhledávání funkčních bloků a jejich vzájemné napojování, což jsou dovednosti, které budou potřebovat v následujícím tématickém bloku při programování chování robota. Většina studentů nebude mít na robotu žádné vstupní prvky (čidla, senzory atd.), které zatím nebyly v programu představeny, proto bude jejich program obsahovat primárně příkazy pro robota, které budou řídit jeho pohyb, vydávat zvuk a podobně.

Studenti si vyzkoušení manipulaci s rozhraním programovacího prostředí, základní skládání funkčních bloků, jejich propojování a podobně. Zde jsou také představeny jednotlivé typy řídících prvků, ačkoliv je jejich potřeba zatím omezená jen na ty základní.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Baví vás experimentování? Dělá váš robot to, co má? Splňuje vaše stavba vaše očekávání? Zvládl váš robot požadované úkoly? Skládá se vám se stavebnicí dobře? Povedlo se vám na robotovi vytvořit dostatečně široký prostor pro připojení VR trackeru? Je váš programovací blok na robotovi dostatečně přístupný? Fungovalo vám propojení vašeho robota s mobilem dobře?





3.1.3 Praktická ukázka programování robota

1. hodina

Forma a bližší popis realizace

Skupinová forma. Analýza vztahu mezi programovými bloky a ovládáním motorů způsobem nahrávání programu do robota a jeho spuštěním.

Metody

Samostatná práce, průběžné testování, konzultace a dotazy

Pomůcky

LEGO Mindstorms, počítače s potřebným SW

Učitel vytvoří elementární program skládající se z jediného bloku pro ovládání motorů, přičemž postup je promítán žákům, kteří jej zároveň provádějí. Nejprve je oběma motorům po stranách podvozku nastavena stejná rychlost, což způsobí jízdu v přímém směru. Tento program je následně nahrán do robota a spuštěn. Poté, co žáci úspěšně provedou tyto předvedené kroky a robota tak rozhýbají, dostanou za úkol změnit program tak, aby robot objel jeden celý kruh. Budou tak muset nastavit každému z motorů jinou rychlost a zároveň vhodně zvolit čas/rozsah pohybu, aby robot vykonal právě jeden kruh a zastavil přibližně na místě, odkud původně vyjel. Pokud se to některým skupinám povede rychleji než jiným, mohou program doplnit o zvukové či obrazové efekty.

Tato hodina je výrazněji zaměřena na experimentování s programováním robota, může zde však dojít k úpravám a případným přestavbám v těch případech, kdy studenti zjistí, že jim jejich nynější stavba nevyhovuje či nefunguje. V průběhu programu je potřeba dosáhnout jistého kompromisu mezi vylepšováním stávajícího řešení a postupem k novým nástrojům a technikám.

2. hodina

Forma a bližší popis realizace

Skupinová forma. Tvorba složitějšího programu pro pohyb s využitím cyklů.

Metody

Samostatná práce, průběžné testování, konzultace a dotazy

Pomůcky

LEGO Mindstorms, počítače s potřebným SW

V této hodině žáci vytvoří mírně složitější variantu programu z hodiny předchozí – robot se nebude pohybovat v kruhu, ale po trajektorii tvaru čtverce. Na tomto programu lze demonstrovat použití počítaných cyklů (cyklů se známým počtem opakování) – tělem cyklu je ujetí určité vzdálenosti v přímém směru a následné otočení o 90 ° na místě (pohyb náprav opačným směrem), přičemž tuto akci je třeba zopakovat čtyřikrát.

Program lze následně opět doplnit o rozličné efekty, přičemž i tyto mohou být prováděny v cyklu (např. robot může s využitím počítaného cyklu a funkčního bloku pro čekání třikrát zablikat).





3. hodina

Forma a bližší popis realizace

Hromadná forma. Teoretické představení jednotlivých senzorů sloužících jako vstup robota.

Metody

Prezentace, ukázky, konzultace a dotazy

Pomůcky

LEGO Mindstorms, počítače s potřebným SW

V této hodině budou představeny senzory, které jsou dostupné pro použití se systémem LEGO Mindstorms, slouží jako vstupní zařízení robota a umožňují mu tak získávat informace o jeho okolí. K dispozici jsou senzor na měření vzdálenosti, detektor barvy a intenzity světla a tlačítko; ke zjišťování informací o rozsahu či rychlosti rotačního pohybu lze navíc využít velké motory.

Žákům budou nastíněny možnosti, které využití těchto senzorů otevírá, a budou předvedeny příklady jejich použití. Tato hodina tak slouží jako motivace žáků k následujícímu tematickému bloku, který jim práci se senzory představí z praktické stránky. V této hodině je také možné předvést konkrétní příklad, který využívá senzoru ve svém fungování. Můžete mít nějakého předpřipraveného robota, na kterém fungování ukážete, tento robot může být vyroben studenty z předchozího běhu programu nebo lektory programu, jeho vlastnosti budou pak samozřejmě záviset na tom, jak vysoko bude postavena cílová laťka tohoto programu.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Je pro vás prostředí přehledné? Dělá váš robot to, co po něm vyžadujete? Jede váš robot dopředu rovně a zatáčí správně? Zkoušeli jste nastavit různou rychlost a sílu motorů? Jak se vám dařila práce v programovacím prostředí? Je prostředí dostatečně přehledné? Zamyslete se, k čemu by získané hodnoty mohl robot ještě využít?





3.1.4 Dokončení základů programování robota

1. hodina

Forma a bližší popis realizace

Skupinová forma. Přidání senzoru barvy na robota.

Metody

Prezentace, ukázky, konzultace a dotazy

Pomůcky

LEGO Mindstorms

V této hodině je žákům připomenut a podrobněji představen senzor pro rozpoznání barvy či intenzity světla. Tento senzor studenti vhodně zakomponují do stávající konstrukce robota tak, aby bylo možné zjišťovat barvu podložky pod robotem, nejvhodnější umístění je typicky na předozadní ose robota (tj. v polovině šířky robota) před osou jeho otáčení při otáčení na místě (tj. před nápravami) a těsně nad podložkou. Pomocí výpisu stavu senzoru ve vývojovém prostředí následně ověří jeho korektní zapojení a funkčnost. Zároveň bude přichystána testovací plocha s dráhou tvořenou barevnou linkou, která bude využívána ve zbývající části tohoto tematického bloku i v blocích následujících.

Žáci sami chápou, že jejich schopnosti a znalosti se v tomto ohledu prohlubují. Rozhodli se se účastnit tohoto programu a tím projevují zájem o informatiku jako obor. Je velmi pravděpodobné, že tito žáci nějaké základní znalosti měli již před tímto programem, a proto není vhodné podceňovat jejich schopnost se učit nejen ve škole, ale i sami od sebe. Není tedy nutné jim "prodávat" informace z tohoto programu, protože žáci mají svoji vlastní motivaci k uchování těchto informací. Naopak, snaha jim je vnutit může vést k externalizaci této motivace a její následné ztrátě.

2. hodina

Forma a bližší popis realizace

Skupinová forma. Rozbor vlastností jednotlivých senzorů a jejich kombinování s podmíněným vykonáváním bloků programu – detekce čáry na podložce.

Metody

Prezentace, ukázky, samostatná práce, průběžné testování, konzultace a dotazy

Pomůcky

LEGO Mindstorms, počítače s potřebným SW

Žáci se naučí aplikovat získané znalosti o vstupních hodnotách senzorů jako vstupu pro vyhodnocování podmínky při větvení programu. Budou jim představeny řídicí bloky pro větvení a pro podmíněný cyklus, které mohou testovat hodnotu ze senzoru. Jejich úkolem pak bude vytvořit program, díky kterému pojede robot rovně, dokud nenarazí na černou čáru.

Možností, jak zadaný úkol zpracovat, je několik: lze např. o kousek popojet a poté otestovat, zda senzor snímá černou barvu – pokud ne, robot znovu popojede a tak dále; alternativně lze zapnout motory ("na neurčito"), poté v cyklu zjišťovat barvu, pokud je barva černá, cyklus opustit, a následně motory vypnout.





Druhý z popsaných způsobů je obvykle vhodnější, neboť jízda robota je plynulá, nedochází k jeho cukání způsobenému jednotlivými "kroky" motorů mezi měřeními. Zde je také vhodné žákům vysvětlit parametr zastavování u bloku pro ovládání motorů a blok pro čekání do splnění podmínky.

3. hodina

Forma a bližší popis realizace

Skupinová forma. Samostatná tvorba komplexního programu pro pohyb – sledování čáry na podložce.

Metody

Samostatná práce, průběžné testování, konzultace a dotazy

Pomůcky

LEGO Mindstorms, počítače s potřebným SW

Žáci dostanou za úkol naprogramovat robota tak, aby našel a sledoval na podložce dráhu – okruh tvořený dvojicí barevných čar, např. černou a červenou, nakreslených nebo nalepených těsně vedle sebe. Robot přitom vždy začíná uvnitř okruhu. Pro splnění úkolu této hodiny musí žáci využít všechny schopnosti, které se dosud naučili. Musí umět robota rozpohybovat, číst data z detektoru barvy, správně je interpretovat a podle nich upravit pohyb robota. To vše za využití cyklů a podmínek, které se už naučili. Po zvládnutí toho úkolu by již žáci neměli mít problém samostatně vytvořit konkurenceschopný program předepisující chování robota za účelem účasti v turnaji konaném na konci kroužku.

Princip vhodného řešení je následující (předpokládejme, že podložka je bílá, červená čára je na vnitřní straně okruhu, černá je vně, chceme, aby robot jel po směru hodinových ručiček a senzor barvy je umístěn na předozadní ose robota před osou jeho otáčení při otáčení na místě):

1. Robot jede rovně, dokud nenarazí na černou čáru (jako v případě úkolu řešeného v minulé hodině).

2. Robot se otáčí doprava, dokud nenarazí na červenou barvu (resp. dokud je na černé barvě).

3. Robot jede rovně, dokud je na červené barvě (resp. dokud nenarazí na černou nebo bílou barvu).

4a. Pokud je na bílé barvě, otáčí se doleva (resp. dokud nenarazí na červenou barvu).

4b. Pokud je na černé barvě, viz 2.

5. Viz 3.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Myslíte, že roboti vnímají barvy lépe než lidé? Jak se naše vnímání liší? Kolikrát jste během této hodiny otestovali své řešení? V čem vám to pomohlo? Zamyslete se a řekněte, v čem vám vaše nynější stavba robota zkomplikovala programování.





3.1.5 Praktická práce s dalšími senzory

1. hodina

Forma a bližší popis realizace

Hromadná forma. Představení senzoru s tlačítkem.

Metody

Prezentace, ukázky, konzultace a dotazy

Pomůcky

LEGO Mindstorms

V tomto tematickém bloku budou představeny možnosti rozšíření stávající funkcionality obohacující roboty o další informace o jejich okolí. Tato rozšíření nejsou nezbytně nutná pro účast v závěrečném turnaji, nicméně jejich použití je dovoleno a mohou poskytnout značnou výhodu.

Senzor s tlačítkem lze použít především pro detekci kolize s překážkou nebo jako "uživatelské rozhraní", např. pro spuštění odezvy robota v reakci na jeho zmáčknutí, byť za tímto účelem je možné programově zjišťovat též stav navigačních tlačítek na "programovatelné kostičce". Žáci si mohou tento senzor připevnit na robota a samostatně si vyzkoušet, jak se s ním pracuje v programu.

2. hodina

Forma a bližší popis realizace

Hromadná forma. Představení senzoru vzdálenosti.

Metody

Prezentace, ukázky, konzultace a dotazy

Pomůcky

LEGO Mindstorms

Senzor vzdálenosti lze použít dokonce k prevenci kolize s překážkou. Je schopen předávat data např. v podobě počtu centimetrů od detekovaného objektu. Při zjištění blízké překážky tak může robot např. začít couvat nebo se otočit. Žáci si opět mohou samostatně vyzkoušet práci s tímto senzorem.

3. hodina

Forma a bližší popis realizace Hromadná forma. Využití senzoru na měření vzdálenosti.

Metody

Prezentace, ukázky, konzultace a dotazy

Pomůcky

LEGO Mindstorms





80

Žáci jsou v této hodině motivování k vymýšlení a případné realizaci vlastních rozšíření zvyšující kvalitu jejich řešení. V předchozích hodinách a také v této byly postupně představovány různé části robota a funkční bloky programovacího prostředí. Nyní by měli mít studenti k dispozici dostatek nástrojů k vymyšlení a implementaci nějakého vlastního řešení. Zde je opět třeba mít na paměti, že není vhodné bránit studentům v jejich kreativitě, ale zároveň je třeba jim pomoct nebo poradit, pokud se na něčem zaseknou.

Funkční program využívající všech senzorů bude tedy postupně sestavován v průběhu této a dalších hodin. Efektivní využití obzvláště vzdálenostního senzoru bude vyžadovat testování. Je možné, že některé skupiny se rozhodnou senzor na měření vzdálenosti nakonec vůbec nepoužít, to je také v pořádku.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Co by se se vám hodilo k tomu, abyste mohli lépe a snáze testovat? Přinesly vám nové možnosti i nové nápady? Který senzor vám přišel nejužitečnější? Který vám způsobil nejvíce potíží?

3.1.6 Zahájení prací na vlastním technickém řešení – 3 hodiny

1. hodina

Forma a bližší popis realizace

Skupinová forma. Detailní zadání úkolu, návrh chování a konstrukce vlastního soutěžního robota, pro zájemce představení pokročilých možností programování robotů.

Metody

Prezentace, samostatná práce

Pomůcky

LEGO Mindstorms, počítač s projektorem

Žákům budou představena pravidla závěrečného turnaje, která musí každý zúčastněný robot respektovat:

- Robot ponese Vive tracker, jehož pozice bude reprezentovat pozici cíle ve virtuální realitě. Na tento cíl bude člen konkurenčního týmu střílet virtuální šípy. Bude sledován počet zásahů za jednu hru (hra trvá pevnou dobu). Tým tvůrců robota, který obdrží nejméně zásahů, vyhrává. Úkolem robota tedy bude vyhýbat se virtuálním šípům, tj. pohybovat se (resp. pohybovat trackerem) způsobem, který činí zásah co možná nejobtížnějším rychle, nepředvídatelně atd.
- Tracker musí být umístěn na dobře viditelném, odkrytém místě. Robot nesmí v rámci své funkcionality záměrně zakrývat tracker žádným neprůhledným objektem.
- Vyjede-li robot během hry celým objemem z vytyčené oblasti (černá čára na bílé podložce), bude tým diskvalifikován.





- V oblasti pro pohyb robota budou umístěny překážky, kterým by se měl robot vyhnout. Při vytlačení překážky mimo vytyčenou oblast bude týmu udělena penalizace v podobě přičtení zásahu navíc.
- Povoleno je využití všech dostupných senzorů, avšak lze používat pouze jedno tlačítko a jeden kanál na dálkovém ovladači robota.
- Žádný ze žáků nesmí během hry robota nijak fyzicky posouvat nebo mu naopak bránit v pohybu.
- Startovní pozici robota i překážek určuje tým, který bude střílet, robot i překážky však musí být celým objemem uvnitř vytyčené plochy a robot musí být umístěn podvozkem na podložku (tj. nesmí být položen "vzhůru nohama" nebo na některou z překážek). Robot také nesmí být překážkami zcela obklopen.

Žáci následně započnou návrh chování a konstrukce vlastního robota, kterého na konci kroužku do soutěže nominují.

Zkušenějším programátorům bude představena možnost programování robota v některém z vyšších programovacích jazyků. Tuto možnost mohou využít i pro programování soutěžního robota. Průzkum programového rozhraní robota je ponechán na zájemcích ve formě samostudia, pouze s možností konzultací případných nesnází s učiteli kroužku.

2. – 3. hodina

Vzhledem k podobnému charakteru vývoje a organizaci následujících dvou hodin je jejich metodická část popsána společně.

Forma a bližší popis realizace

Skupinová forma. Práce na vlastním řešení robota.

Metody Samostatná práce, dotazy

Pomůcky

LEGO Mindstorms

Žáci realizují konstrukční i programové řešení robota na základě návrhu požadované funkcionality z minulé hodiny. Mohou využít stávající konstrukci podvozku a libovolně ji vylepšit, nebo naopak změnit celou koncepci a začít zcela od začátku, každopádně by však měli použít pouze součástky z vlastní sady.

Konstrukční řešení musí podporovat umístění Vive trackeru, tj. musí být k dispozici místo pro jeho uchycení na držáku, který bude žákům ukázán, a takové umístění by nemělo způsobit převrácení robota jeho převážením apod. Je proto vhodné, aby žáci tento aspekt konstrukce dostatečně otestovali, a případně upravili např. rychlosti pohybu či jiné parametry tak, aby s patřičným rozložením hmotnosti počítali.

Lektoři prochází mezi skupinami a kontrolují průběh sestavování robota. Individuálně vedou a radí, vše doplňují dalšími otázkami.





Problémy studentů mohou být způsobeny netriviálně špatně propojenými součástkami v robotovi, není v silách tohoto textu popsat ani zlomek problémů, které potenciálně mohou nastat, je na lektorech, aby v takových případech studentům poradili. V nejhorším případě mohou přestavět robota v nějakou jednodušší variantu.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Máte již vymyšlenou týmovou taktiku? Co by mohly být přednosti vašeho robota? Máte všechny komponenty, které potřebujete? Zkontrolovali jste správné zapojení všech součástek?

3.1.7 Dokončování vlastního robota a prezentace VR – 3 hodiny

1. – 3. hodina

Vzhledem k podobnému charakteru vývoje a organizace následujících dvou hodin je jejich metodická část popsána společně.

Forma a bližší popis realizace

Hromadná forma. Detailní zadání úkolu, návrh chování a konstrukce vlastního soutěžního robota, pro zájemce představení pokročilých možností programování robotů.

Metody

Prezentace, samostatná práce

Pomůcky

LEGO Mindstorms, HTC Vive, počítač pro VR

Žáci pokračují ve vytváření vlastních soutěžních robotů a jejich průběžném testování, souběžně s tím je jednotlivým účastníkům kroužku umožněno vyzkoušet si používání virtuální reality na aplikaci The Lab (dostupné zdarma na platformě Steam), která obsahuje mj. minihru založenou na střílení z virtuálního luku, tj. principem podobnou aplikaci, kterou budou žáci vytvářet v následujících tematických blocích a ve které se budou snažit uspět při závěrečném turnaji. Jednotliví žáci se budou střídat po cca 10 až 15 minutách.

Je vhodné zajistit samostatný dozor u VR headsetu, který bude dohlížet na dodržování předepsané doby, správné zacházení s headsetem a pomáhat při nasazování a sundávání (např. podávat ovladače a asistovat žákům s brýlemi), zatímco učitelé kroužku budou k dispozici pro zodpovídání dotazů žáků věnujících se skládání a programování robotů. Pokud by někteří žáci v rámci tohoto bloku nestačili zcela dokončit práci na svém robotovi, budou mít možnost vrátit se k ní ještě v budoucnu.

Tento výukový blok nemá hodiny vyhraněny na jednotlivé aktivity, VR headset může mít na hlavě pouze jeden člověk najednou a dokončování robotů nebudou všichni studenti stíhat ve stejnou dobu Časovou organizaci tohoto bloku tedy bude třeba řešit podle aktuálních potřeb a často individuálně.





Podstatnou částí je zde seznámení se s technologií, ke které většina studentů nejspíše nemá přístup – nechat studenty si chvilku jen odpočinout a hrát není jen v pořádku, je to klíčové k dalšímu úspěšnému postupu, a každému studentovi bychom měli umožnit se seznámit s prostředím virtuální reality a zvyknout si na jeho využívání.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Jste spokojeni s tím, jak váš robot jezdí? Líbí se vám virtuální realita? Stihli jste dokončit vše, co jste načali?

3.2 Unity 3D VR – 12 hodin

Celá sekce, ve které probíhá práce v Unity, se může zásadně odlišovat v závislosti na znalostech, které žáci mají, a to v oboru programování, 3D modelování a nebo přímo ovládání Unity samotného. Cíle toho bloku vedou k personalizaci předpřipravené VR scény a zde jsou jen předloženy návrhy, ze kterých si musí lektor programu vybrat v závislosti na jeho konkrétních okolnostech. Zároveň nic nebrání skupině vycházející z pokročilejšího zázemí zvolit si cestu kompletně vlastní.

Pracovní listy

Výchozí scénář je poté založený na dodaných pracovních listech/návodech, které poskytnou čtenářům úvod do základních stavebních bloků poskytnutých enginem Unity, postupy na jejich vytváření a jednoduché editování. Součástí pracovních listů je také úvod do skriptování, pro který jsou vhodné alespoň základní znalosti programování, je však použitelný i bez nich.

Podrobně rozpracovaný obsah programu rámcově následuje pokyny právě těchto pracovních listů, v listech jsou však dodány detailnější informace v kontextu jednotlivých pokynů a celé postupy doprovází obrázky s popisky a šipkami, jenž slouží k dalšímu zjednodušení následování kroků postupu. Očekávaná verze Unity je 2019.

Testování a nasazení nové scény

Jeden z problémů, který je nutné v rámci druhé poloviny programu řešit, je jakým způsobem otestovat žáky vytvořené scény dohromady s virtuální realitou. Headset je často připojen a nastaven pouze na jeden počítač a není triviální přenášet vytvořené scény mezi počítači žáků a testovacím počítačem. Možných řešení je několik a opět se odvíjí od schopností žáků a možností, které mají lektoři programu k dispozici. Nejjednodušší, ale nejméně uspokojivé řešení, je použít pro VR předpřipravenou scénu.

Dále je možné například uspořádat "mini-soutěž" o nejlepší scénu, a tu pak použít pro potřeby soutěže. V případě dostatku času (což také záleží na počtu vývojářských skupin) je možné postupné nahrávání žáky vytvořených scén do počítače s VR, a to třeba pomocí flash disků, sdíleného disku/úložiště anebo nějakého repozitáře. Míra testování scén ve virtuální realitě je pak na rozhodnutí lektorů programu, přičemž musí být řešena časová náročnost tohoto přístupu, například časovým omezením testování jedné skupiny.

Samotné nahrání scény do Unity je v zásadě pouze překopírování souboru scény, pokud však žáci v rámci úprav scény vytvoří nové assety (skripty, modely, materiály, prefaby etc.), je nutné tyto assety





překopírovat spolu se scénou. V tomto případě je pak lepší mít v rámci projektu vytvořenou složku pro každou skupinu, která zamezí vzájemnému přepisování souborů ve složkách projektu. Při vytváření vlastních skriptů je pak třeba dbát na to, aby žádné dva skripty nenesly stejné jméno.

Rozšíření skriptování

Pokud je ve skupině žáků větší počet pokročilejších programátorů, pak je možné uvnitř scény vytvářet další komponenty s vlastním chováním, případně (po dohodě s lektory) měnit skripty, které jsou součástí předpřipraveného projektu. Některé možné příklady jsou uvedeny níže. Je třeba však dodat, že takové úpravy mohou vyžadovat i změnu pravidel turnaje na konci kroužku. Pokud se skupina žáků dostane do této fáze vývoje, je možné zapojit značnou míru kreativity.

- 1) Editace předpřipravených skriptů.
 - a) Skripty luku a šípů. Chování luku a šípu je možné upravit větším množstvím způsobů, od základních věcí, jako je rychlost letu šípu nebo základní doby mezi výstřely po komplexnější věci, jako je více různých typů šípů nebo luk s naváděnými střelami, možností je velké množství, ale zůstává faktem, že tyto úpravy silně mění charakter turnaje na konci programu.
 - b) Skript robota. Základní chování robota představuje pouze nastavení pozice "terče" ve scéně na pozici trackeru oproti headsetu dle souřadnic ze skutečného světa. Jako jedna z méně invazivních úprav je například přidání šipky, aby bylo ve virtuální realitě vidět, jakým směrem je robot natočený. Je však také možné přidat různé nové vlastnosti virtuálnímu robotovi, změnit jeho relativní pohyb oproti pohybu skutečnému nebo mu přidat nějaké "obranné" možnosti, například štít, rozdvojení (poté by musel střelec poznat, který robot je pravý po zvuku ze skutečného světa) nebo nějaký úhybný manévr po vertikální ose. Tyto úpravy samozřejmě také výrazně mění ráz soutěže a je nutné adekvátně upravit ukončení programu.
- 2) Vytvoření nového chování do scény. Možnosti nových prvků, které je možné přidat do scény, je široké a do velké míry je zde možné upravit průběh samotné hry. Do této kategorie spadají všechny úpravy scény, které zároveň mění charakter činnosti prováděné žákem ve virtuální realitě. Možné je například přidat do scény nějaký sekundární úkol (ve virtuální realitě například funguje poměrně dobře házení různými předměty) nebo úplně otočit scénář tak, že by hráč ve VR měl za úkol "chránit" jezdícího robota před nějakými virtuálními nástrahami.
- 3) Grafické efekty propojené s chováním ve scéně. Pod tuto kategorii spadají úpravy ve scéně, které nevedou k zásadní změně chování uživatele. Například přidání vlastního efektu zásahu v závislosti na trefenou překážku nebo například zrychlený průběh dne a noci ve scéně měnící vizáž scény v průběhu.

Využití 3D modelování pro Unity

Tvorba vlastních modelů může mít v tomto projektu široké využití. Tato podkapitola se zaměřuje na tvorbu 3D modelů do Unity scény. Žáci mohou vytvořit vlastní grafiku pro libovolnou součást scény nebo herní objekt. Je však důležité dohlédnout na několik klíčových bodů, pokud to žákům bude umožněno. Je například vhodné, aby vzhled objektů stále zhruba odpovídal jejich účelu (například aby byl jasně rozpoznatelný luk a jak jej používat). Dále je důležité, aby předměty byly viditelné, a také, aby nevhodně vytvořené modely výrazně nesnižovaly výkon aplikace.

Velké množství extrémně náročné geometrie by totiž mohlo program učinit nepoužitelným podobně jako nevhodně zvolené modely pro reprezentaci objektů. Pokud si budou žáci přát projevit svého





uměleckého ducha, je to vítáno, avšak je důležité mít na mysli, že velmi jemné detaily modelů nejsou v praxi vždy tvořeny pomocí polygonů, nýbrž normálových či paralaxních map. Texturování a tvorba materiálů jsou také možné, ale od žáků se vzhledem k zaměření a hodinové dotaci tohoto programu příliš neočekávají.

V zásadě však žáci mohou tvořit modely a materiály pro libovolné existující objekty ve scéně a rovněž přidávat nové. Zajímavým rozšířením také může být tvorba animací pro objekty scény. Unity samo v novějších verzích obsahuje nástroj ProBuilder, který pro tyto účely může postačovat, avšak pro náročnější práci je doporučeno použít specializovanou aplikaci jako například Blender, což je zdarma dostupný profesionální open-sourcový nástroj, který umožňuje jak tvorbu 3D modelů a materiálů, tak animací a mnoho dalšího.

Využití 3D modelování pro 3D tisk

Kromě tvorby nových objektů do virtuální scény lze žáky povzbudit k tvorbě 3D modelů za účelem jejich tisku na 3D tiskárně. Pro účely projektu tak byly vytisknuty pro roboty držáky na Vive trackery, ale lze pro žáky také například vytisknout malý suvenýr jako upomínku na program nebo trofej pro vítěze závěrečného turnaje, kterou si žáci sami vymodelovali.

Při tvorbě 3D modelu za účelem 3D tisku je však velmi důležité splnit několik požadavků, aby tisk proběhl úspěšně a v ideálním případě rovněž efektivně. Je důležité myslet například na následující body:

- všechny části modelu a jejich spoje musí mít (dostatečnou) hloubku (např. nesmí být část modelu tvořena pouze jednou stěnou)
- všechny normály stěn musí směřovat ven z objektu
- model musí být uzavřený
- model se nesmí překrývat
- model musí mít vhodnou velikost (přesnost tiskárny vs maximální velikost objektu)
- a další



3.2.1 Úvod do Unity – 3 hodiny

1. – 3. hodina

Vzhledem k podobnému charakteru vývoje a organizaci následujících dvou hodin je jejich metodická část popsána společně. Provozní varianta pracovních listů je sjednocena s druhou části této metodiky a studenti by měli větší část tohoto bloku strávit učením se Unity a postupováním ve vlastním tempu.





Forma a bližší popis realizace

Hromadná forma. Základní vlastnosti Unity. Rozbor předpřipravené scény a interpretace hlavních myšlenek fungování jejích částí. Nastínění možností rozšíření.

Metody

Ukázky, konzultace a dotazy

Pomůcky

HTC Vive, počítač pro VR, počítač s Unity, pracovní listy

V této hodině se žáci naučí základní vlastnosti editoru Unity a vyzkouší si aplikaci některých prvků programu, které lze v enginu použít. Poznají podobu oken editoru (Scene, Game, Project, Hierarchy, Inspector) a dozví se jejich účel. Zjistí, že v Unity se tvoří *scény* (Scene), z nichž v každé jsou umístěny *herní objekty* (GameObject). Herní objekty se skládají z *komponent* (Component). Komponenty určují pozici, natočení a velikost objektů (Transform), fyzikální chování (Rigidbody), zjišťují kolize s ostatními objekty (Collider), určují jejich tvar a vzhled (Renderer) či chování (Script). To vše bude předvedeno na jednoduché ukázkové scéně, s níž mohou žáci samostatně experimentovat.

V době tvorby pracovních listů byla aktuální stabilní verze Unity 2019 (2020 byla v experimentální fázi.) a vzhledem k tomu, že je Unity každoročně aktualizováno, je vhodné držet se verze, pro kterou byl tento program vytvářen. Je samozřejmě možné provést úpravy s tím, že bude třeba aktualizovat informace v pracovních listech. Pravděpodobné je, že většina základních postupů zůstane ještě několik dalších let stejná a bude se lišit pouze vzhled rozhraní, nelze na to však spoléhat.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Překvapilo vás, z čeho se skládá virtuální scéna? Líbí se vám prostředí Unity? Přijde vám manipulace s objekty intuitivní? Jaký objekt ve scéně jste vytvořili? Šlo vám skládání objektů, jak jste chtěli?

3.2.2 Částicové efekty v Unity – 3 hodiny

1. – 3. hodina

Forma a bližší popis realizace Individuální forma. Úpravy předpřipravené scény.

Metody Samostatná práce, konzultace a dotazy

Pomůcky

HTC Vive, počítač pro VR, počítač s Unity, pracovní listy

Tento blok opět není obsahově rozdělen na jednotlivé hodiny, studenti si po projití pracovních listů vnořených přímo do jejich scénáře vytyčí vlastní cíl.





V těchto hodinách se žáci věnují především grafické úpravě částí virtuální scény, jejíž předem připravený základ obdrží na začátku tohoto bloku. Tento základ bude obsahovat oblast pro pohyb robota, avatar robota tvořící virtuální cíl a základní implementaci luku a šípů. Bude možné se v něm pohybovat jak prostřednictvím vlastního fyzického pohybu při použití VR headsetu, tak pomocí klávesnice a myši (za účelem rychlého testování). Bude jim předvedena možnost tvorby částicových efektů, kterou si mohou následně samostatně vyzkoušet, přičemž vzniklé efekty mohou později použít pro efekty zásahu či minutí cíle ve virtuální scéně.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Jaký efekt byste chtěli nejraději vytvořit? Chovaly se částice ve vaší scéně, jak jste chtěli? Zkusili jste vytvořit více částicových efektů? Jaké možnosti nastavení vám nejvíce schází? Přežilo Unity vaše experimentování? Podařilo se vám vytvořit vizuálně působivý efekt?

3.2.3 Osvětlení a materiály v Unity – 3 hodiny

1. – 2. hodina

Forma a bližší popis realizace

Individuální forma. Úpravy předem připravené scény.

Metody

Samostatná práce, konzultace a dotazy

Pomůcky

HTC Vive, počítač pro VR, počítač s Unity

Náplní počáteční části tohoto bloku bude práce se světly. Od jednoduchého ambientního osvětlení přes základní bodové a směrové zdroje světla až případně po aplikace odrazů obrazu z okolí (Reflection Probe).

Za tím účelem jsou připraveny pracovní listy v kategorii Unity – grafika, zaměřené na bodové světlo. Obsah pracovních listů je současně vložený do scénáře této metodiky. Jádro těchto dvou hodin je o něco jednodušší než těch předchozích, může se stát, že někdo dokončí práci dříve. Jako postup pro tento případ doporučuji nechat je po vlastní ose pokračovat do třetí hodiny a poté mohou začít bez omezení experimentovat s grafickými editacemi jejich vlastní scény. Experimentování však nebraňte ani mimo tento případ. Závěrečným cílem několika bloků těchto hodin bude vytvořit scénu co nejvíce odpovídající vašim představám, zde se dozvíte, jaké máte k dispozici nástroje k jejich dosažení.

3. hodina

Forma a bližší popis realizace

Individuální forma. Úpravy předem připravené scény.





Metody

Samostatná práce, konzultace a dotazy

Pomůcky

HTC Vive, počítač pro VR, počítač s Unity

Žákům bude dále předveden způsob importu textury a vytvoření materiálu pro určení vizuálních vlastností vykreslovaných objektů. Budou jim přiblíženy nejjednodušší zákonitosti zobrazování na fyzikálním základě (Physically based rendering), s jehož využitím lze definovat realisticky, věrně vypadající materiály. Bude vysvětlen vliv a význam vlastností jako je barva materiálu, odraz a rozptyl světla (vliv na parametry Specular a Diffuse), dopad hladkosti (lesku; Smoothness) materiálu, odlišnosti mezi kovy a nekovy (Metalness). K této hodině se dá využít pracovní list z Unity – základy, zaměřený na materiály.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Rozumíte, proč stačí informace o směru, a nikoliv o pozici zdroje světla? (Tip: Je to daleko.) Líbí se vám více směrové nebo bodové světlo? V jakých situacích byste raději použili raději směrové? Jaké barvy jste použili? Jak barvy interagovaly s vámi nastaveným světlem?

3.2.4 Úprava tvaru scény v Unity – 3 hodiny

1. hodina

Forma a bližší popis realizace Individuální forma. Úpravy předem připravené scény.

Metody Samostatná práce, konzultace a dotazy

Pomůcky

HTC Vive, počítač pro VR, počítač s Unity

V této hodině se žáci naučí některé metody vhodné k úpravě tvaru virtuální scény, které Unity nabízí prostřednictvím instalovatelného modulu ProBuilder. Jak tento modul nainstalovat a základy jeho použití je detailně popsáno ve studentské části této metodiky. Unity disponuje automatickým systémem instalace balíčků, který je k tomu účelu využit. ProBuilder připomíná programy na 3D modelování, ale je o něco jednodušší a, což je nejdůležitější, napojený přímo na Unity. Jeho napojení umožňuje z modelů získat dodatečné informace, které by jinak bylo třeba vyrobit ručně, například kolizní boxy objektů.

ProBuilder je v mnoha ohledech relativně přímočarý, pokud tedy bude příliš složitý (třeba aplikace textury v něm může být trochu kostrbatá a je třeba mít představu o 3D objektu), je možné na místo probuilder skládat scénu z Unity základních tvarů. Na druhou stranu, pokud bude nedostačující (nemá





89

podporu pro animace a není vhodný pro složité modely), je možné použít téměř libovolný externí program a jeho výstup pak do Unity naimportovat. (Tento postup silně závisí na zvoleném programu a vyžaduje externí znalosti, které zde nebudou rozebírány.)

2. – 3. hodina

Forma a bližší popis realizace

Individuální forma. Úpravy předem připravené scény.

Metody

Samostatná práce, konzultace a dotazy

Pomůcky

HTC Vive, počítač pro VR, počítač s Unity

Aplikací znalostí získaných v průběhu předchozích lekcí si žáci mohou dotvořit virtuální prostředí podle svých představ. Práce v následujících dvou hodinách bude pokračovat dostatečně homogenně, pro její jednotný popis.

Při následování postupu ze scénáře je třeba občas pohlídat, zdali student nepřeskočil některý z předchozích kroků. Větší část této a následující hodiny bude věnována grafickému rozvoji scény, přičemž doba uskutečňování jednotlivých kroků bude záležet na cílech jednotlivých studentů. Je možné, že někteří studenti si vytyčí laťku relativně nízko a budou hotoví předčasně.

Pravděpodobně bude docházet i k opačnému případu, kdy si studenti naplánují příliš mnoho a čas těchto hodin jim nebude stačit. Ke korekci těchto potíží budete potřebovat analyzovat, jak rychle studenti postupují a případně jejich cíle usměrnit. Na prvním místě v motivaci těchto hodin je pak samozřejmě učení se nových dovedností v oblasti 3D modelování. Součástí této metodiky (ve třetí části) jsou i teoretické průpravy 3D grafiky pro lektory. V případě, že někteří studenti budou na pochybách o tom, jak pokračovat, je vhodné následovat různé postupy spolu s nimi a ukázat jim řešení.





Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.

Jakou verzi Unity používáte? Způsobila instalace nových balíčků nějaké potíže? Vymodelovali jste i nějaký vlastní objekt? Bavilo vás 3D modelování? Jakým způsobem jste vylepšili vaši scénu? Jaký objekt bylo nejtěžší vytvořit?

3.3 Vylepšování – 4 hodiny

3.3.1 Finalizace obou částí – 4 hodiny

1. – 4. hodina

Vzhledem k podobnému charakteru vývoje a organizace následujících hodin je jejich metodická část popsána společně.

Forma a bližší popis realizace

Skupinová forma. Úpravy a dokončení robota i VR aplikace, příprava taktiky pro závěrečný turnaj.

Metody

Samostatná práce, dotazy

Pomůcky

LEGO Mindstorms, HTC Vive, počítač pro VR se síťovým úložištěm

Žáci dokončují práci, kterou nestihli v uplynulých lekcích, a případně upravují a vylepšují svá řešení robota i VR aplikace. Mohou například při konstrukci robota aplikovat nové poznatky získané během práce na VR aplikaci a podobně. Současně si připravují strategii pro závěrečný turnaj, přičemž mohou využít znalosti silných a slabých stránek vlastního řešení, případně obtížnosti ošetření jednotlivých problémů, se kterými se při přípravách potýkali, k odhadu slabin soupeřů. Zároveň intenzivně testují spolehlivost vlastního řešení a brání se "průmyslové špionáži" konkurenčních týmů.

V těchto závěrečných hodinách sloužících na dokončení a vylepšování žákovských projektů poskytujeme svoji pomoc, co se všech aspektů projektu týče, a pomáháme účastníkům kurzu v testování jejich řešení. Přestože bychom neměli žáky nijak ovlivňovat v jejich celkovém návrhu, je v pořádku se projít kolem pracovišť jednotlivých skupin a ptát se na fungování daného robota. Některé skupiny, které by jinak o pomoc nepožádaly, si vzpomenou na otázky, když se jich sami budeme ptát. A i v případě, že skupina žádnou pomoc nepotřebuje, využijeme informace při následné organizaci turnaje – budou se nám snáze vymýšlet zajímavější páry pro turnajový žebřík – a celá akce bude o to zajímavější.

Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod.





Jste hrdí na svůj dokončený projekt? Co byste vylepšili? Co byste udělali znova?

3.4 Turnaj – 3 hodiny

3.4.1 Zkoušení výsledků a turnaj – 3 hodiny

1. – 3. hodina

Vzhledem k podobnému charakteru vývoje a organizace následujících dvou hodin je jejich metodická část popsána společně.

Forma a bližší popis realizace

Skupinová forma. Zde dochází k otestování závěrečných výsledků a k turnaji mezi žáky.

Metody

Aktivizační, soutěž, diskuze.

Pomůcky

LEGO Mindstorms, HTC Vive, počítač pro VR se síťovým úložištěm

1. V závěrečných hodinách proběhne samotný turnaj. Žákovské týmy obeznámíme s pravidly a strukturou turnaje, a po tomto krátkém úvodu se týmy postupně vystřídají a jsou obodovány za kvalitu jejich řešení v různých aspektech. V případě menšího počtu skupin se dá využít způsob turnaje "každý s každým", v případě větších skupin se pak dá sestavit klasický turnajový žebříček, ve kterém jdou vždy dvě skupiny proti sobě a vítěz se posouvá do dalšího kola, kde jde proti vítězům z ostatních párů, a tak dále až k vítězi posledního kola, a tedy i celého turnaje.

2. V případě lichého počtu skupin v kole je vhodnější nechat jednu skupinu se střetnout se dvěma různými protivníky než nechat ji přeskakovat kola, dokud nebude skupinek licho bez ní, ale je možné, že na takovou akci nebude dostatek času.

K turnaji je možné přistoupit dvěma způsoby, v závislosti na počtu zúčastněných týmů. V případě nízkého počtu vícečlenných týmu je možné provést turnaj systémem vzájemných soubojů "každý s každým", tedy tak, že si každý tým vyzkouší stát proti každému dalšímu týmu a každý robot byl otestován v každé aplikaci.

V případě většího množství týmů je pak možné přistoupit k časově úspornějšímu provedení turnaje formou vyřazovacích bojů mezi dvojicemi, které jsou určeny losem. Je nutné počítat s časovou režií, např. pro vložení VR trackeru do robota, přípravou herní oblasti apod. Také je při návrhu pořadí soubojů vhodné vzít v úvahu fakt, že žák v pozici virtuálního střelce pak s každou další hrou bude získávat zkušenosti a jeho výsledky se budou pravděpodobně zlepšovat.

Po skončení turnaje bude vyhlášen vítězný tým, a zejména budou prodiskutována a vyzdvihnuta obzvlášť zdařilá či originální technická řešení robotů a grafická zpracování virtuálních prvků. Během průběhu turnaje je vhodné pořizovat fotodokumentaci vytvořených robotů, neboť tyto bude nutné závěrem alespoň částečně dekonstruovat.

Na závěr doporučuji pohovořit se studenty o návrzích, které fungovaly nejlépe, a pochválit dobré nápady.





Reflexe

V závěru probíhá popisná zpětná vazba k práci. Reflexe je důležitou součástí hodiny – zpětná vazba od účastníků: jak se jim pracovalo, jakou si odnáší zkušenost, dovednost, co pro ně bylo nové, neočekávané, obtížné apod. Líbily se ti technologie použité v tomto programu? Spolupracovalo se ti dobře ve skupině? Bavilo tě tvořit scénu a skládat robota?





3.5 Zdroje a doporučená literatura

EV3 Lessons [online]. Pennsylvania: Droids Robotics, 2014 [cit. 2021-9-3]. Dostupné z: https://ev3lessons.com/en/

Návody na stavění pro 31313, LEGO[®] MINDSTORMS[®] EV3, LEGO[®] MINDSTORMS[®]. *LEGO* [online]. Billund, Dánsko: LEGO [cit. 2021-9-3]. Dostupné z: https://www.lego.com/cscz/service/buildinginstructions/31313

PŘÍRUČKY PRO NOVÉ MAJITELE [online]. Praha: Prusa Research, 2012 [cit. 2021-9-3]. Dostupné z: https://www.prusa3d.cz/novy-uzivatel-mk3/

HTC Vive [online]. Tchao-jüan, Tchaj-wan: HTC Corporation, 2015 [cit. 2021-9-3]. Dostupné z: https://www.vive.com

Unity Documentation [online]. San Francisco: Unity Technologies, 2005 [cit. 2021-9-3]. Dostupné z: https://docs.unity3d.com/

LaViola, Joseph J., Ernst Kruijff, Ryan P. McMahan, Doug A. Bowman, and Ivan Poupyrev. *3D User Interfaces: Theory and Practice*. Boston: Addison-Wesley, 2017. ISBN 978-0134034324.

Záznamy obrazovky dostupné v programech:

LEGO[®] MINDSTORMS[®] EV3, LEGO[®] MINDSTORMS[®]. *LEGO* [online]. Billund, Dánsko: LEGO [cit. 2021-9-3]. Dostupné z: https://www.lego.com/cs-cz/themes/mindstorms/downloads

Unity [online]. San Francisco: Unity Technologies, 2005 [cit. 2021-9-3]. Dostupné z: https://unity.com/





Příloha č. 1 - Seznam metodických příloh – pracovních listů

- 1. Lego mindstorms EV3 skládání
- 2. Unity základy 1–5
- 3. Lego Mindstorms EV3 programování
- 4. Unity grafika 1–5
- 5. Unity scriptování 1-4





Příloha č. 2 – Závěrečná zpráva o ověření programu v praxi



Výzva Budování kapacit pro rozvoj škol II Povinně volitelná aktivita č. 3, 4, 6 a 7

Zpráva o ověření programu v praxi

závěrečná¹

V závěrečné zprávě uvádějte souhrnné informace shrnujíci/rekapitulující celý proces (všechna) ověření programu, popř. informace o ověření programu, které bude realizováno jen s jednou ověřovací skupinou.

I.

Příjemce	Středisko služeb školám a zařízení pro další vzdělávání pedagogických pracovníků Brno, příspěvková organizace	
Registrační číslo projektu	CZ.02.3.68/0.0/0.0/16_032/0008190	
Název projektu	Ve škole po škole	
Název vytvořeného programu	Moderní technologie	
Pořadové číslo zprávy o realizaci	7.	

II.

Místo ověření programu	Datum ověření programu	Cílová skupina, s níž byl program ověřen ²
Gymnázium Zastávka	10/2019 až 6/2021	11 žáků 2. a 3. ročníku střední školy, nadaní žáci





Nehodící se škrtněte.
I kvoďto stružně obaval

Uveďte stručně charakteristiku a velikost skupiny (např. 25 žáků 7. ročníku ZŠ apod.) a název organizace.



Nejlepším řešením je mít k dispozici učebnu s moderními počítači, alespoň jeden pro každou skupinu. Program je zvládnutelný i na slabších počítačích, ale mohou nastat zbytečná zdržení. Dále je vhodné program realizovat mimo letní měsíce nebo v dostatečné klimatizované učebně.

c) Bude/byl vytvořený program upraven?

Po obsahové stránce nebyl program na základě ověření upraven. Pouze byla podrobněji rozpracována jednotlivá témata.





d) Jak a v kterých částech bude program na základě ověření upraven?

V návaznosti na druhé ověřování nebyly doporučeny žádné obsahové a programové úpravy.

3. Hodnocení účastníků a realizátorů ověření³

a) Jak účastníci z cílové skupiny hodnotili ověřovaný program?

Pandemická varianta závěru nedala prostor k organizovanému zhodnocení programu ze strany účastníků, ale už jen ten fakt, že se v posledním měsíci školního roku podařilo sehnat několik náhradníků a s ostatními lidmi ověření dokončit v méně než polovičním plánovaném čase mluví za výraznou atraktivitu programu.

b) Co bylo v programu hodnoceno v rámci ověřovací skupiny nejlépe?

Z pohledu lektora studenti nejvíce oceňovali možnost přístupu a hlavně velký rozsah možností s typicky nedosažitelným hardwarovým vybavením.

c) Jak byl hodnocen věcný obsah programu?

Studenti projevovali nezanedbatelný zájem o všechny obsahové složky programu a jeho naplnění přistupovali aktivně, často nad hodinový rámec. Otevřený "problém" se snažili vyřešit a především pochopit jednotlivé kroky. Následně bylo znát získání samostatnosti v určitých krocích.

d) Jak bylo hodnoceno organizační a materiální zabezpečení programu?

Organizační a materiální zabezpečení programu bylo adekvátní jeho požadavkům. S drobnými lokálními potížemi viz problémy z ověřování.

e) Jak byl hodnocen výkon realizátorů programu?

Účastníci programu hodnotili výkon realizátorů pozitivně.

f) Jaké měli účastníci výhrady/připomínky?

Naprosto drtivá většina výhrad a připomínek týkajících se programu byly směrovány vůči problémům způsobeným pandemickou situací. Objevily se výhrady, že byly jednotlivé bloky hodin časově příliš vzdáleny od sebe. V posledním měsíci realizace byly bloky častěji, ale žáci byli dost zaneprázdněni plněním ostatních, především školních povinností. Nicméně můžeme konstatovat, že menší časový odstup mezi jednotlivými bloky byl pro účastníky přínosem.

g) Opakovala se některá výhrada/připomínka ze strany účastníků častěji? Jaká?

Žádná závažnější připomínka se nevyskytla, několik studentů by preferovalo hodiny programu blíže u sebe.

 h) Budou případné připomínky účastníků zapracovány do další verze programu? Pokud ne, proč?





³ Vychází z evaluačních dotazníků žáků, učitelů, realizátorů programů – pracovníků neformálního vzdělávání či záznamů z rozhovorů s dětmi, které příjemce uchovává pro kontroly na místě.

Žádné připomínky účastníků nesměřují k nutnosti upravení programu.

i) Jak byl program hodnocen ze strany realizátorů programu?

Ověření programu bylo pro realizátory velmi náročné opět z důvodu uzavření škol.

j) Navrhují realizátoři úpravy programu, popř. jaké?

Po ověření nevyvstaly žádné nutné úpravy programu.

k) Budou tyto návrhy realizátorů zapracovány do další verze programu? Pokud ne, proč?

Doporučení o pravidelnosti a častosti konání jednotlivých bloků programu nebude změněno, jelikož připomínky přišli pouze od části studentů, při čemž zbytek studentů byl naopak proti této změně.

 Konkrétní výčet úprav, které budou na základě ověření programu zapracovány do další/finální verze programu:

Po druhém ověření byla podrobněji rozpracována jednotlivá témata. Obsahové úpravy nebyly prováděny.

	Titul, jméno, příjmení	Datum a místo	Podpis
Zpracoval/a	Ing. Miroslav Novotný	19. 7. 2021	New



