



Pedagogická  
fakulta  
Faculty  
of Education

Jihočeská univerzita  
v Českých Budějovicích  
University of South Bohemia  
in České Budějovice

Jihočeská univerzita v Českých Budějovicích  
Pedagogická fakulta  
Katedra informatiky

Bakalářská práce

# Tvorba projektů pro výuku programování v prostředí Unreal Engine 4

Vypracoval: Dominik Paclt  
Vedoucí práce: doc. PaedDr. Jiří Vaníček, Ph.D.

České Budějovice 2020

## Anotace

Bakalářská práce se věnuje metodice výuky základů programování na druhém stupni ZŠ prostřednictvím prostředí Unreal Engine 4.

Teoretická část se zabývá popisem prostředí Unreal Engine 4, spolu se způsobem práce v něm. Dále je toto prostředí porovnáno z pohledu vhodnosti k výuce programování s prostředími Scratch a MakeCode.

Praktická část obsahuje postup, kterým jsem vytvářel a ověřoval sadu úloh, podle které se žáci učí základům programování a základní orientaci v prostředí Unreal Engine 4. V práci jsou dále obsaženy metodické podklady pro vyučujícího, zahrnující poznámky k úlohám.

V příloze jsou obsaženy pracovní listy pro žáky k jednotlivým lekcím. Projekt, ve kterém žáci za pomoci pracovních listů programují a vzorový projekt pro vyučujícího s vypracovanými částmi kódu.

## Klíčová slova

Výuka programování, věk žáka, Informatické myšlení, Unreal Engine 4, Epic Games Launcher, herní prostředí

## Annotation

The bachelor thesis deals with the methodology of teaching the basics of programming at the second stage of elementary school through the environment Unreal Engine 4.

The theoretical part deals with the description of the Unreal Engine 4 environment, together with the way of working in it. Furthermore, this environment is compared in terms of suitability for teaching programming with Scratch and MakeCode.

The practical part contains the procedure by which I created and verified a set of tasks, according to which students learn the basics of programming and basic orientation in the Unreal Engine 4. The work also contains methodological materials for the teacher, including notes on the tasks.

The attachment contains worksheets for students for individual lessons, a project in which students program with the help of worksheets and a sample project for the teacher with developed parts of the code.

## Keywords

Teaching programming, pupil age, Informatics thinking, Unreal Engine 4, Epic Games Launcher, game environment

## Poděkování

Děkuji doc. PaedDr. Jiřímu Vaníčkovi, Ph.D. za možnost realizace této práce pod jeho vedením, za rady a informace, které mi poskytl. Především za důvěru a trpělivost během realizace mé práce. Dále bych rád poděkoval Základní škole Choustník, zvláště panu řediteli Mgr. Karlu Zvěřinovi za možnost spolupráce a umožnění vedení kroužku, na kterém jsem svou práci mohl testovat.

## Prohlášení

Prohlašuji, že svoji diplomovou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své diplomové práce, a to v nezkrácené podobě – v úpravě vzniklé vypuštěním vyznačených částí archivovaných Pedagogickou fakultou elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

# JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH

Pedagogická fakulta

Akademický rok: 2019/2020

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	Dominik PACLT
Osobní číslo:	P17611
Studijní program:	B7507 Specializace v pedagogice
Studijní obor:	Fyzika se zaměřením na vzdělávání Informační technologie se zaměřením na vzdělávání
Téma práce:	Tvorba projektů pro výuku programování v prostředí Unreal Engine 4
Zadávací katedra:	Katedra informatiky

### Zásady pro vypracování

Kvalifikační práce je věnována výuce základů programování na druhém stupni ZŠ pomocí tvorby programovacího projektu v prostředí pro vývoj počítačových her Unreal Engine 4. Cílem práce je vytvořit a ověřit sadu přípravných úloh, které seznámí žáka s klíčovými prvky a částmi prostředí, stejně jako se základními koncepty, potřebnými k vytvoření projektu – jednoduché počítačové hry.

V teoretické části práce student stručně popíše program Unreal Engine 4 a způsob práce v něm, zhodnotí výhody a nevýhody Unreal Engine 4 pro výuku programování a následně porovná s jinými systémy pro výuku programování, např. Scratch.

V praktické části student vytvoří a ověří ve výuce sadu úloh, podle kterých budou žáci schopni samostatně postupovat v projektu s cílem vytvořit finální produkt. Úlohy budou vytvořeny tak, aby podporovaly aktivitu žáka, objevování nových poznatků, jeho tvořivost a samostatnost. Pozice vyučujícího v tomto kurzu bude koncipována nikoliv v roli předavače znalostí, ale v roli tutora (motivační, organizační, konzultační podpora žáka). Student též připraví sadu úvodních úloh na seznámení s prostředím Unreal Engine 4. Student dále vytvoří metodické podklady pro vyučujícího. Takto upravenou sadu úloh student publikuje pod otevřenou licenci.

Rozsah pracovní zprávy:	40
Rozsah grafických prací:	CD ROM
Forma zpracování bakalářské práce:	tištěná

### Seznam doporučené literatury:

1. LEE, Joanna. Learning Unreal Engine Game Development [online]. Packt Publishing Limited, 2016. ISBN: 9781784398156. Dostupné z: [https://subscription.packtpub.com/book/game\\_development/9781784398156/1/ch01m1sec10/the-history-of-unreal-engine](https://subscription.packtpub.com/book/game_development/9781784398156/1/ch01m1sec10/the-history-of-unreal-engine).
2. HORNEMAN, Jurie. Coder Dojo – Tvořte kódy: Navrhnete vlastní hru. Egmonth, 2017. ISBN: 978-80-252-4038-0.
3. MAX Wainwright. Programuj – Průvodce programováním. Svojtka&Co, 2018. ISBN: 978-80-256-2048-9.
4. REMEŠ, R. Programujeme v jazyku Python. Č. Budějovice: EF JU, 2008. ISBN 9788073941284.
5. FISHER, R. Učíme děti myslet a učit se. Praha, Portál, 1997. ISBN:978-80-262-0043-7.
6. TONUCCI, Francesco. Vyučovat nebo naučit? Praha, UK 1991. ISBN: 809010651X.
7. MCMANUS, Sean. Jak se naučit programovat v 10 lekcích. Svojtka&Co., 2017. ISBN: 978-80-256-2046-5
8. ČÁPKA, David. ITnetwork.cz [online]. 2019. Dostupné z: <https://www.itnetwork.cz/it-e-learning>.
9. Epic Games. Unreal Engine 4 Documentation [online]. 2019. Dostupné z: <https://docs.unrealengine.com/en-US/index.html>
10. CLEGG, Brian. BIRCH, Paul. Intenzivní kurz vedení lidí. Computer Press (CPress), 2004. ISBN: 80-251-0356-0

Vedoucí bakalářské práce: doc. PaedDr. Jiří Vaniček, Ph.D.  
Katedra informatiky

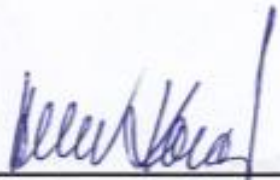
Datum zadání bakalářské práce: 19. listopadu 2019  
Termín odevzdání bakalářské práce: 30. dubna 2020

Informace o vypracování

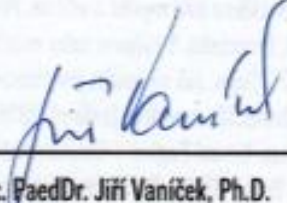
Práce je vypracována v souladu s požadavky katedry informatiky a je určena k obhajobě. Práce obsahuje teoretické a praktické části. Teoretická část je zaměřena na analýzu a návrh řešení. Praktická část obsahuje implementaci navrženého řešení. Práce je vypracována v souladu s požadavky katedry informatiky a je určena k obhajobě. Práce obsahuje teoretické a praktické části. Teoretická část je zaměřena na analýzu a návrh řešení. Praktická část obsahuje implementaci navrženého řešení.

Práce je vypracována v souladu s požadavky katedry informatiky a je určena k obhajobě. Práce obsahuje teoretické a praktické části. Teoretická část je zaměřena na analýzu a návrh řešení. Praktická část obsahuje implementaci navrženého řešení. Práce je vypracována v souladu s požadavky katedry informatiky a je určena k obhajobě. Práce obsahuje teoretické a praktické části. Teoretická část je zaměřena na analýzu a návrh řešení. Praktická část obsahuje implementaci navrženého řešení.

Práce je vypracována v souladu s požadavky katedry informatiky a je určena k obhajobě. Práce obsahuje teoretické a praktické části. Teoretická část je zaměřena na analýzu a návrh řešení. Praktická část obsahuje implementaci navrženého řešení. Práce je vypracována v souladu s požadavky katedry informatiky a je určena k obhajobě. Práce obsahuje teoretické a praktické části. Teoretická část je zaměřena na analýzu a návrh řešení. Praktická část obsahuje implementaci navrženého řešení.

  
doc. RNDr. Helena Koldová, Ph.D.  
děkanka



  
doc. PaedDr. Jiří Vaniček, Ph.D.  
vedoucí katedry





# Obsah

1.	Úvod.....	11
1.1.	Motivace výběru práce .....	11
1.2.	Informatické myšlení.....	12
1.3.	Cíle práce.....	12
1.4.	Metody práce.....	13
2.	Unreal Engine 4 .....	14
2.1.	Herní engine .....	14
2.2.	Úvod k prostředí Unreal Engine 4.....	14
2.3.	Actor.....	15
2.4.	Systém Blueprint .....	15
2.5.	Orientace v UE4 .....	21
2.6.	Instalace UE4 .....	22
3.	Porovnání UE4 a jiných systémů ve výuce programování .....	24
3.1.	Scratch.....	24
3.2.	MakeCode .....	25
3.3.	Unreal Engine 4.....	26
4.	Postup tvorby programovacích úloh .....	28
4.1.	Plán posloupnosti lekcí.....	28
4.2.	Vytváření lekcí.....	30
4.3.	Testování lekcí a vytváření podkladů pro vyučující .....	31
5.	Sada úloh.....	32
5.1.	Úvodní poznatky .....	32
5.2.	Část 1. Sestavování scénáře .....	34
5.3.	Část 2. Opakování bloků .....	45
5.4.	Část 3. Vlastní bloky .....	53
5.5.	Část 4. Vstupy a výstupy.....	60

5.6.	Část 5. Podmínky .....	72
5.7.	Část 6. Parametry .....	80
5.8.	Část 7. Volané události.....	88
5.9.	Část 8. Proměnné.....	96
6.	Závěr .....	103
	Seznam použité literatury a zdrojů .....	105
	Přílohy.....	107

# 1. Úvod

V dnešní době je každodenní život silně propojen s informačními technologiemi. Proto není divu, že čím dál větší část populace, především pak děti, tráví značnou část svého času na telefonu či u počítače, na kterých v mnoha případech hrají i hry. Spolu s tím se zvyšuje počet volných pozic v oblasti IT. To spolu se zvyšujícím finančním ohodnocením za příslušné profese dělá z těchto pozic lukrativní zaměstnání. Jednou z takových profesí je vývojář, v našem případě programátor.

Tato práce je tedy zaměřena na naučení dětí základům programování. K tomuto účelu využiji systém Unreal Engine 4 (dále pak UE4), který je v dnešní době jedno z nejpokročilejších prostředí pro vývoj her. Další výhodou tohoto prostředí je fakt, že lze použít grafické programování, respektive systém Blueprint, jenž by mohl, spolu s možností vidět svůj výtvar v provozu v téměř jakýkoliv moment vývoje, být vhodným prostředím pro pochopení principu programovacího jazyka.

## 1.1. Motivace výběru práce

V Unreal Engine 4 pracuji již více než pět let. Za tu dobu jsem v tomto prostředí vytvořil řadu osobních programů.

Již na střední škole mě uchvátil systém Blueprint, v němž se programuje skrze vizuálního skriptování (rozbor systému je v následujících kapitolách). Díky tomu jsem se v tomto prostředí naučil jednotlivé prvky programování mnohem lépe než při výuce na střední škole v jazyce C# v prostředí Visual Studio. Z toho důvodu mě překvapilo, že UE4 není k výuce programování v České republice téměř vůbec využíván.

*„Unreal Engine 4 je v dnešní době jeden z nejpobulárnějších systémů na trhu.“* [1] Jsou v něm vytvořeny tituly jako je Fortnite, GTA V, Player Unknown Battlegrounds a další.[2] Krom toho se UE4 začíná využívat i ve filmovém průmyslu. Důkazem toho může být televizní seriál „The Mandalorian“ vydaný na Disney+, který využil UE4 pro vytvoření speciálních efektů.[3].

Z těchto důvodů mě napadla myšlenka vytvoření vlastního výukového plánu v tomto prostředí. Z jedné strany se totiž jedná o prostředí, které by mohlo být vhodné pro výuku programování, a z druhé strany se jedná o prostředí, které může žáky zaujmout už jen díky produktům, které byly za pomoci UE4 vytvořeny.

## 1.2. Informatické myšlení

*„Je to způsob myšlení, který se zaměřuje na popis problému, jeho analýzu a hledání efektivních řešení.“[4]. „Informaticky myslící člověk ve svém životě odhaluje rutinní postupy a snaží se je optimalizovat, aby mu nezabíraly tolik času, a automatizovat je tak, aby se místo nich mohl věnovat třeba rodině nebo koníčkům.“[4]*

Jednou z výukových metod Informatického myšlení je výuka programování. *„Žáci zjistí, že mohou stroji zadat ke splnění i poměrně složité úkoly. Musí k tomu ale řešení přesně zformulovat a zapsat.“[4]*

Ve výuce informatického myšlení se využívá několika principů. Jedním z nich je skutečnost, že se člověk učí ze svých chyb, a proto je přirozené je dělat. Dalším je naučit žáky věřit svým vlastním schopnostem, naučit je vytrvale pracovat na větších projektech a získávat z této tvorby potěšení. V neposlední řadě se žáci učí lépe pracovat v týmu, neboť problémy dnešního světa jsou čím dále komplexnější a *„k jejich řešení jsou potřeba velké a dobře pracující týmy.“[4]*

## 1.3. Cíle práce

### Popis programu UE4 a zhodnocení vhodnosti pro výuku programování

Popíšu program Unreal Engine 4, jeho výhody pro vývoj her a jiných produktů. Vysvětlím pojmy herní engine, Actor a systém Blueprint. Systém Blueprint dále popíšu a demonstruji práci s ním na příkladu.

### Porovnání UE4 a jiných programů pro výuku programování

Zhodnotím výhody a nevýhody výuky programování za použití tohoto systému. Poté jej porovnáím s jinými systémy, určenými pro výuku programování, konkrétně se systémy Scratch a MakeCode. Ve své práci porovnáím mimo jiné přehlednost jednotlivých systémů, vhodnost pro výuku a možnosti komplexnějšího programování.

### Vytvořit sadu přípravných úloh

Vytvořím sadu úloh, podle kterých budou žáci schopni samostatně pracovat. Sada bude rozdělena do částí, z nichž se každá bude věnovat nějakému prvku programování. Tuto sadu úloh ověřím prostřednictvím vedení zájmového kroužku. Tato sada bude dále fungovat jakožto sada úloh pro základní orientaci v prostředí UE4.

## Vytvoření metodických podkladů pro vyučujícího

Vytvořím podklady pro vyučujícího, který by eventuelně mou sadu úloh chtěl použít při výuce. Podklady budou sloužit k popisu jednotlivých částí a dílčích lekcí, poukáží na doporučený způsob výuky dané části a poukáží na úlohy, které mohou vyžadovat zvýšenou aktivitu vyučujícího.

### 1.4. Metody práce

Více než pět let před samotným počátkem mé práce jsem v prostředí UE4 tvořil své osobní programy. Popis tohoto prostředí proto provádím dle své vlastní zkušenosti, kterou podporuji literaturou, jež jsem před anebo během své práce studoval.

Při snaze porovnat UE4 s jinými programy, které se využívají k výuce programování, jsem postupoval analyticky. Prozkoumal jsem koncepty, kterými se v těchto programech vyučuje. Vzhledem k tomu, že se mi o použití UE4 v Českém školství nepodařilo najít žádné informace, provedl jsem porovnání pouze ze své vlastní zkušenosti.

Z jednotlivých konceptů výuky programování jsem stvořil svůj vlastní koncept výuky, ze kterého jsem stvořil sadu úloh v prostředí UE4. Tuto sadu jsem tvořil za použití vlastního procesu, jenž se inspiroval procesem ADDIE. *ADDIE je zkratka pro pět fází vývojového procesu: analýza, návrh, vývoj, implementace a hodnocení. Model ADDIE se spoléhá na to, že každá fáze se provádí v daném pořadí, ale se zaměřením na reflexi a iteraci. Tento model poskytuje efektivní a soustředěný přístup, který poskytuje zpětnou vazbu pro neustálé zlepšování.*[4] Více o ADDIE na: <https://www.learnupon.com/blog/addie-5-steps/>. Napřed jsem zanalyzoval současné způsoby výuky programování. Podle těchto způsobů jsem navrhl vlastní konstrukci úloh a způsob distribuce žákům. Následně jsem podle této konstrukce vyvíjel jednotlivé úlohy, které jsem průběžně implementoval ve formě výuky na základní škole. Z této výuky jsem vyhodnocoval silné a slabé strany úloh. Podle informací z vyhodnocování jsem upravoval jednotlivé úlohy, nebo celou konstrukci úloh tak, aby byli mé úlohy co možná nejlepší.

Ze zjištěných poznatků při ověřování jsem vytvořil podklady pro vyučujícího, které mu usnadní práci s žáky, poukáží na místa, se kterými žáci mohou mít problémy.

## 2. Unreal Engine 4

Vývojářské prostředí Unreal Engine 4 je herní engine, vytvořený firmou Epic Games. První verze byla vydána roku 1998. V UE4 je mimo jiné vytvořena i populární hra „Fortnite“. V posledních letech se systém zaměřuje nejen na vývoj her, ale stává se z něj i nástroj pro úpravu videí, či nástroj pro architektky.[1] [6]

V následujících kapitolách popíšu termín herní Engine, představím UE4 a jeho základní součásti. Následně popíšu důležitou třídu jménem Actor, jež zastupuje mimo jiné programovatelné postavy. Dále popíšu systém Blueprint, u něhož je kód tvořen graficky, a využívám jej ve své sadě úloh. Jeho aplikaci popíši skrze příklad. Na závěr popíšu rozložení UE4 a jeho instalaci.

### 2.1. Herní engine

Herní engine je software, který poskytuje propojené nástroje a programy, jež pomáhají při konstrukci hry. Může obsahovat grafické programy, programovací programy, programy pro modifikaci zvuku či simulační programy. Díky tomu dává znatelný náskok oproti jiným způsobům vytváření her.[1]

### 2.2. Úvod k prostředí Unreal Engine 4

*„Unreal Engine 4 je v dnešní době jeden z nejpokročilejších systémů na trhu.“*[1] Můžou za to jeho možnosti pro doladění grafické stránky přímo ve hře (především pak systém světla a stínů, který je v dnešní době jedním z nejpokročilejších systémů), jeho rozsáhlé přizpůsobitelnosti pro uživatele, schopnost převést hru na velké množství platform a v neposlední řadě možnost zdarma vyvíjet jakýkoliv produkt (až do chvíle, kdy produkt začne generovat zisk).[1]

Programovacím jazykem UE4 je C++. Tímto jazykem je prostředí naprogramováno, i se s ním mohou za pomoci prostředí „Microsoft Visual Studio“ programovat produkty. Krom jazyka C++ je v UE4 ještě jeden způsob programování, systém Blueprint. Jde o systém vizuálního skriptování, který umožňuje rozšířit funkčnost kódu pomocí vizuálního skriptovacího jazyka (uzly, propojené čarami). Tento systém, si přiblížíme v následujících kapitolách.[6]

Další výhodou je volný přístup ke zdrojovému kódu Unreal Engine, který dává uživatelům možnost přetvořit samotný systém a vytvořit tím téměř cokoliv i nad rámec původních možností UE4.[1]

Unreal Engine se skládá z několika součástí, které spolupracují na řízení produktu. Některé z nich se věnují vzhledu finálního produktu (př. „Zvukový engine“ nebo „Grafický engine“) jiné se věnují chování finálního produktu (př. „Fyzikální engine“, „Systém umělé inteligence“, či „Systém Blueprint“).[1] [6].

Tyto jednotlivé součásti jsou vytvořeny tak, že pro práci v jedné části není potřeba znát části jiné. Pokud je však potřeba testovat např. kód postavy, je již vhodné pro názornost ovládat základy práce s mapou.[6]

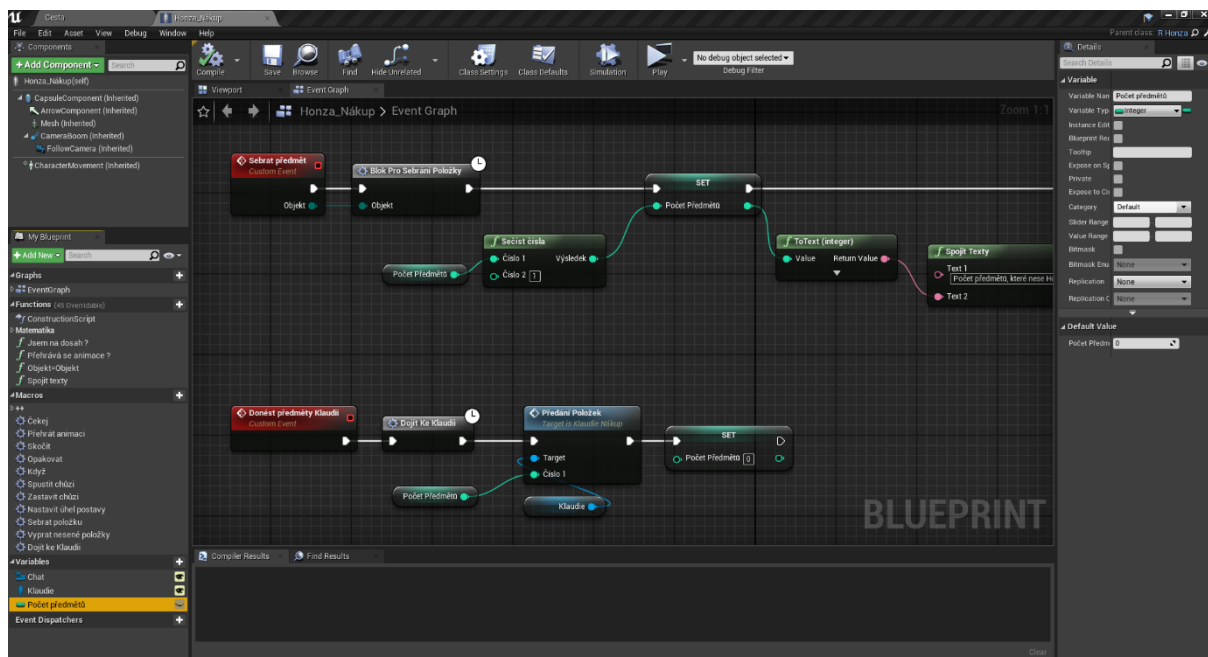
### **2.3. Actor**

Actor je objekt, který lze umístit do úrovně. Actors jsou obecná třída, která podporuje 3D transformace, jako je lokace, rotace a měřítko. Na Actors lze v určitém smyslu pohlížet jako na kontejnery, které obsahují speciální typy objektů zvané Komponenty. Tyto komponenty se dále starají o různé vlastnosti Actor, jako je například jeho ovládání, způsob, jakým se pohybuje, či způsob vykreslení.[6]

### **2.4. Systém Blueprint**

Jedná se o funkci přidanou až v Unreal engine ve verzi 4. Jak již bylo řečeno, jde o systém vizuálního skriptování, který umožňuje rozšířit funkčnost kódu pomocí vizuálního skriptovacího jazyka. Systém programování by se dal přirovnat k vytváření vývojového diagramu.[6]

Díky této schopnosti není nutné psát kód, čímž částečně odpadá potřeba kontrolovat syntaktické chyby. Skvělou funkcí Blueprint je, že můžete vytvářet proměnné, funkce i makra kliknutím na tlačítka „+ Function“, „+ Macro“ a „+ Variable“. Není proto potřeba vytvářet kód pro deklarování. S tím je spojena i výhoda, že takto vytvořené proměnné, funkce a makra jsou zobrazena v příslušných seznamech, a jsou viditelná po celou dobu skriptování v dané třídě (daném objektu) viz obrázek 1, levá část obrazovky.[1][6]



OBRÁZEK 1 UKÁZKA SYSTÉMU BLUEPRINT

Výrazný nástroj pro podporu vývoje v systému Blueprint je i možnost spustit testování téměř v každém momentu hry. Během testování je možné vidět průběh událostí a hodnot vlastností vizuálně přímo v kódu a v reálném čase.[1].

V tomto systému lze kromě tříd Actors programovat samostatné funkce i Makra, které lze následně používat globálně (s určitým omezením). Dále v něm lze modifikovat vlastnosti samotné úrovně, nastavovat třídy, starající se o celkový průběh výsledného produktu (jako je například třída „Player Controller“, starající se mimo jiné o přijímání vstupů na uživatele). V neposlední řadě s ním jde programovat vizuální rozhraní pro uživatele „HUD“, na něž lze zobrazit tlačítka, obrázky a jiné části, sloužící pro komunikaci s uživatelem.[6]

## 2.4.1. Vzhled prostředí Blueprint

V této kapitole popíšu systém Blueprint stručně a jen několik nejpodstatnějších částí, neboť komplexní programování v tomto systému je poměrně složité. Kompletní přehled o systému je k přečtení v oficiální dokumentaci UE4 na stránce: <https://docs.unrealengine.com/en-US/Engine/Blueprints/index.html>

Systém Blueprint, jak je možné vidět na obrázku dvě, je rozdělen na pět hlavních částí.

Levá část obrazovky se věnuje prvkům, jež daný Actor obsahuje. Ve vrchní s názvem „Components“ části jsou zobrazeny Komponenty, které Actor využívá. Pod tímto okénkem je



okénko „My Blueprint“, věnující se událostem, funkcím, makrům a dalším částem, specifickým pro UE4. Lze použít filtr, kterým si můžeme nastavit zobrazování pouze některých částí.[6]

Pravá část obrazovky nesoucí název „Details“ se věnuje popisu zvolené konkrétní části Actor (ať už de jedná o proměnnou, funkci, Komponent, ...). Může obsahovat informace jako je například. výchozí hodnota proměnné, datový tip, název, vstupy a výstupy funkcí a Maker atd. Téměř všechny informace z tohoto okénka lze upravovat klikáním na ně.[6]

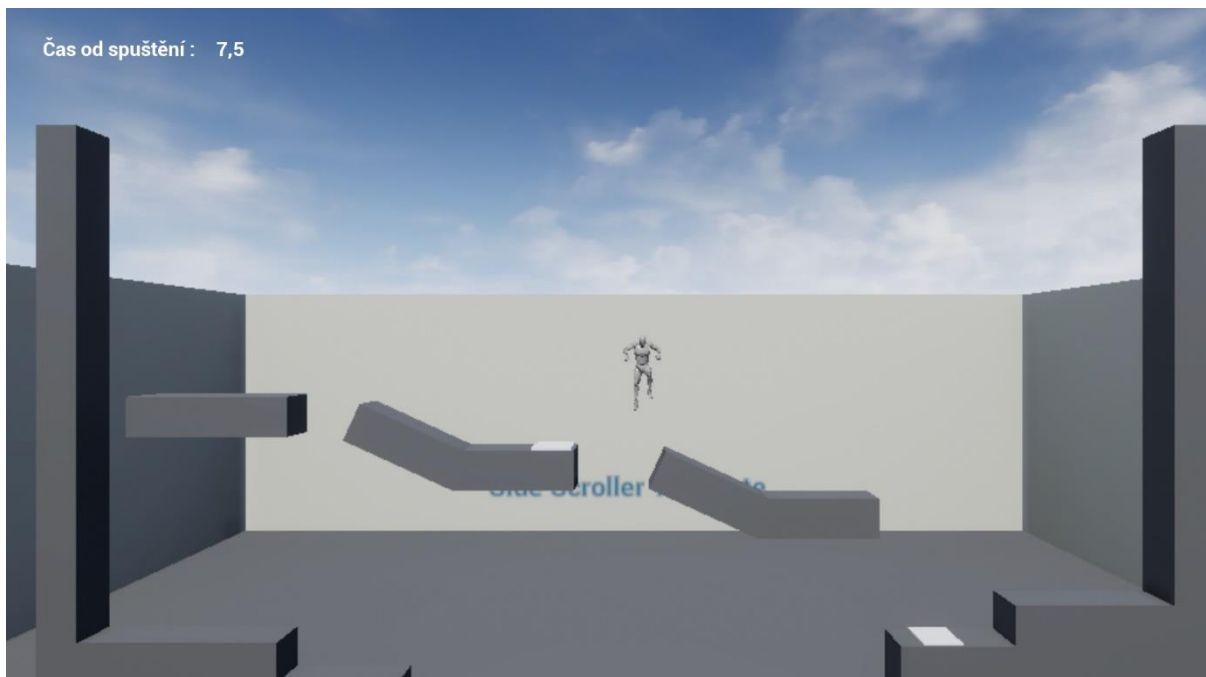
Spodní část obrazovky „Compiler Results“ a „Find Results“ se věnuje výpisu stavu Actor. Okénko „Compiler Results“ zobrazuje informace týkající se průběhu simulace, či chyby v programu (př. nedosažení odkazu na Actor při volání události). Skrze okénko „Find Results“ lze vyhledávat všechny místa použití určité části Actor, jako jsou funkce get a set proměnné.[6]

Vrchní část obrazovky (bez názvu) funguje jakožto lišta možností práce s Actor jako celkem. Lze zde ukládat Actors, testovat je, simulovat je, změnit nastavení jeho třídy apod.[6]

Střed obrazovky je věnován samotné konstrukci kódu, eventuelně v něm lze zobrazit reprezentaci vzhledu Actor v úrovni. Orientace v něm je možná pomocí kolečka myši pro oddalování a přibližování kódu nebo stisknutím pravého tlačítka myši s následným pohybem myši do stran, pro pohyb v kódu. Lze v něm zobrazovat a modifikovat jak grafy událostí, tak konkrétní funkce a makra. Přidávání nových uzlů je možné buďto přetáhnutím uzlu z okénka „My Blueprint“, nebo kliknutím pravého tlačítka myši do prázdného prostoru mezi uzly a následném vybrání požadovaného uzlu z nabídky.[6]

## **2.4.2. Vzorové použití systému Blueprint**

Způsob fungování kódu v systému Blueprint ukáži na příkladu. Pro demonstraci jsem zvolil příklad z úvodní sady úloh pro zlepšení orientace v systému Blueprint z dokumentace UE4. Tento příklad jsem mírně modifikoval pro demonstrační účely. Zadání, vedoucí k zhotovení neupravené formy příkladu, je dostupné na adrese: <https://docs.unrealengine.com/en-US/Engine/Blueprints/QuickStart/index.html>.



OBRÁZEK 2 UKÁZKA SIMULACE PŘÍKLADU Z KAPITOLY 2.6.2

### **Popis příkladu**

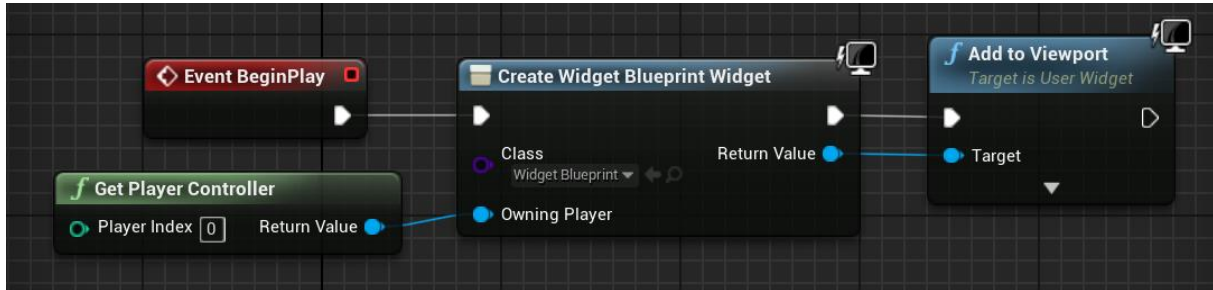
V tomto příkladu uživatel ovládá postavu, se kterou se může pohybovat doprava a doleva klávesami A a D. Zároveň s postavou může skákat za pomoci kláves W a mezerník. V úrovni jsou postaveny rampy, po kterých se může postava pohybovat. V úrovni jsou též dvě plošiny, které při kontaktu s postavou postavu vymrští do vzduchu. Na závěr je zde v levém horním rohu obrazovky časovač, který zaznamenává čas na zahájení simulace.

### **Při spouštění simulace**

Po zahájení simulace systém vygeneruje vzhled úrovně, načež začne spouštět jednotlivé objekty, které jsou buďto v úrovni přímo umístěny, nebo jsou s úrovní propojeny jiným způsobem (příklad v následujícím odstavci).

## Spouštění rozhraní

Příkladem je spuštění objektu „Widget Blueprint“, který má v tomto případě za úkol zobrazovat uživateli čas na spuštění simulace viz. obrázek 3.

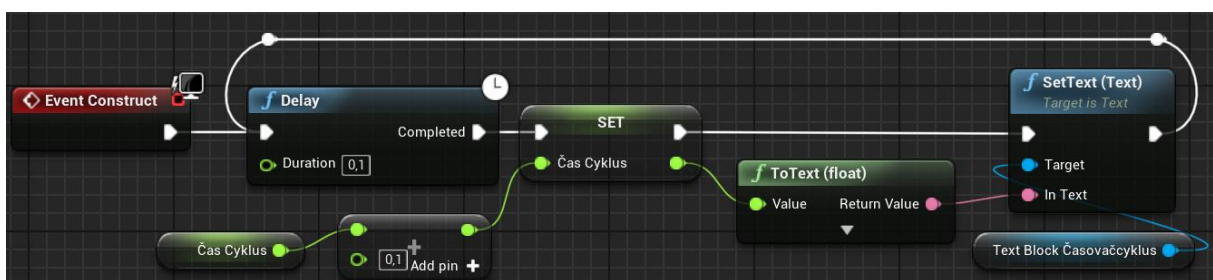


OBRÁZEK 3 UKÁZKA KÓDU Z ÚROVNĚ V PŘÍKLADU Z KAPITOLY 2.6.2

Tento objekt je do úrovně přidán skrze třídu věnující se vlastnostem úrovně. „Event BeginPlay,, je událost, která je zavolána ve chvíli, kdy jsou v úrovni spuštěni všichni Actors. Po zavolání událost zařídí vytvoření instance třídy „Widget Blueprint“ za pomoci uzlu „Create Widget Blueprint Widget“ pro konkrétního uživatele za pomoci uzlu „Get Player Controller“ (definování přiřazeného uživatele je nutné z důvodu možnosti systému pracovat přes internet pomocí modelu klient-server) a následně tento objekt přidá do uživatelovy obrazovky pomocí uzlu „Add to Viewport“.

Připravil jsem si dva příklady, které demonstrují možnosti počítání času. Oba příklady jsou vytvořeny ve třídě „Widget Blueprint“

## První verze kódu počítání času

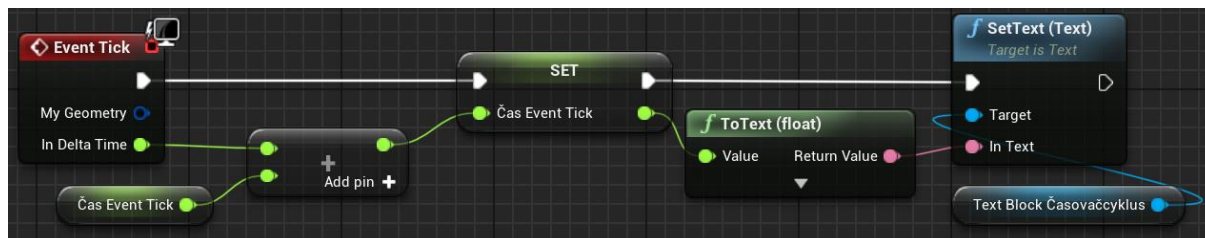


OBRÁZEK 4 POČÍTÁNÍ ČASU PŘÍKLAD 1. V PŘÍKLADU Z KAPITOLY 2.6.2

V prvním příkladu je využita událost „Event Construct“, která je zavolána po vytvoření objektu, ve které se událost nachází. Vytvořil jsem zde proměnnou typu float „Čas Cyklus“, která má nastavenou defaultní hodnotu na 0. Následně nastane čekání 0,1 sekundy díky uzlu „Delay“. Po této době se k aktuální hodnotě proměnné „Čas Cyklus“ připočítá 0,1 na což je výsledná hodnota poslána jakožto text do Text bloku „Text Block Časovačcyklus“. Konec kódu události je propojen s jeho počátkem, díky čemuž nastane nekonečné opakování.

Tento příklad je názorný pro žáky ve chvíli, co vyučující vysvětluje cykly, je však pro systém výpočetně složitější a v měření času nepřesnější než příklad následující.

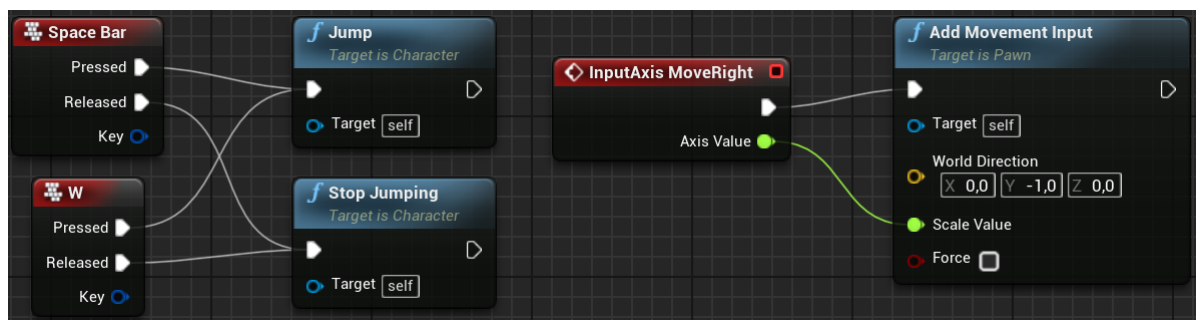
## Druhá verze kódu počítání času



OBRÁZEK 5 POČÍTÁNÍ ČASU PŘÍKLAD 2. V PŘÍKLADU Z KAPITOLY 2.6.2

Ve druhém příkladu je využita událost „Event Tick“, která je volána po každém novém snímku obrazovky. Frekvence volání události je tudíž stejná jako obnovovací frekvence obrazovky použitá v prostředí. Tato událost poskytuje hodnotu typu float „In Delta Time“, jejíž velikost je rovna době, která uběhla od předchozího zavolání této události. Díky této vlastnosti není potřeba kód cyklit, ani využívat vyčkávací funkce.

## Ovládání postavy



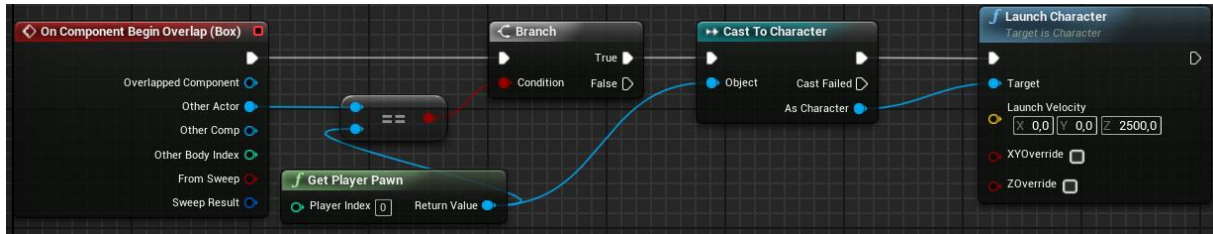
OBRÁZEK 6 KÓD OVLÁDAJÍCÍ POSTAVU V PŘÍKLADU Z KAPITOLY 2.6.2

Tento kód je umístěn ve třídě Actor „Postava“. Skákání postavy je zajištěno pomocí událostí „Space Bar“ a „W“, které jsou volány z vrchního výstupu po stisknutí a ze spodního výstupu po puštění příslušné klávesy. Následně příslušné uzle buďto zajistí výskok nebo konec výskoku postavy. Jednou z vlastností systému Blueprint je možnost využívání stejných uzlů jako reakce na více událostí tak, jak můžete vidět na obrázku 6 vlevo.

Pohyb postavy doprava a doleva je zajištěn pomocí události „InputAxis MoveRight“ která je volána stejným způsobem, jako událost „Event Tick“. Rozdílem těchto událostí je v tom, že při stisknutí klávesy A poskytuje hodnotu -1, při stisknutí klávesy D poskytuje hodnotu 1 a v jiných případech poskytuje hodnotu 0. S touto hodnotou následně pracuje další uzel, který zajistí

příslušný pohyb postavy. Událost „InputAxis MoveRight“ spadá do kategorie „Axis Mapping“. Tyto události jsou vytvořeny uživatelem a lze je do určité míry modifikovat.

## Způsob vymrštění postavy při kontaktu s plošinou

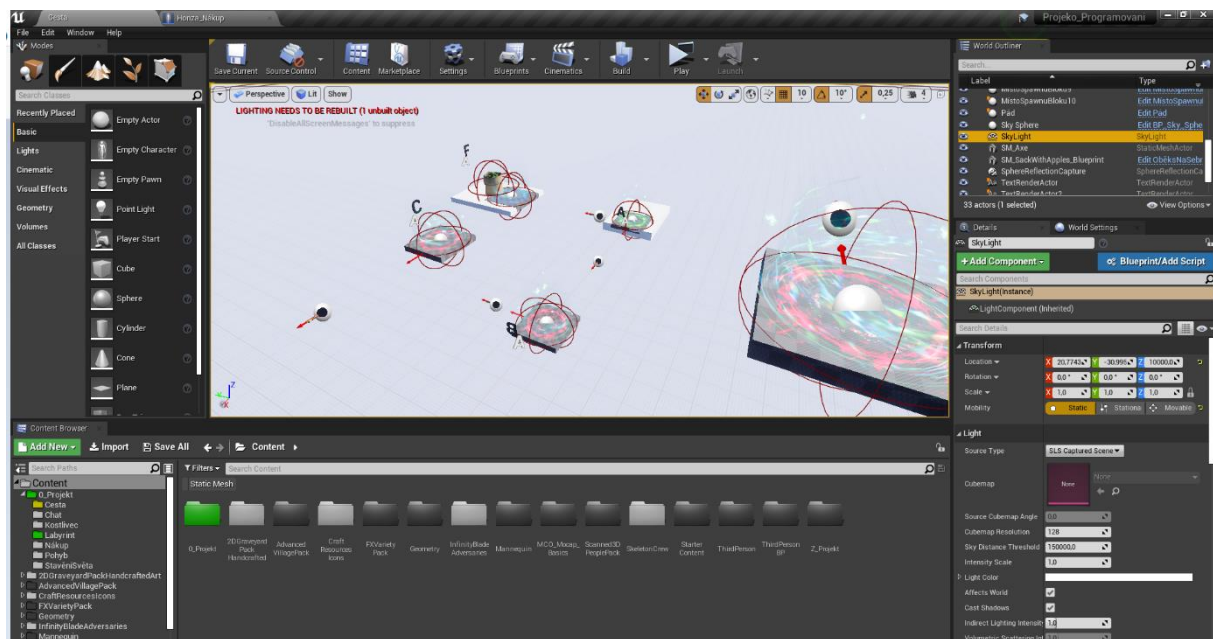


OBRÁZEK 7 KÓD PLOŠINY V PŘÍKLADU Z KAPITOLY 2.6.2

Tento kód je umístěn ve třídě Actor „Plošina“. Událost „On Component Begin Overlap (Box)“ je provázána s komponentou „Box Collision“, která je připojena k Actor „Plošina“ a je volána ve chvíli, kdy dojde ke srážce s jiným Actor, přičemž definici srážky lze modifikovat. Tato událost poskytuje informace o srážce. Jednou z informací je i odkaz na Actor, který kolizi aktivoval. Kód díky této informaci vyhodnotí, zda kolizi způsobila hraná postava díky uzlu „Branch“. Pokud je výsledek „True“, postavě je změněno zrychlení v ose Z pomocí uzlu „Launch Character“.

## 2.5. Orientace v UE4

UE4 má bohatou dokumentaci všech svých částí, která je k přečtení na adrese: <https://docs.unrealengine.com/>. Na obrázku 8 vidíte vzhled prostředí po otevření souboru.



OBRÁZEK 8 VZHLED PROSTŘEDÍ UE4

Ve spodní části obrazovky se nachází adresářová část, ve které jsou umístěny všechny komponenty, jež jsou využívány k fungování projektu (třídy, objekty, úrovně, materiály, zvuky, ...) Lze si zde zvolit, ve které komponentě se chce v danou chvíli pracovat.[6]

Uprostřed obrazovky se nachází okénko, které slouží k zobrazení a interakci s úrovní prostředí, která je aktuálně zvolená (viz předchozí odstavec). Pohybovat se v tomto okénku lze stisknutím pravého tlačítka myši zároveň s klikáním kláves W S A D, či samotným pohybem myši. Do tohoto okénka lze také z adresářové části vkládat jednotlivé Actors (speciální objekty) a dále s nimi pracovat pomocí okénka popsáno v následujícím odstavci.[6]

Pravá část obrazovky je věnovaná Actors umístěným v dané úrovni. Je rozdělena do dvou okének. Vrchní okénko se věnuje výčtu všech Actors použitých v dané úrovni. Jednotlivé Actors v něm lze vyhledávat, mazat, přejmenovávat, či jinak upravovat. Ve spodní části se zobrazují informace o aktuálně zvoleném Actor, jako je například jeho lokace, rotace a velikost v úrovni, či materiál, jenž je na tohoto Actor nanesen. Téměř všechny informace z tohoto okénka lze upravovat klikáním na ně.[6]

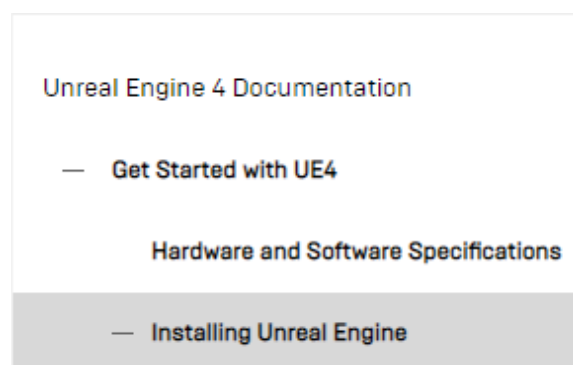
Levá část obrazovky obsahuje nástroje k editaci úrovně, které pro potřeby této práce není nutné rozvádět.[6]

Vrchní část se věnuje projektu a úrovni obecně. Lze zde spustit simulaci projektu, či nastavit výchozí nastavení projektu jako je například platforma, pro kterou je projekt určen.[6]

Pozn. V mé práci žáci pracují pouze se spodní, prostřední a vrchní částí. Prostřední část je použita velice okrajově a vrchní část pouze pro účely simulace.

## 2.6. Instalace UE4

Pro instalaci prostředí Unreal Engine 4 je nutné vytvoření účtu na platformě Epic Games. Tato platforma je určena jednak k distribuci her (obdobně jako platforma Steam na společnosti Valve) a jednak zajišťuje zázemí pro vývojáře využívající UE4. Zajišťuje mimo jiné aktualizaci prostředí, výukové materiály, obchod s různými předpřipravenými částmi, které lze importovat do projektu, a přehled



OBRÁZEK 9 UKÁZKA ROZLOŽENÍ DOKUMENTACE UE4

projektů, s nimiž uživatel na daném zařízení pracoval.[6]

Podrobný návod k instalaci Epic Games a následně prostředí Unreal Engine 4 je na adrese: <https://docs.unrealengine.com/> na tlačítku Installing Unreal Engine viz obrázek 9.

## 3. Porovnání UE4 a jiných systémů ve výuce programování

V této kapitole provedu porovnání prostředí UE4, Scratch, a MakeCode. Pro prostředí Scratch jsem zvolil z důvodu vysokého využívání na školách při výuce programování. MakeCode jsem zvolil pro jeho prvotní orientaci na možnost programování mikropočítače MICRO:BIT za pomoci grafického prostředí.

Prostředí budu porovnávat podle několika kritérií. Mezi mnou zvolená kritéria patří způsob zápisu kódu, intuitivnost systému, práce s proměnnými a objekty a možnosti testování kódu.

Ověřování těchto kritérií jsem uskutečnil prozkoumáním jednotlivých systémů a konceptů, které se při výuce používají. Vzhledem k tomu, že se mi o použití UE4 v českém školství nepodařilo najít žádné informace, provedl jsem porovnání pouze ze své vlastní zkušenosti.

Výsledky porovnávání jsou subjektivní, vychází z mého studia těchto prostředí a z mé osobní zkušenosti s nimi.

### 3.1. Scratch

#### Zápis kódu

Velkou výhodou Scratch je systém bloků, jenž zastupují příkaz v textových programovacích jazycích. Tyto bloky lze následně zapojovat na sebe, čímž vznikají části kódu. Díky tomuto systému je výsledný kód dobře přehledný, pro nezkušeného na první pohled pochopitelnější. Zároveň tím odpadá problém se syntaktickými chybami. Ovládání je zde též jednoduché, neboť jednotlivé bloky lze přesouvat pomocí myši podobně jako ikony na ploše Windows.[7]

Nevýhoda prostředí je nemožnost zobrazení jen určitých uzlů, díky čemuž by mohli být žáci vedeni k vyřešení určité situace pouze za použití uzlů vhodných pro danou úlohu.

#### Intuitivnost systému

Silnou stránkou prostředí Scratch je i rozložení v obrazovce. Zatímco uprostřed obrazovky je vytvářen kód pomocí bloků, které jsou vybírány z přehledných seznamů umístěných v levé části obrazovky, tak v pravé části obrazovky lze v reálném čase sledovat důsledky kódu.[8]



## **Práce s proměnnými**

Vlastností prostředí Scratch je představení proměnných bez datového typu. To se může zdát jakožto nevýhoda, na druhou stranu je díky této vlastnosti prostředí pro žáka jednodušší. Je poté na každé instituci, zda se jedná o výhodu či nikoliv.

## **Práce s objekty**

Jednou z nevýhod prostředí Scratch je nemožnost využívat funkce tříd, jako je například dědičnost. V prostředí se totiž nacházíme pouze na úrovni objektů, které po přidání do projektu jsou ihned zapojeny do dění celku.

## **Testování kódu**

Dle mého názoru nejsilnější stránkou prostředí Scratch je však možnost testovat části kódu bez nutnosti zahájení simulace. Díky tomu není při testování potřeba části kódu připojovat k událostem, které by je spustily. Díky této výhodě lze koncipovat výuku mnohdy zajímavější formou než zapojováním částí kódů za události.[8]

## **3.2. MakeCode**

Na rozdíl od Scratch byl MakeCode navržen na programování MICRO:BIT, což je mikropočítač s jednoduchou led obrazovkou a několika tlačítky. To umožňuje žákům pozorovat důsledky svého kódu i v reálném světě.[9]

### **Zápis kódu**

Prostředí MakeCode, obdobně jako Scratch, využívá systém bloků, které lze zapojovat do sebe, čímž vznikají části kódu. Kromě něj je možné využít i jazyk JavaScript.

V prostředí MakeCode lze v kteroukoliv část vytváření kódu přepínat mezi programováním s pomocí bloků a programováním v JavaScriptu. Díky tomu je možné vidět, jak by vypadaly stejné části kódu v textové formě.[9]

### **Intuitivnost systému**

Rozložení na obrazovce je velice podobné jako u Scratch. Pravá část obrazovky je věnována vytváření kódu, pomocí bloků z přehledných seznamů v levé části obrazovky blíže ke středu. V levé části obrazovky je možné sledovat důsledky kódu.

Seznamy bloků jsou zde navíc rozděleny podle obtížnosti kódu na dvě části. První část je zobrazena defaultně a obsahuje bloky jako je třeba blok „opakuj“, „když“, matematické bloky, vstupy a výstupy zařízení apod. Druhá část je skryta pod tlačítkem „Rozšířené“ a obsahuje bloky pro vytvoření a správu polí, předpřipravené bloky pro vytváření hry, tvoření funkcí apod.

### **Práce s proměnnými**

Proměnné jsou zde též jako v Scratch bez datového typu. I práce s nimi je velice obdobná.

### **Práce s objekty**

Objekty jsou v prostředí MakeCode možné vytvořit skrze JavaScript.[8] Nepodařila se mi dohledat žádná zmínka o tom, že by bylo možné pouze skrze grafickou část prostředí vytvářet nové objekty.[10]

### **Testování kódu**

Prostředí podporuje testování kódu pomocí simulace. Je proto nutné části kódu umisťovat do příslušných událostí.

## **3.3. Unreal Engine 4**

Prostředí UE4 nebylo prioritně zamýšleno k výuce programování, ani se mi nepodařilo dohledat, že by bylo v České republice k výuce programování použito. Obsahuje však možnosti, díky nimž jsem přesvědčen, že by prostředí UE4 mohlo být pro výuku použito.

### **Zápis kódu**

System Blueprint je systém vizuálního skriptování, díky čemuž získává výhody oproti výuce programování v textovém programovacím jazyce. Díky tomu, že systém Blueprint je stvořen pro vývojáře jakožto alternativa jazyka C++, obsahuje spoustu vlastností, které lze využít i nad rámec běžné výuky. Dále díky tomu, že koncept systému Blueprint je založen na principu vývojového diagramu, jsou otevřeny nové způsoby, jak žáci mohou chápat programování.[6]

Další velká výhoda je možnost zobrazení jen určitých uzlů, díky čemuž mohou být žáci lépe vedeni k vyřešení určité situace pouze za použití uzlů vhodných pro danou úlohu.

Velkou nevýhodou oproti prostředím z předchozích kapitol je neošetřenost prostředí proti logickým chybám. Může se proto stát, že při testování nebude program fungovat správně,

což může vést až k restartu UE4. Další velká nevýhoda je množství možností, které lze v UE4 krom programování využívat (př. možnosti úpravy úrovně).

### **Intuitivnost systému**

Nevýhodou prostředí pro výuku programování je to, že UE4 je prioritně určen pro velké projekty nepedagogického směru, díky čemuž je do systému implementováno mnoho nástrojů, které žák nevyužije. Díky tomu se snižuje přehlednost prostředí.

Další nevýhoda spočívá v rozměrech uzlů, díky kterým může nastat situace, že žák neuvidí celý svůj kód najednou a bude nucen za pomoci myši posouvat kód do stran.

Výhodou však je, že v každém momentu, co je otevřen zdrojový kód nějakého Actor, mohou být přehledně zobrazeny všechny proměnné a uzly, jež žák potřebuje k vyřešení daného úkolu.

### **Práce s proměnnými**

V UE4 se pracuje se standartními datovými typy. Tyto proměnné mají každá svou přidělenou barvu, díky níž není nutné po žácích chtít, aby si pamatovali názvy jednotlivých datových typů. Takto barevně se odlišují i samotné spoje v kódu, což zlepšuje orientaci v kódu a mohlo by napomáhat lepšímu pochopení významu datových typů.[6]

### **Práce s objekty**

Systém podporuje plnohodnotnou práci s Objekty, jako je například práce s třídami. To umožňuje například použití stejné postavy pro více úloh.

### **Testování kódu**

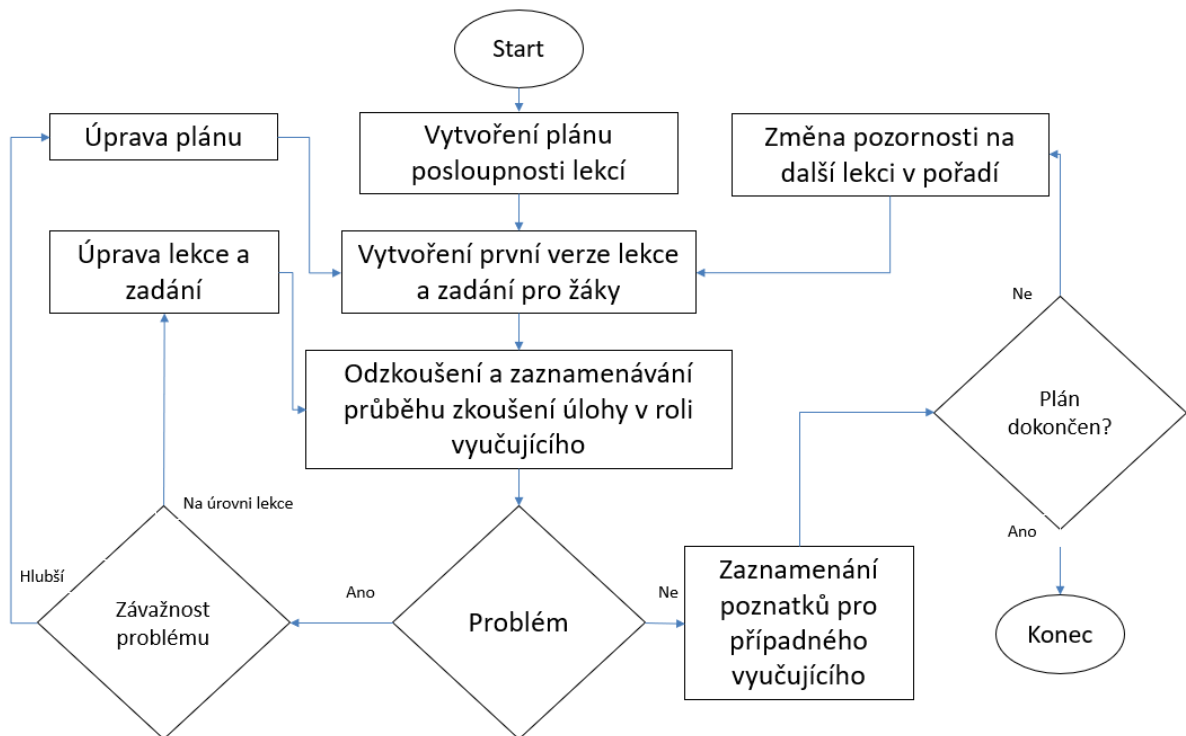
Nevýhodou oproti Scratch však je, že nepodporuje testování částí kódu bez spuštění simulace a je proto nutné jako v případě MakeCode využívat k testování příslušné události.

## 4. Postup tvorby programovacích úloh

Svou sadu úloh jsem vytvářel na základě konceptů výuky programování, jež jsem studoval a uzpůsobil ji pro potřeby prostředí UE4.

Při vývoji své sady úloh jsem postupoval za použití procesu ADDIE. Obrázek 10. blíže popisuje schéma, podle kterého jsem postupoval.

Následující kapitoly jsou věnovány rozboru dílčích kroků.



OBRÁZEK 10 METODYKA PŘÍPRAVY LEKČÍ

### 4.1. Plán posloupnosti lekcí

#### 4.1.1. Tvoření plánu

Při vytváření plánu mi pomohla pravidla, jako je například co nejdříve umožnit tvorbu programů, nebo nepoužívat prvky jazyka, které ještě nebyly vysvětleny, či nevytvářet úlohy, jejichž řešení se musí realizovat za pomoci přílišného množství pomocných operací.[11]

Nedodržení těchto postupů u první verze svých úloh se projevilo při testování, neboť se v mých úlohách časem začaly objevovat části kódu, které žáky mátlý, neboť ještě nebyly vysvětleny, či

realizace úloh vyžadovala obrovské množství uzlů a použití schopností, přesahující možnosti žáků základní školy.

Z toho důvodu jsem hledal přístup, který byl založen na těchto principech. Tento přístup jsem našel na webu IT network v kategorii základní konstrukce jazyka C#, odkaz: <https://www.itnetwork.cz/csharp/zaklady>. Zde jsem se setkal se systémem založeným na paralelním výkladu určitého prvku programování a tvorbou samotného programu, využívající vykládané prvky. Zároveň jsem zde měl k dispozici i sled prvků, ve kterém bych mohl vytvářet své úlohy. Problémem bylo, že kurz z této stránky nebyl primárně určen pro žáky základní školy, což se projevovalo na tom, že sled úloh obsahoval například pole, či třídy, což nejsou témata, která bych chtěl ve svých úlohách probírat.[12]

Z toho důvodu jsem hledal alternativní přístup k posloupnosti, který jsem objevil v učebnici „Programování ve Scratch pro 2. stupeň základní školy“. Učebnice obsahovala nejen pěkně zkonstruovanou posloupnost prvků programování pro výuku, ze které jsem se inspiroval, a terminologii, kterou jsem z části převzal (například bloky-uzly a kód-scénář). Obsahovala i mnoho rad, jakým způsobem lze vést samotnou výuku, například nesdělovat žákům správná řešení, či ponechat žáky pracovat ve dvojicích.[8]

#### **4.1.2. Popis plánu**

Z poznatků získaných prostudováním těchto a jiných literatur jsem vytvořil systém, ve kterém jsem sadu rozdělil do 8 částí. Každá část se věnuje naučení jednoho prvku programování s tím, že využívá i prvků z předešlých částí. Díky tomu v průběhu práce na této sadě žák nejen objevuje nové prvky, ale zároveň je propojuje s těmi předešlými, čímž si utváří obrázek o tom, jak programování funguje jako celek.

Každá část obsahuje dvě až tři lekce. Každá lekce se věnuje nějaké konkrétní situaci. Příkladem by mohla být lekce „Dáma v nesnázích“, která se věnuje tomu, že postava bez kódu defacto nic neumí a je jí nutné naprogramovat, aby začala například chodit. První lekce každé části se z pravidla věnují uvedení nového prvku programování, jeho vysvětlení a ukázkou jeho možné aplikace v kódu. Další lekce mohou v případě potřeby prohloubit znalosti daného prvku, ale většinou slouží spíše pro trénování daného prvku.

### 4.1.3. Osnova částí

Část 1. Sestavování scénáře

Část 2. Opakování bloků

Část 3. Vlastní bloky

Část 4. Vstupy a výstupy

Část 5. Podmínky

Část 6. Parametry

Část 7. Volané události

Část 8. Proměnné

## 4.2. Vytváření lekcí.

Při vytváření samotných lekcí sem se snažil uplatnit takový přístup, aby si žáci nový poznatek nejprve sami vyzkoušeli, následně se zamysleli nad tím, jak by daný poznatek mohl fungovat, a nakonec buďto sami přišli na funkci poznatku anebo v kombinaci s jejich přemýšlením byl poznatek vyložen.

Ve většině lekcích je provedena další dělba podle toho, jakou aktivitu žák právě vykonává.

Některé lekce jsou vytvořeny způsobem, kde je žákovi ukázán problém, na jehož vyřešení je potřeba několik kódů, konajících podobnou činnost. Na prvních částech je žákovi představen buďto nový prvek nebo jeho možné aplikace. Zbylé části žák zhotovuje sám formou procvičování. Příkladem takovéto lekce je lekce 1. Cesta v části 1., ve které žák postupně dostává úkoly vytvořit cestu z plošiny A na B, B na C, C na D, D na E a E na F.



OBRÁZEK 11 VÝCHOZÍ STAV SIMULACE U LEKCE 1. CESTA

Na přechodu z plošiny A na B je demonstrováno zapojování uzlů k událostem s tím, že žáci v tuto chvíli událost vnímají jakožto takový „Start kódu“. Na přechodu z plošiny B na C žáci zopakují, co se naučili v předchozí části. Na přechodu z plošiny C na D se demonstruje možnost zapojování více uzlů za sebou. Následné přechody dále procvičují tyto poznatky.

### **4.3. Testování lekcí a vytváření podkladů pro vyučující**

Po vzoru postupů popsaných v předchozích kapitolách jsem vytvořil první verzi posloupnosti úloh. Následně jsem vytvořil několik prvních úloh, které jsem testoval jakožto vedoucí kroužku s žáky 2. stupně základní školy. Žáci pracovali ve dvojicích, což pozitivním způsobem zlepšovalo diskuzi žáků.

Testování probíhalo způsobem jedna úloha testovaná za jednu vyučovací hodinu. Po otestování dané úlohy jsem si sepsal poznatky o průběhu testování jako je například čas, za který byla většina dětí lekci schopná dodělat, části úloh, které nebyly vhodně koncipovány či vhodně zadány, nebo obecné poznatky, které jsem získal a využil pro konstrukci podkladů pro učitele.

Vzhledem k tomu, že má první verze úloh měla nedostatky jako jsou například špatně sestavené posloupnosti úloh a jejich konceptu, byl jsem nucen svůj koncept úloh přepracovat. Vytvořil jsem tedy 2. posloupnost úloh. Tuto posloupnost jsem rozdělil na části a následně na lekce. Lekce jsem testoval stejným způsobem jako úlohy v první verzi.

U tohoto konceptu se výrazné problémy neobjevovaly (objevovaly se jen na úrovni lekce). Pokud jsem nějaký objevil, opravil jsem jej. Následně jsem takto opravenou lekci nechal otestovat žáky, kteří zpravidla chyběli předchozí hodiny. Takto jsem postupoval až do chvíle, kdy jsem otestoval všechny lekce, které jsem měl v zásobě.

Kroužku se účastnilo v průměru jedenáct žáků. Nejvíce jich bylo sedmnáct, a to druhou vyučovací hodinu. Nejméně žáků bylo pět, a to z organizačních důvodů školy, na které jsem kroužek vedl. Všechny úlohy úspěšně dokončilo šest žáků. Zásadní vliv na počet dokončení měla občasná neúčast žáků na hodinách a především vyhlášení karantény. Čtyřem žákům by stačily k dokončení 1-2 hodiny výuky. Celkem bylo realizováno 22 hodin výuky. Několik hodin bylo věnováno první verzi zadání, která již neexistuje. Výraznější problémy s druhou verzí úloh se objevovaly povětšinou v prvních lekcích prvních částí. Ke konci již žáci pracovali bez větších problémů.

## 5. Sada úloh

Tato kapitola obsahuje zadání pro žáky spolu s poznámkami pro vyučujícího. Kapitola je vytvořena jakožto podpora vyučujícího při výuce.

### 5.1. Úvodní poznatky

#### 5.1.1. Podpůrné materiály

Na webové adrese: <https://sites.google.com/view/vyuka-programovani-v-ue4>, nebo také jakožto příloha na paměťové kartě, která je přiložen k této práci, se nachází následující položky:

1. Pracovní listy pro žáky  
-obsahují zadání pro žáky
2. Projekt\_Programování (na webu zkomprimováno do jednoho archivu .7z )  
-složka souborů, obsahující projekt UE4, ve které žáci plní úkoly ze svého zadání
3. Projekt\_Programování\_Vzor (na webu zkomprimováno do jednoho archivu .7z )  
-složka souborů, obsahující projekt UE4, ve které jsou vytvořeny vzorové scénáře jednotlivých lekcí
4. Kopie této práce

#### 5.1.2. Technické instrukce

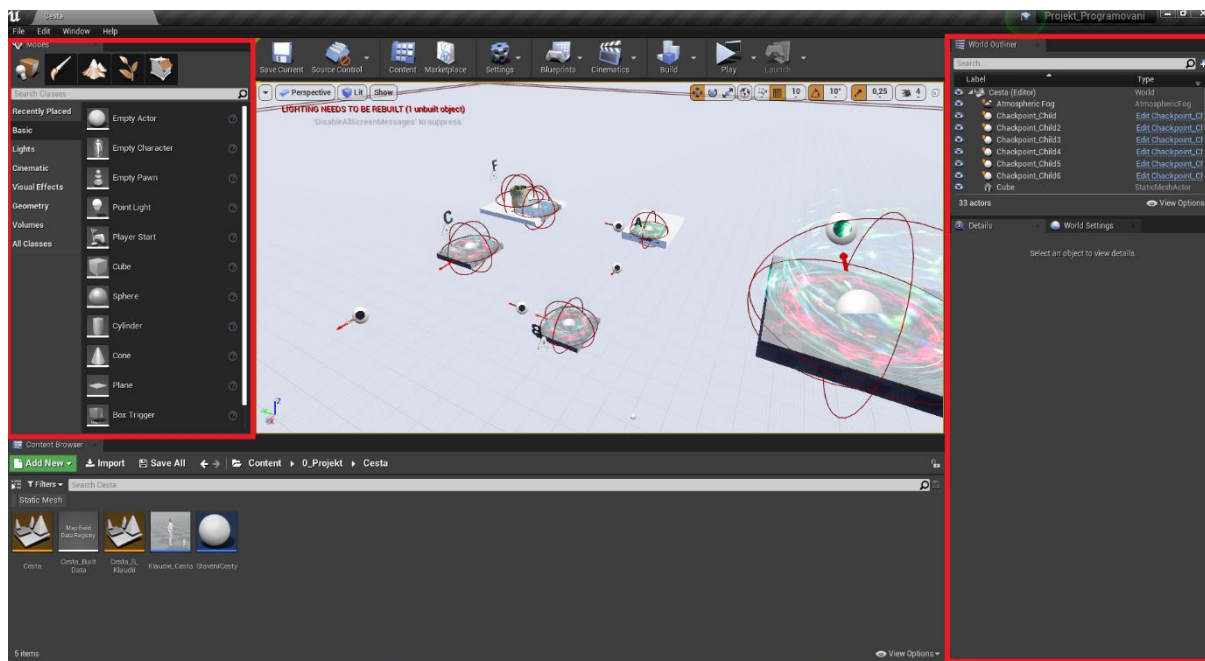
Před spuštěním projektu UE4 je nutné nainstalovat prostředí UE4 verze nejméně 4.24.0.

Pro spuštění projektu UE4 je nutné otevřít složku tohoto projektu a následně spustit soubor nesoucí název tohoto projektu v prostředí UE4. Projekt se bude zapínat 1-3 minuty v závislosti na vytížení zařízení.

Po prvním spuštění projektu je vhodné jej 5-15 minut nechat spuštěný a nepracovat s ním. Program v tuto chvíli zpracovává své části, jako jsou stíny, materiály či objekty. Při následujícím spuštění projektu jsou tyto části již zpracovány.

Doporučuji zmenšit okénka v pravé a levé části hlavní stránky prostředí. Tato okénka během výuky nebudou použita. Jejich zmenšením se docílí lepší přehlednosti pro žáky. Zmenšení provedete posunem myši na hranici okénka, které chcete zmenšit, a okénka v prostřední části obrazovky. Následně zatáhněte tuto hranu do strany monitoru.





OBRÁZEK 12 OKÉNKA VHODNÁ PRO ZMENŠENÍ

### 5.1.3. Poznámky k výuce

Při výuce je vhodné žáky rozdělit do dvojic. Ve dvojicích je totiž pozitivně zlepšována diskuze žáků nad daným problémem.

Lekce jsou vytvořeny tak, aby v nich žáci mohli postupovat bez zásahu vyučujícího (pokud není v poznámkách pro danou lekci, či podkapitolu řečeno jinak). Pozice vyučujícího v tomto kurzu je koncipována nikoliv v roli předavače znalostí, ale v roli tutora (motivační, organizační, konzultační, podpora žáka).

Žákům je třeba vysvětlit, že odrážky s černým puntíkem obsahují činnost, kterou mají žáci vykonat. Až následně mohou přejít k další odrážce. Odrážky s bílým puntíkem obsahují rady, popisy, či vysvětlují danou problematiku. Žáci po přečtení mohou pokračovat dále.

Orientační časové dotace u jednotlivých lekcí jsou pouze vodítkem pro přibližnou délku výuky. Vyučující může délku výuky upravit dle potřeby žáků.

Jednotlivé skupiny by neměly být (kromě soutěží) mezi sebou poměřovány. Pomalejší skupiny by totiž mohly být demotivovány k postupu v lekcích. To souvisí i s přístupem vyučujícího k jednotlivým skupinám. Vyučující by neměl dávat na vědomí své zaujetí vůči určitým skupinám.

## 5.2. Část 1. Sestavování scénáře

### Cíl výuky

- Žák dokáže spustit projekt vytvořený v UE4
- Žák dokáže sestavit scénář, jímž splní zadaný úkol
- Žák dokáže scénář upravovat mazáním bloků, či změnou jejich pořadí
- Žák dokáže spustit simulaci a ověřit správnost svého scénáře

### Témata lekcí

- Stavění cesty mezi plošinami s cílem zajistit možnost přejít z plošiny A na plošinu F
- Vytváření jednoduchých scénářů, které rozpohybují postavu
- Vytvoření scénáře zajišťující přesun postavy z plošiny A na plošinu F

#### 5.2.1. Lekce 1. Cesta

Orientační časová dotace: 30 minut

Využitá úroveň: Cesta

Lekce představuje postavu Emanuela, jehož žáci mohou během simulace ovládat. Emanuel se potřebuje dostat z plošiny A přes plošiny B, C, D, E na plošinu F. K tomu je zapotřebí vytvořit cesty mezi těmito plošinami.

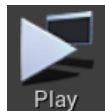


OBRÁZEK 13 UKÁZKA ÚROVNĚ CESTA

## Jednotlivé aktivity

### PRVNÍ KROKY

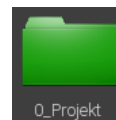
- Spust' program „Projekt\_Programování“. Cestu k tomuto programu ti řekne vyučující
- V levém spodním rohu uvidíš zelenou složku „0\_Projekt“. Otevři ji
- Nyní otevři složku „Cesta“, a v ní spust' soubor „Cesta“
- Stiskněte tlačítko „Play“ vpravo nahoře



OBRÁZEK 16  
TLAČÍTKO PLAY



OBRÁZEK 15  
SOUBOR CESTA



OBRÁZEK 14 SLOŽKA  
0\_PROJEKT

- Nyní se ti otevřelo nové okénko, ve kterém ovládáš postavu Emanuela. S Emanuelem můžeš chodit různými směry (po stisku kláves W, S, A, D), nejprve však musíš na toto okénko kliknout. Vyzkoušej to. Obrazovku ukončíš stisknutím klávesy Esc
- V tuto chvíli se nacházíš na plošině A. Emanuel dostal chuť na jablka, která se nacházejí na plošině F. K tomu, aby se k nim dostal, však musí postavit cesty, jež se vždy objeví na bílých kuličkách, které jsou umístěny mezi plošinami
- Stiskni tlačítko „Esc“ pro ukončení okýnka s Emanuelem

### POZNÁMKY PRO VYUČJÍCÍHO

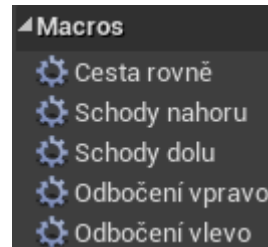
*Podkapitola „První kroky“ je věnována spuštění lekce a následné simulaci.*

*Zde je důležité žákům sdělit umístění projektu, aby jej mohli spustit.*

*Žáci mohou mít problémy spojené s prvotní orientací v prostředí, proto je možné jim ukázat polohu jednotlivých složek, tlačítek a souborů. Většina žáků by však měla být schopna pracovat samostatně.*

## BLOKY

- Nyní si otevři soubor StavěníCesty
- Zde můžeš vidět několik „bloků“, které jsou pojmenované Event Úkol 1, Event Úkol 2, ...
  - V tuto chvíli se zaměř pouze Event Úkol 1
  - Vlevo uprostřed můžeš vidět seznam bloků „Macros“
- Vyber si jeden z nich a přetáhni jej někam do blízkosti bloku Event Úkol 1
  - Nyní se zaměř na zobáčky, jež z jsou v těchto blocích umístěné
- Chytni jeden ze zobáčků a přetáhni jej na zobáček druhého bloku tak, jak to vidíš na obrázku níže



OBRÁZEK 17 SEZNAM MACROS



OBRÁZEK 19 BLOKY V SOUBORU STAVĚNÍCESTY

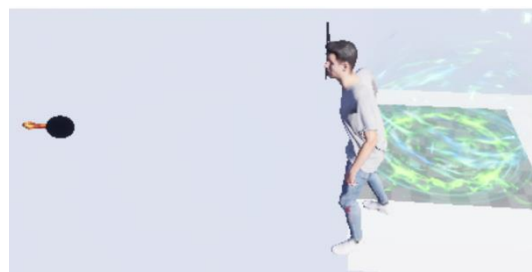


OBRÁZEK 18 PROPOJOVÁNÍ ZOBÁČKŮ

- Nyní stiskni tlačítko Play a vyzkoušej, co se stane
  - Blok smažeš kliknutím na něj a stisknutím klávesy „Delete“
  - Jak nyní vidíš, před tebou se objevila cesta, která má vzhled podle toho, jaký blok jsi zvolil
  - Všimni si také, že tato cesta se postavila na místě a ve směru kuličky s šipkou, která byla mezi plošinou A a B



OBRÁZEK 20 VZHLED ÚROVNĚ PO UMÍSTĚNÍ CESTY



OBRÁZEK 21 VZHLED ÚROVNĚ PŘED UMÍSTĚNÍM CESTY

- Vyber správný blok z nabídky a napoj jej k „Event Úkol 1“ tak, abys se dostal/a na plošinu B. (Jakýkoliv blok lze smazat kliknutím na něj a stisknutím klávesy Delete)
- Po vytvoření cesty přejdi na plošinu B
- Nyní se stejným způsobem dostaň z plošiny B na plošinu C (Zapojuj bloky za „Event Úkol 2“)

#### POZNÁMKY PRO VYUČUJÍCÍHO

*V podkapitole „Bloky“ žáci objevují, že lze za pomoci bloků přidat do simulace část cesty, po které mohou přejít.*

*Žáci zde mohou mít problém odhadnout umístění a rotaci jednotlivých cest, jež budou přidávat. V takovém případě je vhodné poukázat na spojitost nové cesty a kuličky s šipkou. Toho se nejlépe docílí tak, že vyzveme žáka, aby smazal cestu, u které nastal problém, spustil simulaci a zaměřil se na orientaci šipky u kuličky.*

*Žáci mohou mít též problém, že při přechodu z plošiny B na C připojují bloky k „Event Úkol 1“. V takovém případě žáky vyzvete, aby si opět přečetli zadání pro stavbu cesty mezi plošinou B a C.*

*Žáci mohou vytvářet cesty, které sice jsou schůdné, ale nepasují do prostředí. V takovém případě je vhodné, aby je vyučující vyzval ke zvolení jiných bloků.*



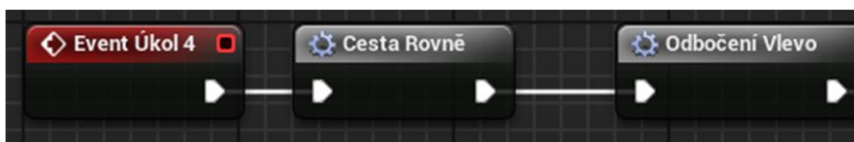
OBRÁZEK 22 SCÉNÁŘ PRO CESTU Z PLOŠINY A NA B



OBRÁZEK 23 SCÉNÁŘ PRO CESTU Z PLOŠINY B NA C

## SCÉNÁŘ

- Mezi plošinou C a D jsou nyní dvě kuličky. To znamená, že ve třetím přechodu je potřeba využít dvou bloků.
- Nyní si vyber dva bloky, u kterých předpokládáš, že by mohly být ty správné, a zapoj je za Event Úkol 3. viz obrázek níže



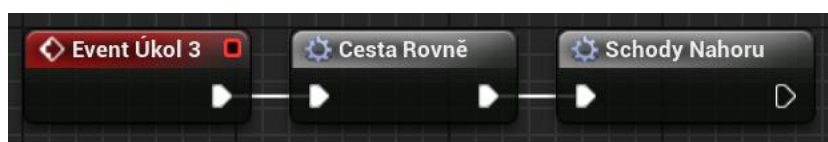
OBRÁZEK 24 SCÉNÁŘ PRO PŘECHOD MEZI PLOŠINOU C A D

- Jak vidíš, tak se v jednom kroku postavily dvě cesty. Takto propojené bloky bílou čarou se označují jako scénář
- Pokud jsi vybral/a špatné bloky, smaž je a vyber jiné tak, abys se dostal/a na plošinu D
- Nyní se dostaň stejným způsobem na plošinu E. Zkontroluj si, kolik bude potřeba bloků
- Na závěr se dostaň k plošině F

## POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Scénář“ se zaměřuje na použití více bloků v jednom scénáři.

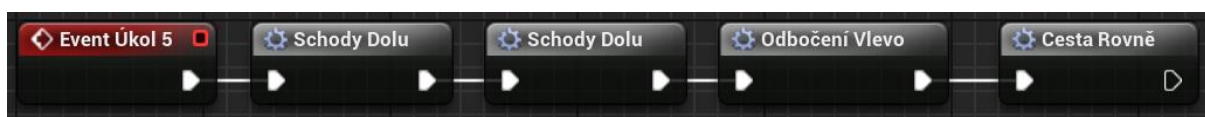
Tato část jde žákům dobře. Problémy mohou nastat při nepozornosti žáků v počítání bloků na přechodu z plošiny E na F. Tento problém jsou však žáci povětšinou schopni vyřešit sami.



OBRÁZEK 25 SCÉNÁŘ PRO CESTU Z PLOŠINY C NA D



OBRÁZEK 26 SCÉNÁŘ PRO CESTU Z PLOŠINY D NA E



OBRÁZEK 27 SCÉNÁŘ PRO CESTU Z PLOŠINY E NA F

## 5.2.2. Lekce 2. Dáma v nesnázích

Orientační časová dotace: 35 minut

Využitá úroveň: Pohyb

V lekci je žákům představena nová postava Klaudie. Tuto postavu již žáci během simulace neovládají a musí její pohyb naprogramovat.

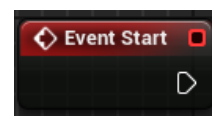


OBRÁZEK 28 UKÁZKA ÚROVNĚ POHYB

### Jednotlivé aktivity

#### NOVÁ POSTAVA

- Otevři složku „Pohyb“, a v ní spust’ soubor „Pohyb“
  - Jak nyní můžeš vidět, kromě Emanuela se na mapě nachází ještě jedna postava jménem Klaudie
- Stiskni tlačítko Play a podívej se na Klaudii zblízka
  - Je zřejmé, že s Klaudií není něco v pořádku. Není totiž naprogramovaná
- Nyní si otevři soubor Klaudie\_Pohyb
  - V souboru s Klaudií je umístěn pouze jediný blok „Event Start“. Tento blok se chová stejně, jakožto bloky Event Úkol 1, Event Úkol 2, ...



OBRÁZEK 29 BLOK  
EVENT START

- Můžeš jej považovat za startovní místo, odkud se spouští všechny následující bloky

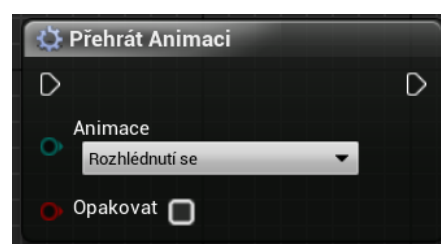
### POZNÁMKY PRO VYUČJÍČIHO

*Podkapitola „Nová postava“ vede žáka ke spuštění lekce. Během spuštění seznamuje žáka s postavou Klaudie, která není nijak naprogramována.*

*Problém může nastat v tom, že žákovi zůstane z minulé lekce otevřená složka Cesta. Převážná většina žáků sama přijde na řešení této situace.*

### ZMĚNA PÓZY

- V prvním kroku zajisti, aby Klaudie neměla takto zaseknutý výraz. Z nabídky bloků „Macros“ vyber „Přehrát animaci“ a otestuj, co tento blok dělá



OBRÁZEK 30 BLOK PŘEHRÁT ANIMACI

- Co se mění při použití tohoto bloku?
- Jaká nastane změna, když zaškrtněš zaškrťovací políčko „Opakovat“?
- Nyní se zaměř na nabídku „Animace“. Otestuj různé volby

### POZNÁMKY PRO VYUČJÍČIHO

*Tato podkapitola představuje blok „Přehrát Animaci“, žák se použitím tohoto bloku dozvídá, jakým způsobem mohou být postavy rozpořehovány.*

*Možné odpovědi na otázky:*

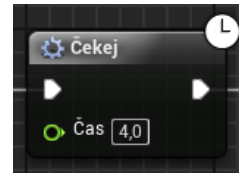
- Co se mění při použití tohoto bloku?
- Jaká nastane změna, když zaškrtněš zaškrťovací políčko „Opakovat“?

### ČEKÁNÍ

- Nyní vyzkoušej dvě animace po sobě. Přetáhni si druhý blok „Přehrát Animaci“ a pokus se zajistit, aby se Klaudie chvíli rozhlížela a poté chodila na místě. Pokud bude Klaudie pouze chodit na místě, pokračuj na další bod



- Přidej mezi bloky nový blok „Čekej“ ze seznamu „Macros“
- Otestuj ho. Změň u něj hodnotu u zobáčku „Čas“ na 7



OBRÁZEK 31 BLOK ČEKEJ

- Jak se tento blok chová?

- Proč Klaudie bez použití bloku „Čekej“ pouze chodila na místě ?

- Povšimni si znaku hodin v pravém horním rohu bloku „Čekej“. Tento znak říká, že následující blok se spustí až ve chvíli, co tento blok dokončí svou činnost (v tomto případě až uběhne čas). Tento znak mají i mnohé jiné bloky.

#### POZNÁMKY PRO VYUČUJÍCÍHO

*Podkapitola „Čekání“ seznamuje žáka s blokem „Čekej“*

*Podkapitola je vytvořena tak, aby žák nejprve udělal chybu, ze které se může poučit. Pokud se žáci dotáží na to, proč jejich Klaudie pouze chodí na místě a nerozhlíží se, přimějte je pokračovat v dalším úkolu této podkapitoly.*

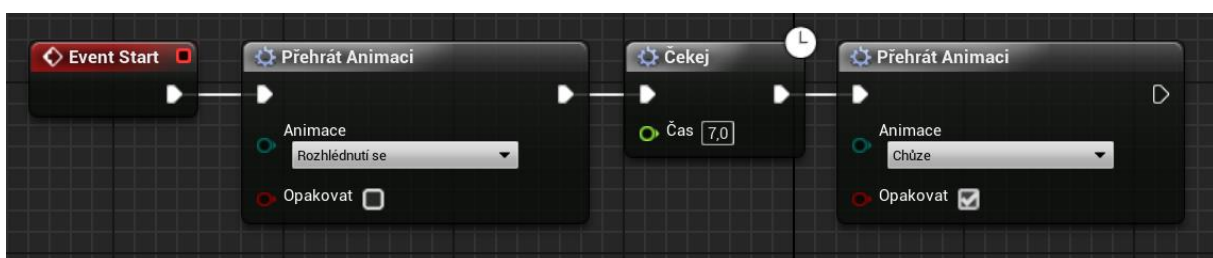
*Možné odpovědi na otázky:*

- *Jak se tento blok chová?*

*Pozastavuje scénář na dobu, kterou nastavíme.*

- *Proč Klaudie bez použití bloku „Čekej“ pouze chodila na místě ?*

*Protože nebylo nastaveno, jak dlouho se má Klaudie rozhlížet.*



OBRÁZEK 32 SCÉNÁŘ PRO ROZHLÉDNUTÍ A NÁSLEDNOU CHŮZI NA MÍSTĚ

## ROZPOJOVÁNÍ BLOKŮ

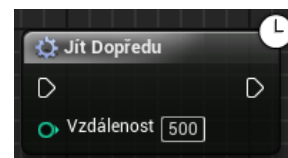
- Odpoj to, co jsi doposud vytvořil/a na bloku „Event Start“. Toho docílíš dvojklikem myši do bílé čáry, jež spojuje “Event Start” s následujícím blokem. Tím vytvoříš na čáře uzel, se kterým lze pohybovat stejně jakožto s bloky. Pokud tento uzel smažeš stisknutím klávesy „Delete“, dojde k odpojení předchozího a následujícího bloku



OBRÁZEK 33 DETAIL ROZPOJENÍ

## POHYB POSTAV

- U některých animací můžeš vidět, že se Klaudie nechová přesně tak, jak by se dalo očekávat (např. při chůzi chodí na místě).
- Nyní zkus Klaudii rozchodit. Přesuň bloky, jež jsi vytvořil/a a odpojil/a někam stranou
- Připoj blok Jít dopředu a otestuj jej. Měň číslo u zobáčku „Vzdálenost“. Použij hodnoty 100, 300, 500
- Klaudie nyní chodí nepřírozeně. Zajisti, aby chodila přirozeně. Využij k tomu blok „Přehrát animaci“
- Nyní zajisti, aby ve chvíli, kdy se Klaudie zastaví, nechodila na místě, ale čekala

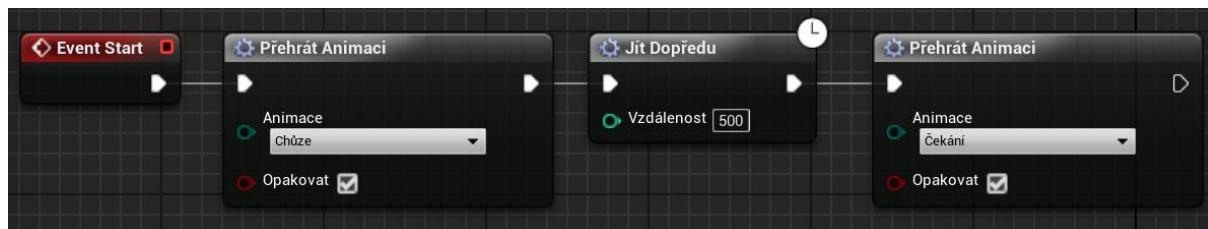


OBRÁZEK 34 BLOK JÍT DOPŘEDU

## POZNÁMKY PRO VYUČUJÍCÍHO

*Podkapitola „Pohyb postav“ se zaměřuje na představení bloku „Jít Dopředu“. Dále poukazuje na použití tohoto bloku v kombinaci s blokem „Přehrát Animaci“.*

*Problémy často nastávají s tím, že žáci vytvoří scénář, ve kterém se Klaudie napřed pohybuje bez animace až po dokončení cesty je animace spuštěna. To je způsobeno tím, že žáci napřed postavou vyšlou dopředu blokem „Jít Dopředu“ a až následně přehrají animaci. Jedná se o úmyslný problém, se kterým by si žáci měli poradit sami.*



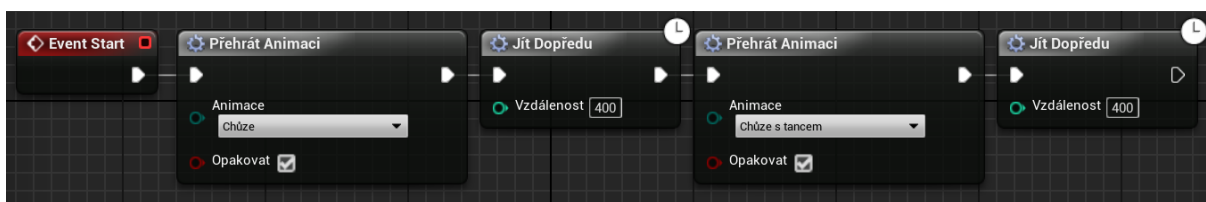
OBRÁZEK 35 SCÉNÁŘ PRO ROZPOHYBOVÁNÍ POSTAVY S NÁSLEDNÝM ZASTAVENÍM

## POKROČILEJŠÍ ÚKOLY

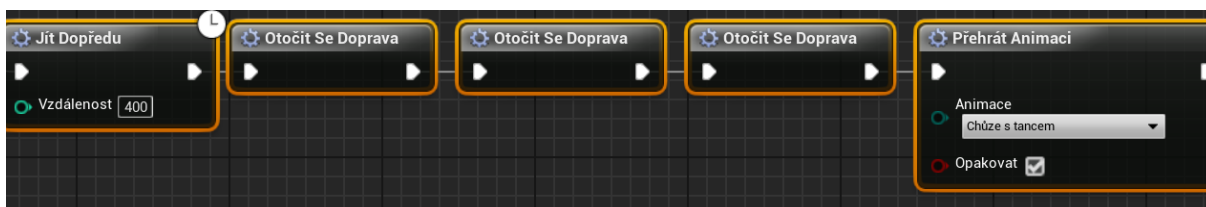
- Tento scénář odpoj a vytvoř nový, ve kterém Klaudie půjde polovinu cesty normálně a polovinu cesty s tancem (vzdálenost 400 a 400). Bloky lze kopírovat klávesami Ctrl+C a Ctrl+V
- Nyní přidej blok „Otočit se doprava“. Použij jej tak, aby se Klaudie v půli cesty otočila
- Zajisti, aby se Klaudie v půli cesty neotočila doprava, ale doleva
- Uprav scénář tak, aby Klaudie šla čtvrtinu cesty normálně, čtvrtinu cesty s tancem, čtvrtinu cesty s rozhlížením se a čtvrtinu cesty šťastně
- Zařid', aby se Klaudie před každou změnou stylu chůze otočila doleva (aby šla do čtverce)

### POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Pokročilejší úkoly“ obsahuje sled úkolů, které žáky procvičují v sestavování scénářů. Úkoly na sebe navazují.



OBRÁZEK 36 SCÉNÁŘ PRO POLOVINU CESTY NORMÁLNĚ A POLOVINU S TANCEM



OBRÁZEK 37 PRINCIP OTOČENÍ DOLEVA

## BONUS

- Na závěr chytni za zobáček u posledního bloku a napoj jej k prvnímu (za blok „Event Start“)
- Sleduj, co se stane

### POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Bonus“ je přidána, jakožto ochutnávka cyklů. Není nutná pro výuku.

### 5.2.3. Lekce 3. Chut' na jablka

Orientační časová dotace: 20 minut

Využitá úroveň: Cesta\_S\_Klaudíí

V této lekci žáci vedou Klaudii po cestě z plošiny A na plošinu F, kterou vytvořili v první lekci



OBRÁZEK 38 UKÁZKA ÚROVNĚ CESTA\_S\_KLAUDÍÍ

#### Jednotlivé aktivity

- Nyní otevři složku „Cesta“, a v ní spusť soubor „Cesta\_S\_Klaudíí“
- Dále otevři soubor „Klaudie\_cesta“
- Tvým úkolem nyní je, aby se Klaudie dostala z plošiny A na plošinu F, neboť Klaudie dostala chuť na jablka a rozhodla se, že je Emanuelovi všechny sní. Vzdálenosti, jež Klaudie musí překonat mezi plošinami, jsou v násobcích 250

#### POZNÁMKY PRO VYUČJÍCÍHO

*S touto lekcí žáci nemívají problém. Někdy je vhodné připomenout žákům, že vzdálenosti, jež musí Klaudie urazit, jsou v násobcích 250.*

## 5.3. Část 2. Opakování bloků

### Cíl výuky

- Žák ví, jak funguje opakování
- Žák pozná, kdy je vhodné použít opakování
- Žák dokáže do scénáře, vložit opakování, pokud jej scénář neobsahuje

### Témata lekcí

- Vytváření předmětů
- Průchod přes překážky
- Vylepšování scénáře pro Klaudii

### 5.3.1. Lekce 1. Záhadná bludička

Orientační časová dotace: 40 minut

Využitá úroveň: Svět

V lekci žáci budou stavět budovy za pomoci postavy Bludička.



OBRÁZEK 39 UKÁZKA ÚROVNĚ SVĚT

## Jednotlivé aktivity

### PROHLÍDKA MAPY

- Nyní otevři složku „StavěníSvěta“, a v ní spust' soubor „Svět“
- Stiskni tlačítko „Play“ a prohlídni si mapu
  - Jak vidíš, Emanuel není na mapě sám. Kousek od Emanuela je záhadná světélkující koule, jež má nadpřirozené schopnosti. Bohužel stejně jako Klaudie neví, jak své schopnosti používat, a tak potřebuje tvou pomoc

### STAVĚNÍ SVĚTA

- Nyní spust' soubor „Bludička“
- Teď otestuj, co dělá blok „Stavba“
- Nyní zajisti, aby po postavení domu z něj bludička vylétla směrem k Emanuelovi (použij bloky „Otočit doprava“ a „Jít dopředu“)

### POZNÁMKY PRO VYUČUJÍCÍHO



OBRÁZEK 40 SCÉNÁŘ PRO POSTAVENÍ DOMU A NÁSLEDNÉ VYLÉTnutí Z NĚJ ZA POUŽITÍ BLOKU OTOČIT SE DOPRAVA

### OTÁČENÍ POSTAVY

- Nyní vyzkoušej to samé, s tím rozdílem, že místo bloku „Otočit doprava“ použiješ blok „Otočit se o“. Tip. Kolik stupňů má kruh ?
- Jak tento blok funguje?
- V čem spočívá jeho výhoda oproti bloku „Otočit doprava“?
- Nyní zajisti, aby se dům postavil vchodovými dveřmi k Emanuelovi. Bludička stále musí dojít k Emanuelovi

### POZNÁMKY PRO VYUČUJÍCÍHO

*V podkapitole „Otáčení postavy“ je představen nový blok „Otočit se o“. V tomto bloku lze nastavit úhel, o který se má postava otočit.*

Žákům může chvílku trvat, než si propojí úhly s otočením postavy. Může být tedy vhodné s žáky diskutovat o tom, co to vlastně úhel je, kolik stupňů má kruh, kolik trojúhelník apod.

Možné odpovědi na otázky:

- Jak tento blok funguje?

Postava se otočí o tolik stupňů, kolik do bloku nastavím.

- V čem spočívá jeho výhoda oproti bloku „Otočit doprava“?

Můžu si nastavit jakýkoliv směr.



OBRÁZEK 41 SCÉNÁŘ PRO POSTAVENÍ DOMU A NÁSLEDNÉ VYLÉTNUTÍ Z NĚJ ZA POUŽITÍ BLOKU OTOČIT SE O

### NEKONEČNÉ OPAKOVÁNÍ

- Nyní propoj poslední zobáček s blokem „Stavba“. (Můžeš si dopomoci vytvořením uzlů na bílé čáře k tomu, aby byl scénář přehlednější tak, jak to vidíš na obrázku)



OBRÁZEK 42 SCÉNÁŘ PRO NEKONEČNÉ STAVĚNÍ DOMŮ VE ČTVERCI

- Co se nyní děje ve světě?

- Jak je funguje tvůj nynější scénář?

- Co by se stalo, kdyby konec čáry nebyl zapojen do bloku „Stavba“, ale do bloku „Otoč se o“ s hodnotou 90. (Napřed přemýšlej, potom ověř)

- Kolik bude v mapě domů po 12. zopakování?

### POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Nekonečné opakování“ představuje možnost opakování stále dokola a snaží se skrze otázky zajistit, aby se žáci nad opakováním zamysleli. Je proto velice vhodné s žáky nad těmito otázkami debatovat.

Možné odpovědi na otázky:

- Co se nyní děje ve světě?

*Ve světě se staví domy do čtverce*

- Jak je funguje tvůj nynější scénář?

*Opakuje se stále dokola*

- Co by se stalo, kdyby ve scénáři nebyl blok „Otočit Se O“?

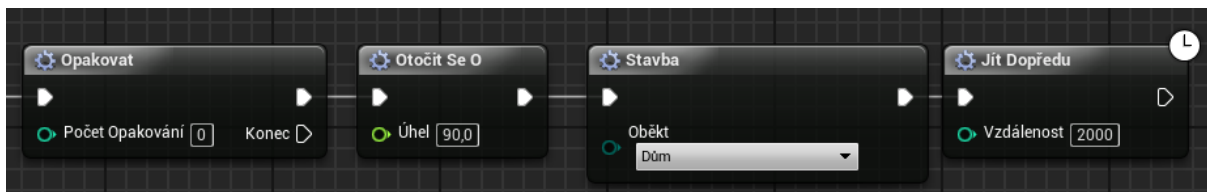
*Domy se staví stále dokola, dokud Bludička nepřepadne za okraj*

- Kolik bude v mapě domů po 12. zopakování?

*12*

### OPAKOVÁNÍ S KONCEM

- Smaž vše, krom bloku „Event Start“
- Nyní se zaměř na blok „Opakovat“. Zpoj jej podobně jako na obrázku níže



OBRÁZEK 43 NEÚPLNÉ ZAPOJENÍ S BLOKEM OPAKOVAT

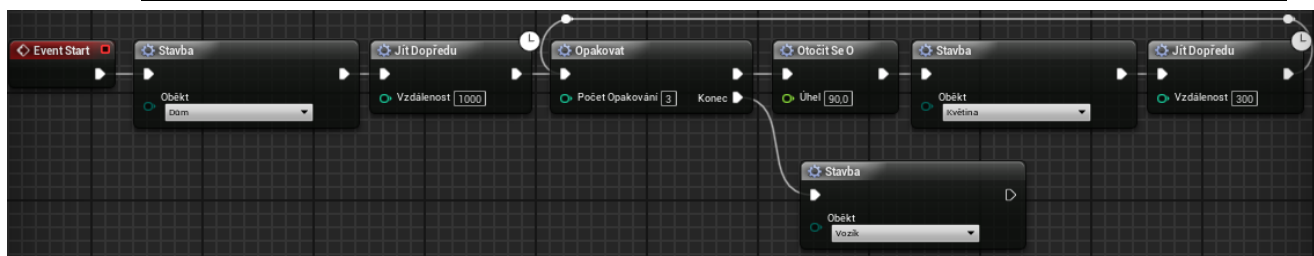
- Jak myslíš, že tento blok bude fungovat?

- Za pomoci bloku „Opakovat“ postav čtyři domy. Aby se zajistilo opakování, je nutné propojit konec části scénáře, která se má opakovat s blokem, který má opakování na starost
- Kdy si myslíš, že se spustí blok připojený na zobáček s nápisem „Konec“?

- Ověř si to nějakým způsobem



- Co by udělal scénář, který vidíš na obrázku níže? (Napřed přemýšlej, poté ověř)

OBRÁZEK 44 SCÉNÁŘ ZADANÝ ŽÁKŮM K ANALÝZE

### POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Opakování s koncem“ představuje nový blok „Opakovat“ a poukazuje na způsob jeho fungování.

Zde je vhodné opět debatovat s žáky nad zadanými otázkami.

Žáci mohou mít problém přijít na to, že pro správnou funkci bloku „Opakovat“ je nutné propojit konec části scénáře, která se má opakovat s blokem „Opakovat“. Je vhodné nechat je chvíli přemýšlet, žáci na problém postupně začnou přicházet. Eventuelně je možnost rady ve formě dotazu typu „Co bylo potřeba dřív, aby se scénář opakoval neustále?“

Možné odpovědi na otázky:

- Jak myslíš, že tento blok bude fungovat?

Bude možné nastavit, kolikrát se určitá část scénáře zopakuje.

- Kdy si myslíš, že se spustí blok připojený na zobáček s nápisem „Konec“?

Až potom, co se dokončí opakování přes tento blok

- Co by udělal scénář, který vidíš na obrázku níže? (Napřed přemýšlej, poté ověř)

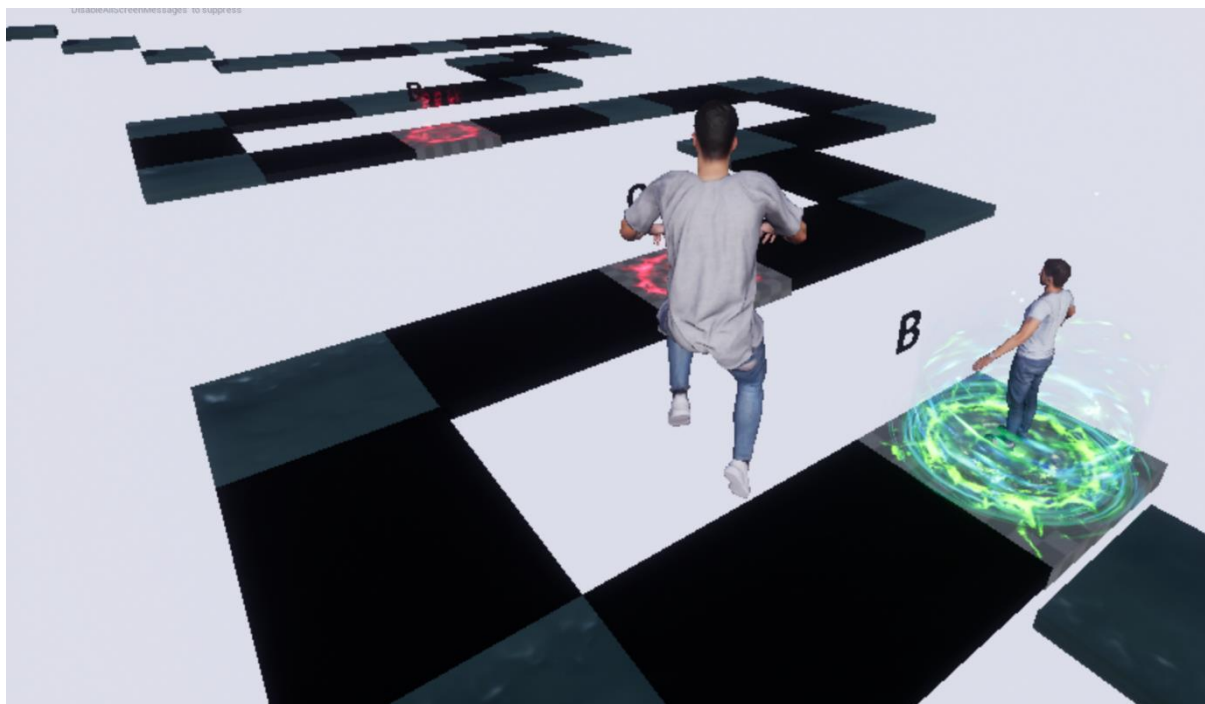
Postaví dům, nechá Bludičku popojít dopředu, následně postaví tři květiny a nakonec vozík

## 5.3.2. Lekce 2. Labyrint

Orientační časová dotace: 35 minut

Využitá úroveň: Labyrint

V lekci žáci vedou Honzu po cestě. Přecházejí po tom po jednotlivých plošinách.



OBRÁZEK 45 UKÁZKA ÚROVNĚ LABYRINT

### Jednotlivé aktivity

#### PROHLÍDKA MAPY

- Nyní otevři složku „Labyrint“, a v ní spusť soubor „Labyrint“
- Stiskni tlačítko „Play“ a prohlídni si mapu
  - Jak můžeš vidět, nachází se zde již postavená dráha, jež je tvořena čtverci (délka čtverce 200 jednotek) a obdélníky (délka obdélníku 300 jednotek)
  - Vedle tebe stojí Honza, který by se potřeboval dostat na poslední plošinu. Pomoz mu
  - Oproti Klaudii je Honza dobrým atletem, a proto je vybaven schopností skákat

#### TRÉNING OPAKOVÁNÍ

- Spusť soubor „Honza\_Cesta“
- Dostaň se s Honzou na plošinu B. Vhodně k tomu využij blok „Opakovat“
- Stiskni tlačítko „Play“

- Jak nyní můžeš vidět, Honza i Emanuel nyní začínají z druhé plošiny. Ber to v potaz
- Odpoj svůj scénář, ale nenič ho, možná se ti bude ještě hodit
- Nyní se dostaň s Honzou na plošinu C. Opět k tomu vhodně využij blok „Opakovat“ vezmi zde v potaz, že není vždy nutné opakovat celý tvůj scénář
- Opět svůj scénář odpoj, ale nenič
- Nyní před sebou máš plošinu D. Dostaň se na ni s použitím bloku „Opakovat“
- Uprav tento scénář tak, aby nikde za sebou nebylo více bloků „Otočit Se Doleva“
- Jak tento nový scénář funguje? Pokus se ho popsat učiteli
- Nyní možná nejtěžší úsek. Dostaň se na poslední plošinu E. Tip: Pořádně si před začátkem stavby dráhu prohlédni. Nejsou ti tam ty části povědomé?

#### POZNÁMKY PRO VYUČJÍCÍHO

*Lekce jako celek má za cíl procvičit používání bloku „Opakovat“.*

*Žáci mohou mít problémy s určováním vzdálenosti, kterou má Honza urazit. Je proto dobré žákům připomenout, že čtverce mají 200 a obdélníky 300 jednotek. Po překonání prvního úseku již žáci tento problém nemívají.*

*Žáci taky často nepřemýšlejí nad tím, jak by aplikovali opakování, a scénář vytváří bez něj. Tato situace se může vyřešit způsobem, že vyučující vyhlásí jakousi formu soutěže a jednotlivé dvojice mezi sebou soutěží o to, kdo bude mít co možná nejkratší scénáře.*

*U posledního úkolu žáci mohou použít všech předchozích scénářů, které mírně upraví. Pokud žáci začnou tvořit scénář od začátku, není to chyba. Vyučující však může žáky navnadit k prozkoumání předchozích scénářů.*

*Pro zpětné testování scénářů lze využít blok „Smazat pokrok“, který vrátí Honzu zpět na plošinu A*

### 5.3.3. Lekce 3. Zpět ke Klaudii

Orientační časová dotace: 20 minut

Využitá úroveň: Cesta\_S\_Klaudii

V lekci žáci upravují scénář pro Klaudii z lekce Chut' na jablka. Aplikují na něj opakování.

#### Jednotlivé aktivity

- Nyní otevři složku „Cesta“ a v ní spust' soubor „Cesta\_S\_Klaudii“
- Dále otevři soubor „Klaudie\_cesta“
  - Klaudie se doslechla, že Honza používá blok „Opakovat“, díky čemuž nemusí ve své paměti uchovávat tolik bloků
  - Jelikož stále chodí Emanuelovi krást jablka, tak tě poprosila, zda bys jí mohl/a předělat scénář tak, aby se v něm vyskytovalo co možná nejméně bloků
- S použitím bloku „Opakovat“ zkrat' scénář tak, aby v něm bylo nanejvýš 15 bloků (včetně bloku Event Start). Odlož si stranou kopii tvého dosavadního scénáře pro případ, že se ti jej nepovede napoprvé přepracovat správně

#### POZNÁMKY PRO VYUČUJÍCÍHO

*Tuto lekci je opět vhodné vyhlásit jakožto formu soutěže, jejímž cílem by tentokrát bylo splnění úkolu v co možná nejkratší době. Je vhodné ocenit všechny, kteří lekci dokončí, ne pouze první místa. Neocenění pomalejších skupin by mohlo vést k jejich demotivaci k plnění následujících lekcí.*

*Předtím než žáci začnou přetvářet svůj scénář, je podstatné jim připomenout, že svůj původní scénář mají nechat nedotčený. Předejde se tím nutnosti začínat od znova v případě výrazných chyb, vedoucích k deformaci scénáře.*

*Ke konci, když již většina skupin bude mít své scénáře upravené, je možné zbývajícím skupinám pomoci odhalit, kde by se dalo opakování použít. Samotnou konstrukci scénáře by však stále měli vytvořit sami žáci.*

## 5.4. Část 3. Vlastní bloky

### Cíl výuky

- Žák dokáže vytvořit a pojmenovat vlastní blok
- Žák dokáže se svými bloky pracovat ve scénáři

### Témata lekcí

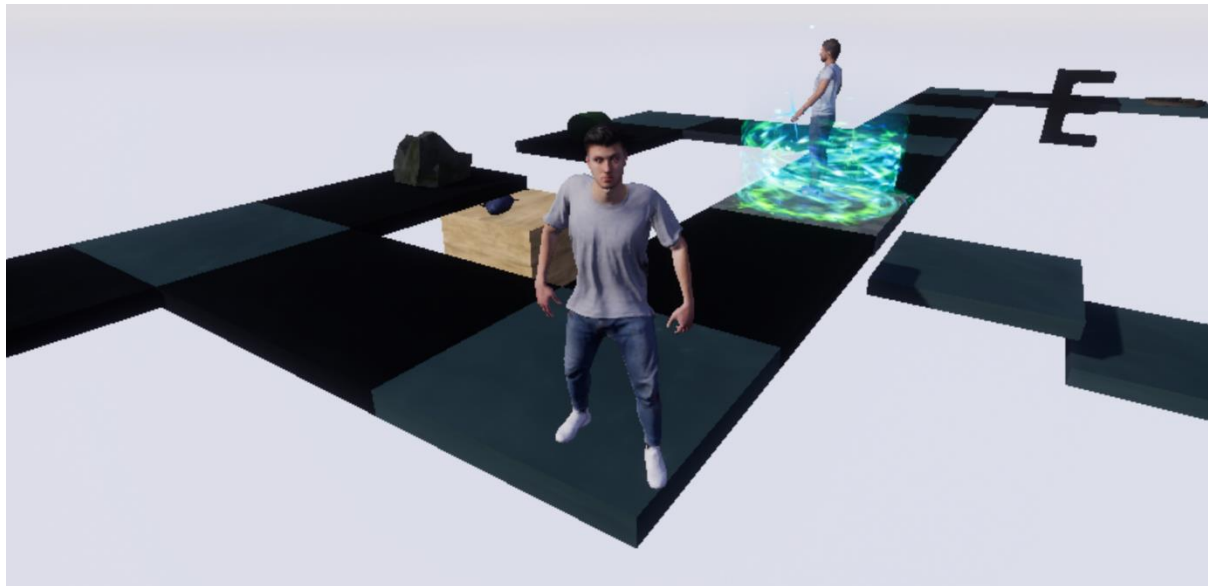
- Vedení Honzy k předmětům
- Stavba osady

### 5.4.1. Lekce 1. Honzovy cesty

Orientační časová dotace: 45 minut

Využitá úroveň: Labyrint

V lekci žáci upravují vedou Honzu k jednotlivým předmětům v úrovni. Učí se u toho vytvářet a využívat vlastní bloky.



OBRÁZEK 46 UKÁZKA ÚROVNĚ LABIRINT

### Jednotlivé aktivity

#### PROHLÍDKA MAPY

- Otevři složku „Labyrint“, a v ní spust’ soubor „Labyrint“
- Stiskni tlačítko „Play“ a prohlídni si zbytek mapy

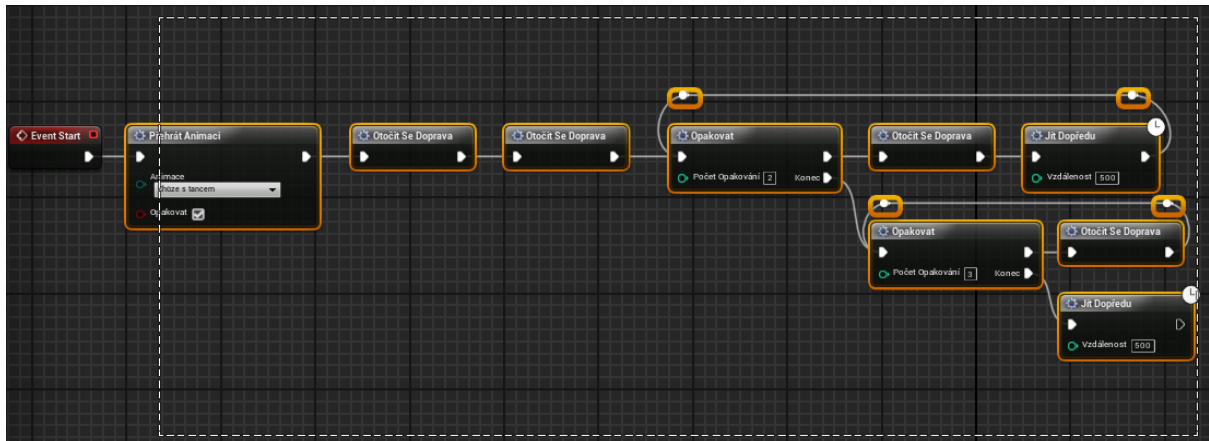
- Jak můžeš vidět, nachází se zde 4 předměty, jež jsou umístěny na dráze
- Honza by tyto předměty potřeboval posbírat

### OPAKOVÁNÍ

- Spust' soubor „Honza\_Cesta“
- Dostaň Honzu k dýni. Nezapomínej na blok „Opakovat“

### VYTVOŘENÍ VLASTNÍHO BLOKU

- I s použitím bloku „Opakovat“ se stále jedná o poměrně dlouhý scénář, obzvlášť vezmeme-li v potaz, že se s Honzou budeme potřebovat dostat i k ostatním předmětům
- Naštěstí existuje způsob, jak tento scénář zkrátit, respektive „Skrýt do bloku“
- Označ si svůj scénář podobně jako na obrázku níže



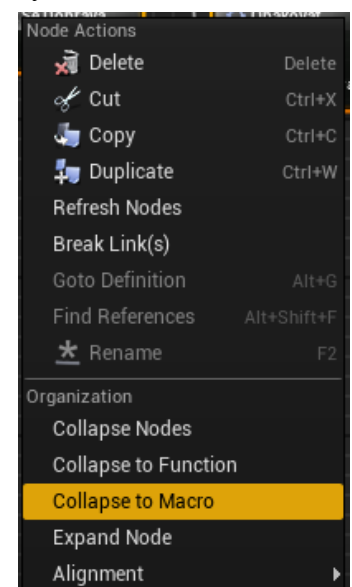
OBRÁZEK 47 OZNAČENÍ SCÉNÁŘE

- Nyní klikni na jakýkoliv ze zvolených bloků pravým tlačítkem myši
- Z uvedené nabídky vyber možnost „Collapse to Macro“ viz obrázek vpravo
  - Nyní vidíš, že místo všech bloků, které byly označeny, se objevil pouze jediný, jenž nese jméno „New Macro 0“



OBRÁZEK 48 NOVÝ BLOK

- Stiskni tlačítko „Play“



OBRÁZEK 49 NABÍDKA MOŽNOSTÍ

- Jaká změna nastala v Honzově chování?

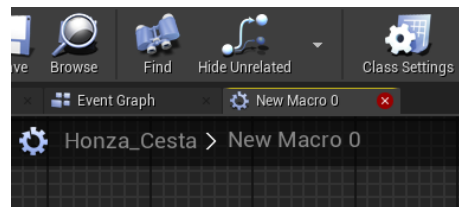
- Co myslíš, že jsi touto změnou přesně učinil/a?

- Učiň dvojklik na tento nový blok

- Co vidíš?

- Pobav se se sousedem o tom, jak tvůj scénář nyní nejspíš funguje. Následně svou domněnku předneš vyučujícímu

- Zpět na svůj scénář se dostaneš stisknutím křížku záložky nesoucí název tohoto bloku, která je umístěna uprostřed nahoře



OBRÁZEK 50 LIŠTA ČÁSTÍ KÓDU

- Zkontroluj si seznam bloků

- Jak můžeš vidět, přibyl zde blok, který si stvořil/a

- Trojklikem na něj v seznamu „Macros“ jej přejmenuj vhodnějším názvem

#### POZNÁMKY PRO VYUČJÍCÍHO

Podkapitola „Vytvoření vlastního bloku“ je stvořena jakožto návod. Cílem je, aby žák byl schopen, prozatím podle návodu, převádět části scénáře do bloků. Otázky jsou zde z toho důvodu, aby nutily žáky zamyslet se nad provedenými změnami.

Vzhledem k charakteru podkapitoly vyučující může žákům radit, na co a v jakém pořadí mají kliknout, aby vytvořili vlastní blok. Odpovědi na otázky by však měly zůstat na žácích, vyučující může s žáky nad těmito otázkami debatovat.

Možné odpovědi na otázky:

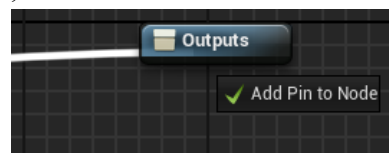
- Jaká změna nastala v Honzově chování?

- Co myslíš, že jsi touto změnou přesně učinil/a?

- Co vidíš?

## TRÉNINK TVORBY VLASTNÍCH BLOKŮ

- Stejným způsobem stvoř bloky, jež dostanou Honzu ke kolu, ke kameni a k melounu (Vždy z plošiny)
  - Podívej se na tvé bloky. Můžeš si všimnout, že mají pouze vstupní zobáček, ale chybí jim výstupní
- Otevři si jeden z těchto bloků
- Přidej výstupní zobáček prostým přetažením zobáčku, z konce scénáře tvého bloku do prázdné části bloku „Outputs“
- Nyní použij své bloky k tomu, aby Honza posbíral všechny předměty na dráze. Dopomoz si blokem „Vrátit postavy na plošinu“



OBRÁZEK 51 PŘIPOJOVÁNÍ VÝSTUPU

- Kolik bloků má tvůj nynější scénář?

- Kolik přibližně by jich mohl mít bez použití tvých bloků?

- Pobavte se ve skupině o tom, jaké všemožné výhody a nevýhody může mít používání vlastních bloků. Následně své domněnky zapište a prodiskutujte s vyučujícím


### POZNÁMKY PRO VYUČJÍCÍHO

*Tato podkapitola „Trénink tvorby vlastních bloků“ má za cíl procvičovat tvorbu nových bloků, ukázat úpravu nutnou k použití v sérii s dalšími bloky a ukázat možné použití vlastních bloků.*

*Žáci opakují postup tvorby nového bloku k tomu, aby si jej lépe zapamatovali. Vyučující může při vytváření prvních bloků pomáhat, u posledních by však již žáci měli postupovat sami.*

*Pokud tak žáci sami neučiní, měl by je učitel vyzvat k využívání opakování v blocích a pojmenovávání bloků. Většina žaku se podle těchto pravidel řídí bez vyzvání.*

*Při připojování bloku „Outputs“ žáci mohou mít problémy. Důvodem je, že je nutné bílou čáru napojit do prázdného prostoru bloku, nikoliv na název. V tomto případě vyučující může žákům pomoci.*



Na konci lekce je velice vhodné, aby vyučující diskutoval nad jednotlivými otázkami, případně třídě pokládal další otázky.

Možné odpovědi na otázky:

- *Kolik bloků má tvůj nynější scénář?*

8

- *Kolik přibližně by jich mohl mít bez použití tvých bloků?*

20-50

- *Pobavte se ve skupině o tom, jaké všemožné výhody a nevýhody může mít používání vlastních bloků. Následně své domněnky запиšte a prodiskutujte s vyučujícím*

*Nevidíme, jak blok funguje; je pracné jej vytvořit, ...*

*Scénář je kratší; Podle názvu bloku zjistím, co by mohl vykonávat, ...*

## 5.4.2. Lekce 2. Osidlujeme svět

Orientační časová dotace: 60 minut

Využitá úroveň: Svět

V lekci žáci vytvářejí vysoké množství objektů za použití svého bloku, který si zde též vytváří, s cílem vytvořit osadu.

### Jednotlivé aktivity

#### STAVBA POZEMKU

- Otevři složku „StavěníSvěta“, a v ní spusť soubor „Svět“
- Nyní spusť soubor „Bludička“
- Postav libovolný smysluplný pozemek . Vyučující ti zadá, kolik na to máš času. Příklad takového pozemku je na obrázku níže
- Zajisti, aby směr bludičky byl na konci stejný jako na začátku

Př:



OBRÁZEK 52 PŘÍKLAD LIDSKÉHO STAVENÍ

#### POZNÁMKY PRO VYUČUJÍCÍHO

*Podkapitola „Stavba pozemku“ je určena k vytvoření podkladů pro následující podkapitolu. Je vytvořena též z důvodu, aby žáci mohli použít svou fantazii a převést ji do scénáře.*

*Tuto podkapitolu je opět vhodné vyhlásit jakožto formu soutěže. V tomto případě mohou být kritéria vyhodnocování různá, od nejkrásnějšího stavení přes nejlepší scénář.*

*Je nutné žákům zadat, kolik mají na stavbu času. Dále je možné zadat upřesňující kritéria pro stavbu, například, že dům musí být obehnan plotem, či že ve scénáři musí být použito opakování. Pokud se vyučující rozhodne nezadat žádná další kritéria, vhodná délka času stavby je 15 minut.*

#### STAVBA OSADY

- Vytvoř z tohoto scénáře svůj blok a vhodně jej pojmenuj
- Postav čtyři tato stavení tak, aby byly umístěny do čtverce
  - Jak vidíš, na mapě vznikla vesnička s použitím jen několika bloků
- Proč bylo nastavit tvůj blok tak, aby byl směr bludičky stejný jako na počátku?

- Pohraj si s počtem opakování, úhlem otočení a vzdáleností. Pokus se zajistit, aby se vesnička vytvořila kolem dokola. Měň úhel otočení bludičky v násobcích či dělitelích 90. Zkus použít blok Nastavit rychlost bludičky
- Připoj na konec scénáře blok „Uložit mapu“ pro uložení tvé osady

- Pokud jsi se svou vesničkou nespokojen/a, obdobně můžeš použít blok „Zbourat stavby“
- Svou osadu libovolně upravuj podle své fantazie

### POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Stavba osady“ ukazuje možnost použití vlastních bloků vícekrát ve scénáři s využitím opakování. Dále ukazuje žákům ucelenost scénáře vlastního bloku.

Podkapitola žáky převážně velice baví, proto zkoušejí různé varianty stavby osady. Tuto iniciativu žáků je vhodné podporovat a případně přidávat další prvky do osady, které žáci mohou stavět.

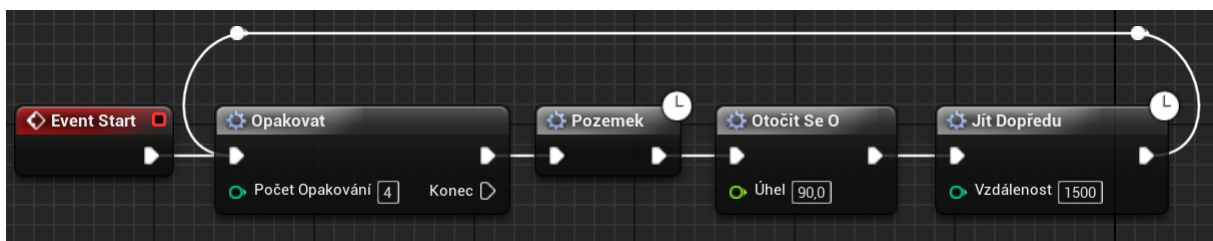
Žáci mohou mít problémy se stavbou osady do kruhu. V takové chvíli může vyučující dávat otázky, vedoucí k souvislosti úhlu otočení, počtem opakování a vzdáleností.

Žáci mohou vyučujícího požádat, zda by nemohli pokračovat ve stavbě své osady i v následujících hodinách. Pokud má vyučující dostatek času, tak tato podkapitola je vhodná k prodloužení, neboť poskytuje vysokou úroveň svobody. Lze se k ní také zpětně vrátit z následujících částí.

Možné odpovědi na otázky:

- Proč bylo nastavit tvůj blok tak, aby byl směr bludičky stejný jako na počátku?

Aby bylo jednodušší ve scénáři stavět více domů.



OBRÁZEK 53 SCÉNÁŘ PRO STAVBU OSADY DO ČTVERCE

## 5.5. Část 4. Vstupy a výstupy

### Cíl výuky

- Žák dokáže vytvářet události závislé na interakci s uživatelem
- Žák dokáže spouštět scénáře z událostí, volaných po stisknutí tlačítek na obrazovce či stisknutí tlačítka na klávesnici
- Žák dokáže upravit popis vstupů podle činnosti, jež vykoná scénář k nim napojený

### Témata lekcí

- Vytváření scénářů po stisknutí tlačítek na obrazovce
- Vytváření scénářů po stisknutí klávesnice

### 5.5.1. Lekce 1. Tlačítka

Orientační časová dotace: 40 minut

Využitá úroveň: Prázdnota

V lekci žáci propojují tlačítka na obrazovce s jednotlivými scénáři. Dále jsou zde zmíněny i některé možnosti výstupu, jako je odeslání textu do chatu.



OBRÁZEK 54 UKÁZKA ÚROVNĚ PRÁZDNOTA

## Jednotlivé aktivity

### PROHLÍDKA MAPY

- Otevři složku „Chat“, a v ní spust' soubor „Prázdnota“
  - Jak je nyní vidět, opravdu v tomto světě není nic moc k vidění. Nebo snad ano?
- Stiskni tlačítko „Play“
- Co vidíš?

- Co se stane po kliknutí na něco?

### POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Prohlídka mapy“ seznamuje žáky s novým konceptem prostředí, ve kterém se prostředí mění až po zahájení simulace. Dále žákovi představí možné vstupy a výstupy v podobě tlačítek a chatu.

Žáci se mohou dotazovat, co se má stát po stisknutí tlačítek. Vyučující by měl šikovně žáky navést k tomu, že „nic“ je správná odpověď.

Možné odpovědi na otázky:

- Co vidíš?

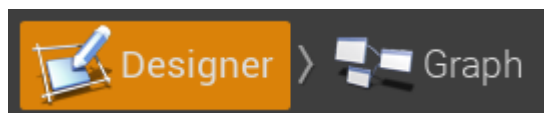
V levém spodním rohu se objevilo několik okének

- Co se stane po kliknutí na něco?

Nic

### VLOŽENÍ TEXTU

- Spust' soubor „Chat“
  - Zde vidíš podobnou scénérii, jakou jsi viděl/a po stisknutí tlačítka „Play“ Tato část se nazývá „Designer“ neboli grafická část. Nám jde však o to, abychom věci, jež vidíš před sebou, přiměli k práci
- Překlikni ze stránky „Designer“ na stránku „Graph“ vpravo nahoře
- Nyní se zaměř na blok „Vložit text“.  
Prohlédni si jej.



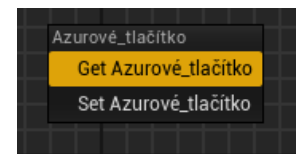
OBRÁZEK 55 PŘEPÍNAČ ZOBRAZENÍ VZHLEDU A BLUEPRINT

- Všimni si, že tento blok obsahuje vstupy „Text“, k němuž jde zapsat libovolný text. Mimo to obsahuje i vstup „Kam“, jenž obsahuje seznam všelijakých věcí. Tento seznam ignoruj

- Připoj jej k „Event Start“ a otestuj
- Co se nyní děje?

- Nyní se zaměř na levou část obrazovky. Všimni si, že na bloky „Macros“ je další seznam bloků „Variables“
- Přečti si jednotlivé položky v seznamu. Zamysli se, z čemu by tyto bloky mohly být dobré

- Jeden z nich přetáhni do blízkosti bloku „Vložit text“
  - Blok se nepřenesे okamžitě, místo toho se objeví tabulka, která obsahuje dvě možnosti



OBRÁZEK 56 MOŽNOSTI VLOŽENÍ PROMĚNNÝCH

- Klikni na první možnost viz. obrázek vpravo
- Tento blok se od všech ostatních bloků, které jsme doposud použili, na první pohled liší. V čem?

- Nyní propoj tento blok s blokem „Vložit text“ obdobně, jako když jsi propojoval/a bloky vždy předtím. Jediný rozdíl je ten, že nyní propojíš modrý zobáček s modrým podobně, jako na obrázku vpravo



OBRÁZEK 57 VLOŽENÍ TEXTU DO TLAČÍTKA

- Otestuj
- Co se nyní stalo?

- Zkus připojit různé bloky ze seznamu „Variables“ k bloku „Vložit text“, který již máš ve scénáři
- Co se mění?

- Přidej další bloky „Vložit text“ a zajisti, aby všechna barevná tlačítka nesla název své barvy

- Zajisti, aby se názvy tlačítek objevovaly každé následující po 0,5 sekundách
- Zajisti, aby se tvůj scénář zkrátil a zároveň dělat totéž. Jak to zařídiš?

### POZNÁMKY PRO VYUČUJÍCÍHO

*V podkapitole „Vložení textu“ je žákovi ukázáno, jak vložit text do určité komponenty, v tomto případě tlačítka. Následně je tato dovednost procvičena.*

*Žáci mohou být zmateni z nabídky objektů v bloku „Vložit text“. V UE4 bohužel není přímá možnost odstranění tohoto seznamu. Je proto žákům třeba vysvětlit, že tento seznam musí ignorovat. Ani v běžném programování v UE4 se tento seznam většinou nevyužívá.*

*Při dotazu „Co se děje po připojení bloku „Vložit text“ za blok „Event Start““ má být správná odpověď opět „nic“. Žáci z toho mohou být zmateni. Toho je možné této situace využít k tomu, aby se zamysleli nad tím, proč se nic nestalo. Toho je možné docílit pokládáním otázek typu: „Kam by se vlastně měl text napsat?“, eventuálně: „Proč zrovna do chatu?“.*

*Při přemýšlení žáků nad významem bloků v seznamu „Variables“ žáci nemusí dojít ke správnému závěru. To v danou chvíli nevadí. Žáci povětšinou přijdou na význam těchto bloků ve chvíli, kdy je začnou propojovat s blokem „Vložit text“.*

*Při úkolu vytvoření vlastního scénáře žákům může chvíli trvat, než jim dojde, jak dané zadání vyřešit. To nemusí být na škodu, žáci si při tomto přemýšlení urovnávají, co se již naučili.*

*Možné odpovědi na otázky:*

- *Co se nyní děje?*

Opět nic.

- *Tento blok se od všech ostatních bloků, které jsme doposud použili, na první pohled liší. V čem?*

Má jinou barvu, Nemá bílé zobáčky.

- *Co se nyní stalo?*

V tlačítku je text „Ahoj“.

- *Co se mění?*

Mění se tlačítko, ve kterém je napsáno „Ahoj“.

- *Zajisti, aby se tvůj scénář zkrátil a zároveň dělat totéž. Jak to zařídiš?*

Vytvoříme vlastní blok.

## STISKnutí TLAČÍTEK

- Nyní klikni na jakékoliv barevné tlačítko ze seznamu „Variables“ (pouze klikni, nepřidávej do scénáře)
- Vlevo dole vidíš seznam pojmenovaný „Events“  
Klikni na první možnost (u této možnosti na +, které má zelené pozadí)



OBRÁZEK 58 VYTVOŘENÍ UDÁLOSTI PO

- Objevil se nový blok. Dokážeš jej přeložit do češtiny?

- Zamysli se, k čemu by tento blok mohl být dobrý
- Zapoj za tento blok scénář, který napíše do chatu text dle tvé fantazie. Blok chatu je umístěn v podseznamu „Chat“. Tento podseznam otevřeš kliknutím na znak zobáčku vedle něj. Pokud ti to nepůjde, požádej o radu napřed jiné skupiny a pokud ani ony nebudou vědět, tak vyučujícího
- Obdobně vytvoř takovéto scénáře pro všechny ostatní barevná tlačítka s tím, že se do chatu napíše vždy jiný text.

## POZNÁMKY PRO VYUČUJÍCÍHO

*V podkapitole „Stisknutí tlačítek“ je žákovi ukázáno, jak vytvořit událost, která je volána po stisknutí určitého tlačítka. Tato dovednost je následně procvičena.*

*V této podkapitole se neobjevovaly žádné větší specifické problémy. Žáci povětšinou chvíli přemýšlejí nad tím, která událost je spojena, s jakým tlačítkem, což není na škodu.*

*Možné odpovědi na otázky:*

- *Objevil se nový blok. Dokážeš jej přeložit do češtiny?*

*Na kliknutí/ Při kliknutí „Název tlačítka“.*

## ČTENÍ TEXTU

- Nyní se zaměř na blok „Vyčist text“. Přidej jej do blízkosti scénáře libovolného tlačítka
- Pozorně si jej prohlédni. Vidíš nějakou podobnost s blokem „Vložit text“?



- K čemu předpokládáš, že by se tento blok dal využít?

- Ověř svou domněnku tak, že po stisknutí tlačítka se do chatu napíše jméno tlačítka, jež bylo stisknuto. Pokud ti to nepůjde, požádej o radu napřed jiné skupiny a pokud ani ony nebudou vědět, tak vyučujícího

### POZNÁMKY PRO VYUČUJÍCÍHO

*Podkapitola „Čtení textu“ je zaměřena na představení bloku „Vyčíst text“. Žáci tento blok dále testují.*

*Některé skupiny mohou mít problém pochopit význam bloku „Vyčíst text“. V diskuzi s ostatními skupinami přicházejí na jeho význam bez zásahu vyučujícího.*

*U druhého dotazu této podkapitoly není vyžadována správná odpověď. Otázka je zde položena pro rozpoutání debaty ve skupině.*

*Možné odpovědi na otázky:*

- *Pozorně si jej prohlédni. Vidiš nějakou podobnost s blokem „Vložit text“?*

*Obsahuje stejné zobáčky, akorát jsou umístěny jinde.*

- *K čemu předpokládáš, že by se tento blok dal využít?*

*Ke zjištění textu, který je umístěn v tlačítku.*

### TEXTOVÉ FUNKCE

- Nyní se opět zaměř na levou stranu obrazovky. Nad seznamem bloků „Macros“ je další seznam „Functions“. Z něj přetáhni do blízkosti scénáře jiného barevného tlačítka blok „Spojit texty“
- Popřemýšlej, k čemu by tento blok mohl sloužit
- Následně zajisti, aby se po stisknutí tebou zvoleného barevného tlačítka napsal do chatu obsah bloku „Spojit texty“
- Do tohoto bloku zapiš do Text 1 „Zdra“ a do Text 2 „vím“. Co se napíše po stisknutí tlačítka? Napřed odpověz, poté otestuj

- Nech tímto způsobem vypsát jiné slovo dle své fantazie
- Vypiš tímto způsobem 2 slova (1. slovo z „Text 1“, 2. slovo z „Text 2“)

- Na co je nutné si dávat pozor, aby se napsala dvě slova?

### POZNÁMKY PRO VYUČUJÍCÍHO

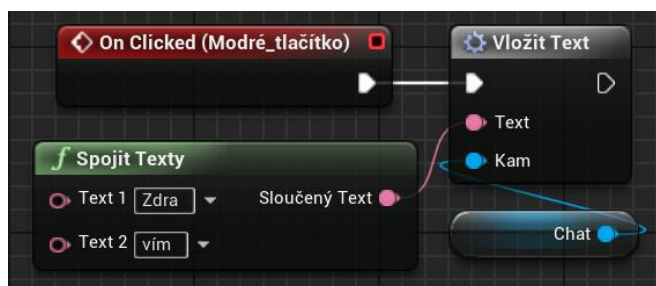
Podkapitola „Textové funkce“ se zaměřuje na používání bloku „Spojit texty“, který dokáže spojit dva řetězce.

Žákům může chvíli trvat, než odhalí způsob, jakým tento blok funguje, avšak nakonec všichni s největší pravděpodobností přijdou na jeho funkci bez zásahu učitele.

Možné odpovědi na otázky:

- Do tohoto bloku zapiš do Text 1 „Zdra“ a do Text 2 „vím“ Co se napíše po stisknutí tlačítka? Napřed odpověz, poté otestuj

- Na co je nutné si dávat pozor, aby se napsala dvě slova?



OBRÁZEK 59 VYPSÁNÍ TEXTU ZDRAVÍM

### POKROČILEJŠÍ ÚLOHY

- Zajisti, aby se po stisknutí každého z barevných tlačítek napsal do chatu text tohoto tlačítka a za něj nějaký text podle tvé fantazie
- Změň text všech barevných tlačítek, který se nastaví po stisknutí „Play“, aby sis ověřil/a, zda vše funguje, jak má
- Co se stane, když stiskneš tlačítko, jemuž se ještě nepřihodil text?

- Proč se tak stane?

- Sestav scénář, který po stisknutí tlačítka „Tlačítko\_odeslat“ napíše do chatu text, jenž během testování napíšeš těsně pod chat

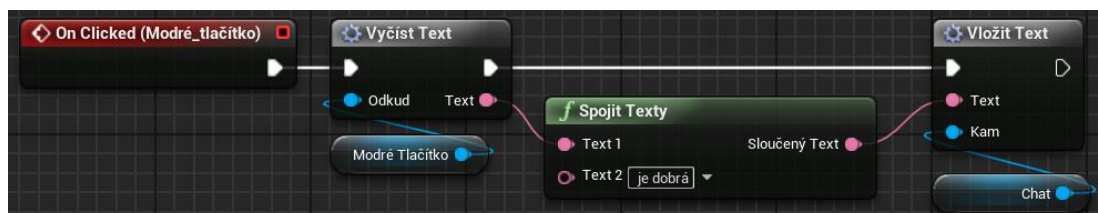
## POZNÁMKY PRO VYUČJÍCÍHO

- *Co se stane, když stiskneš tlačítko, jemuž se ještě nepřiradil text?*

*Napiše se jen druhé slovo*

- *Proč se tak stane?*

*Protože tlačítko žádný text neobsahuje*



OBRÁZEK 60 SCÉNÁŘ, VYPISUJÍCÍ TEXT TLAČÍTKA S DALŠÍMI SLOVY PO STISKU DANÉHO TLAČÍTKA

## 5.5.2. Lekce 2. Kosti na hřbitově

Orientační časová dotace: 40 minut

Využitá úroveň: Hřbitov

V lekci žáci zajišťují ovládání postavy skrze klávesnici.



OBRÁZEK 61 UKÁZKA ÚROVNĚ HŘBITOV

## Jednotlivé aktivity

### PROHLÍDKA MAPY

- Otevři složku „Kostlivec“, a v ní spust' soubor „Hřbitov“
- Stiskni „Play“ a prozkoumej hřbitov
  - Jak vidíš, místo našeho starého známého Emanuela máš před sebou starého kostlivece s mečem. Aby se s ním dalo pohybovat, je potřeba jej naprogramovat

### NOVÉ BLOKY KOSTLIVCE

- Spust' soubor „Kostlivec“
- Pro začátek si prohlédni animace, jež má kostlivec k dispozici
- Věnuj pozornost blokům „Spustit chůzi“ a „Zastavit chůzi“. Jak je rozdílný oproti bloku „Jít dopředu“, který jsme používali dříve? Při testování si dopomoz blokem „Čekej“

- Zajisti, aby kostlivec chodil do čtverce

### POZNÁMKY PRO VYUČUJÍCÍHO

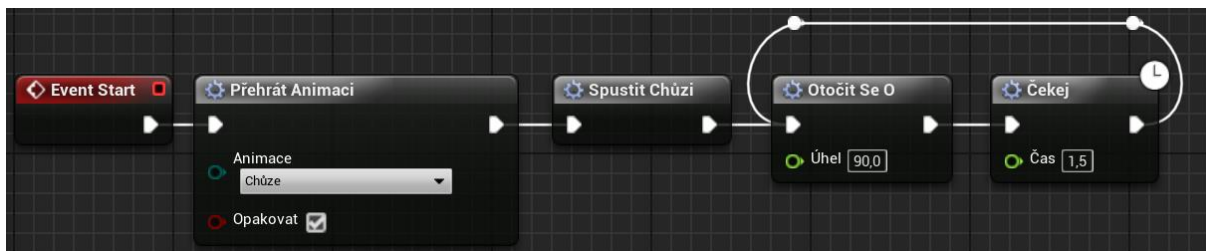
*Podkapitola „Nové bloky kostlivece“ představují animace, jimiž kostlivec disponuje a především bloky „Spustit chůzi“ a „Zastavit chůzi“.*

*Žáci si mohou nějakou dobu hrát s jednotlivými animacemi postavy, díky čemuž nepostupují v lekci. Z počátku je vhodné toto přehlédnout, podpoří se tím motivace práce s kostlivcem. Pokud by to trvalo příliš dlouho, vyučující by měl žáky vyzvat k pokračování v lekci. Je možné je motivovat tím, že v následujících částech budou moci kostlivece ovládat právě oni.*

*Možné odpovědi na otázky:*

- *Věnuj pozornost blokům „Spustit chůzi“ a „Zastavit chůzi“. Jak je rozdílný oproti bloku „Jít dopředu“, který jsme používali dříve? Při testování si dopomoz blokem „Čekej“*

*To, jak daleko postava ujde, není určeno vzdáleností, ale časem.; Jedná se o dva bloky. Jeden chůzi spustí a druhý zastaví.*

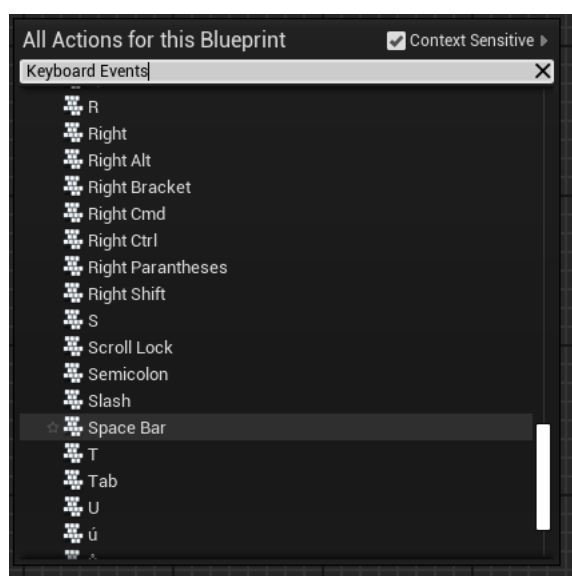


OBRÁZEK 62 SCÉNÁŘ PRO CHŮZI DO ČTVERCE

## STISKNUTÍ KLÁVESY

- Nyní začneme kostlivce ovládat. Napřed přelož slovní spojení „Keyboard events“

- Nyní klikni pravým tlačítkem myši do prázdna okolo tvého scénáře
- Do otevřené tabulky zadej „Keyboard events“
- Z nabídky vyber možnost „Space Bar“ (Mezerník)
  - Na místě, kde jsi prve klikl/a, se nyní objevil nový blok „Space Bar“
- Otestuj, jak fungují bílé zobáčky tohoto bloku. Dopomoz si blokem „Skok“



OBRÁZEK 63 NABÍDKA VÝBĚRU BLOKŮ PO ZADÁNÍ KEYBOARD EVENTS

- Jak se tyto zobáčky chovají?

- Porad' se s ostatními skupinami o tom, jak by se tato vlastnost dala využít

## POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Stisknutí klávesy“ je vytvořena jakožto návod pro přidání události, volané po stisknutí či puštění určité klávesy.

Žáci mohou mít problém samotnou událost do scénáře přidat. UE4 naneštěstí nepodporuje jednodušší způsob vkládání událostí závislých na klávesnici. Je proto možné žákům při vkládání události pomoci.

*Samotné testování události by mělo probíhat bez větších problémů.*

*Možné odpovědi na otázky:*

- *Nyní začneme kostlivce ovládat. Napřed přelož slovní spojení „Keyboard events“*

<i>Události na klávesnici</i>
-------------------------------

- *Jak se tyto zobáčky chovají?*

<i>První zobáček se aktivuje po stisknutí mezerníku</i>
---

<i>Druhý zobáček se aktivuje po puštění mezerníku</i>
---

#### OVLÁDÁNÍ KOSTLIVCE

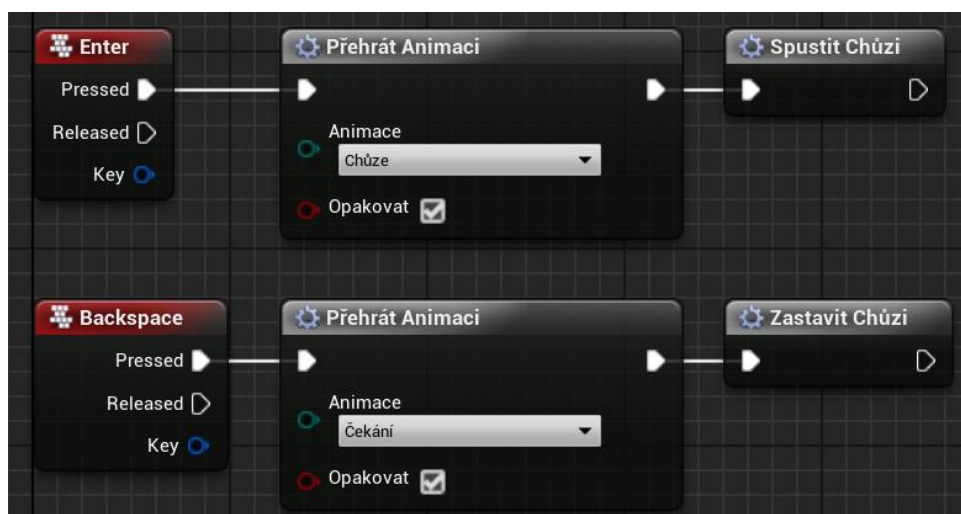
- Zajisti, aby postava po startu pouze čekala
- Nyní zajisti, aby po stisknutí klávesy „Enter“ postava šla rovně a po stisknutí klávesy „Backspace“ zastavila a opět čekala
- Nyní zařid', aby se postava po stisknutí nějaké z šipek na klávesnici otočila jejím směrem. Použij k tomu blok „Nastavit úhel postavy“. (Názvy šipek jsou: „Up“, „Down“, „Right“, „Left“). Pokud nemůžeš přijít na to, jak funguje blok „Nastavit úhel postavy“ porad' se s jinými skupinami
- Vyzkoušej v kombinaci s ostatními tlačítky, které již mají scénář
- Nyní zajisti, aby se po stisku čtyř tebou zvolených kláves (kromě těch, co jsou již použity) postava vydala určitým směrem a po jejich puštění se zastavila. Můžeš využít klávesy W,S,A,D
- Vytvářej další scénáře na různé klávesy dle tvé fantazie

## POZNÁMKY PRO VYUČJÍCÍHO

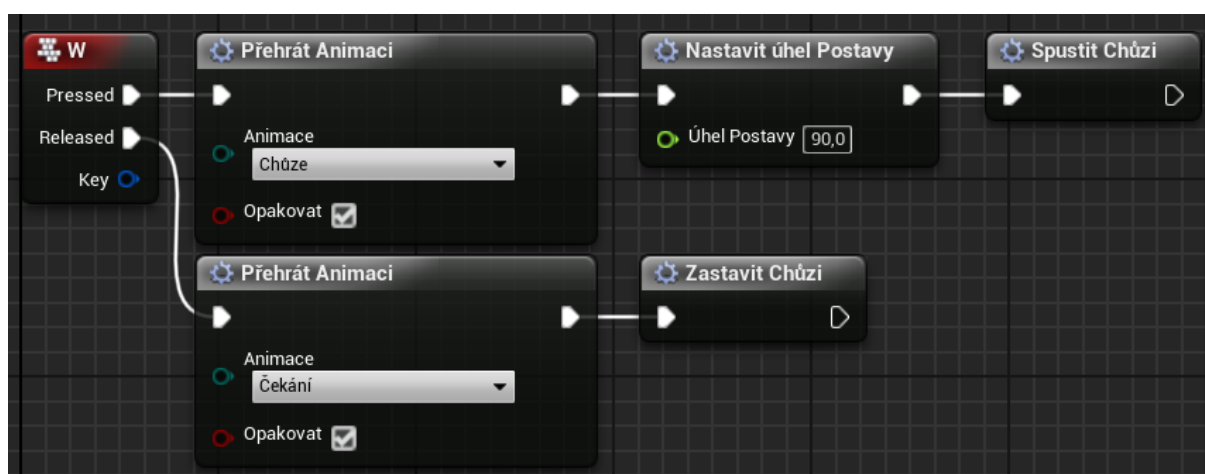
Podkapitola „Ovládání kostlivce“ trénuje vytváření událostí klávesnice. Dále učí žáka možné využití stisknutí a puštění klávesy.

Žáci opakují postup tvorby nové události, aby si jej lépe zapamatovali. Vyučující může při vytváření prvních událostí pomáhat, u posledních by však již žáci měli postupovat sami.

Při zadání nastavování úhlu postavy pomocí šipek se žáci setkají s novým blokem „Nastavit úhel postavy“. Téměř všichni žáci přijdou na funkci bloku intuitivně. Ostatní na funkci bloku přijdou po poradě s jinými skupinami.



OBRÁZEK 64 SCÉNÁŘE PRO START A KONEC CHŮZE



OBRÁZEK 65 SCÉNÁŘ PRO OVLÁDÁNÍ POSTAVY V JEDNOM SMĚRU

## 5.6. Část 5. Podmínky

### Cíl výuky

- Žák dokáže vytvořit podmínku
- Žák dokáže určit kdy je a kdy není podmínka splněna
- Žák dokáže na základě splnění či nesplnění podmínky upravovat scénář za danou podmínkou.

### Témata lekcí

- Tvoření otázek s různými odpověďmi
- Vedení Honzy k jednotlivým předmětům

#### 5.6.1. Lekce 1. Kvíz

Orientační časová dotace: 60 minut

Využitá úroveň: Prázdnota

V lekci je žákům představen blok „Když“. Žáci jej využívají k pokládání a vyhodnocování otázek.

#### Jednotlivé aktivity

##### BLOK KDYŽ

- Otevři složku „Chat“, a v ní spust' soubor „Prázdnota“
- Spust' soubor „Chat“
- Vyhledej blok „Když“ a připoj jej na konec scénáře začínajícím „Event Start“
- Pozorně si jej prohlédni. Jak předpokládáš, že by mohl fungovat?


- Otestuj jej. Dopomoz si bloky „Vložit text“ a „Chat“. Měj zaškrtnuté políčko u zobáčku „Podmínka“
- Vyhledej blok „Text=Text“ a umísti jej do blízkosti bloku „Když“
- Propoj tyto dva bloky
- Co se nyní stane po stisknutí tlačítka „Play“? (Napřed odpověz, až poté otestuj)
- Zařid', aby se změnil výsledek pouze tím, že změníš nastavení bloku „Text=Text“



## POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Blok Když“ se věnuje představení stejnojmenného bloku. Žáci si zde dělají základní obrázek o tom, jak tento blok funguje.

Žáci blok „Když“ postupně testují. Toto testování může zabrat nějaký čas.

V této podkapitole je možné, aby vyučující plnil úkoly společně s žáky a během této doby s žáky vedl debatu o fungování bloků „Když“ a „Text=Text“. Účast vyučujícího není nutná, avšak výrazně pomůže žákům pochopit fungování podmínek.

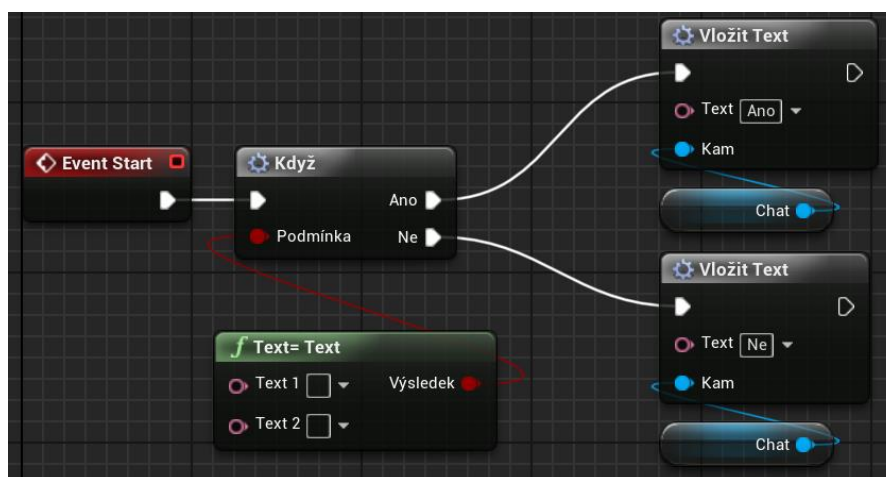
U otázky zadané v této podkapitole není cílem, aby žák odpověděl správně. Otázka je zde umístěna z důvodu, aby se žáci zamysleli nad blokem „Když“.

Možné odpovědi na otázky:

- *Pozorně si jej prohlédni. Jak předpokládáš, že by mohl fungovat?*

Pokud odpověď bude pravdivá, signál vyjde zobáčkem Ano.

Pokud ne, tak zobáčkem Ne.



OBRÁZEK 66 MOŽNÝ ZPŮSOB TESTOVÁNÍ BLOKU KDYŽ

## TVORBA OTÁZEK

- Odstraň text v tlačítkách
- Zajisti, aby po stisknutí tlačítka „Play“ byla v chatu napsána nějaká otázka, na kterou znáš odpověď
- Stiskni „Play“ a odpověz na otázku
- Přidej za scénář pro stisk „Tlačítko\_odeslat“ kontrolu, zda jsi odpověděl správně. Pokud ano, pochval se v chatu. Pokud ne, oznam si, že je odpověď špatná

- Nyní odstraň vše, co se má stát po stisku jakéhokoliv barevného tlačítka
- Co se vyčte za text, když nyní využiješ bloky „Vyčíst text“ a „Chat“ dohromady?

- Zařid', aby se po správném zodpovězení první otázky objevila otázka druhá
- Poté, co se zadá druhá otázka, nastav text několika barevných tlačítek na různé odpovědi (některé špatné, jednu správnou).
- Zařid', abys po stisknutí tlačítka se správnou odpovědí dostal pochvalu. Dej si pozor na to, že pochvalu musíš dostat pouze v případě, že odpovíš správně a že se počítač ptá na druhou otázku. Dopomoz si bloky „Vyčíst text“ v kombinaci s „Chat“
- Vytvářej další otázky, které se spustí vždy po těch předchozích. Nakonec můžeš jít po třídě a zkoušej kvízy ostatních

#### POZNÁMKY PRO VYUČUJÍCÍHO

*V podkapitole „Tvorba otázek“ postupně dostane žák za úkol vytvořit řadu otázek.*

*Podkapitola představuje sérii úloh, které jsou složitější na přemýšlení. Vyučující by měl tuto skutečnost brát v potaz a neprozrazovat žákům správné řešení. Co vyučující může dělat je to, že pokládá žákům otázky směřující ke správné odpovědi, například: „Co by mohla být podmínka pro kontrolu správné odpovědi?“ nebo: „Co by se vlastně mělo stát, pokud je podmínka pravdivá?“.*

*Ošetření uživatelských vstupů u jednotlivých otázek je vhodné řešit pouze u skupin, které jsou výrazně napřed.*

*Ke konci této lekce je možné, aby vyučující podpořil soutěživost žáků.*

*Možné odpovědi na otázky:*

- *Co se vyčte za text, když nyní využiješ bloky „Vyčíst text“ a „Chat“ dohromady?*  
*Vyčte se poslední řádek z chatu.*

*Vyčte se poslední řádek z chatu.*

## 5.6.2. Lekce 2. Nákup

Orientační časová dotace: 45 minut

Využitá úroveň: Nákup

V lekci je žákům představen blok „Když“. Žáci jej využívají k pokládání a vyhodnocování otázek.



OBRÁZEK 67 UKÁZKA ÚROVNĚ NÁKUP

### Jednotlivé aktivity

#### PROHLÍDKA MAPY

- Otevři složku „Nákup“, a v ní spust’ soubor „Nákup“
- Prohlédni si úroveň pomocí simulace
- Spust’ soubor „Honza\_Nákup“
  - Klaudie dostala hlad, a tak poslala Honzu na nákup. K dispozici je pět druhů potravin (dýně, rajče, meloun, houba, ryba)

#### POZNÁMKY PRO VYUČUJÍCÍHO

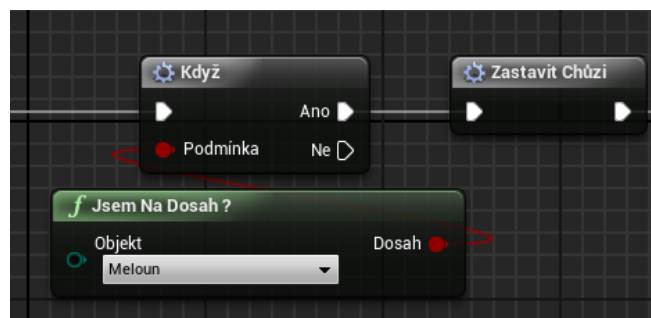
*V podkapitole „Prohlídka mapy“ mohou být jednotlivé předměty na první pohled přehlédnutelné. Vyučující může žákům konkretizovat, kde jsou přesně umístěny.*

## DETEKCE MELOUNU

- Vytvoř scénář, který zajistí, aby Honza došel k melounu. Dej si pozor, Honza nyní umí chodit pouze za použití bloků „Spustit chůzi“ a „Zastavit chůzi“ podobně, jako kostlivec z dřívějšíka. Dopomoz si blokem „Čekej“
- Jak můžeš vidět, není jednoduché určit, jak dlouho má Honza chodit, aby došel k melounu. Zkus to tedy s použitím bloku „Když“. Nahraď blok „Čekej“ blokem „Když“
- V seznamu „Functions“ vyhledej blok „Jsem na dosah?“ a přetáhni jej do blízkosti bloku „Když“
- Prohlédni si tento nový blok. Jak by mohl fungovat v kombinaci s blokem „Když“?

--

- Nyní se zamysli. Pokud bude podmínka splněna, ze kterého zobáčku bude pokračovat signál?
- Pokud nebude podmínka splněna, ze kterého zobáčku bude pokračovat signál?
- Pokus se sestavit scénář tak, aby Honza zastavil u melounu potom, co zareaguje, že je na dosah. Pokud se Honza nechce zastavit, popřemýšlej ještě jednou nad otázkou „Pokud nebude podmínka splněna, ze kterého zobáčku bude pokračovat signál?“
- Je potřeba si uvědomit, že si Honza potřebuje neustále pokládat otázku „Jsem už na dosah?“. Podívej se na toto zapojení:



OBRÁZEK 68 ZAPOJENÍ PODMÍNKY S DOTAZEM, ZDA JE POSTAVA NA DOSAH

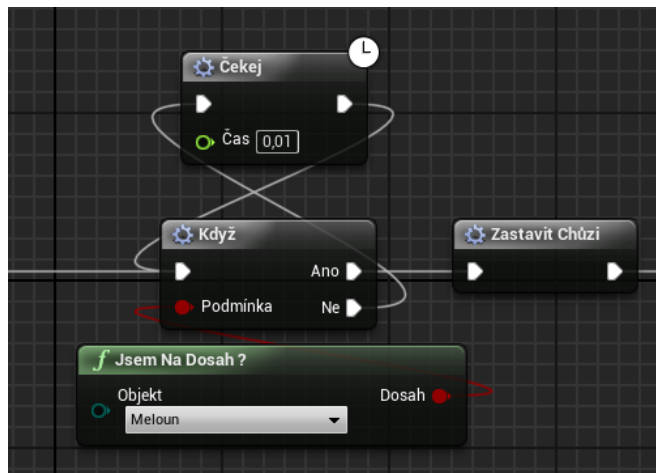
- Co se nyní stane? (Napřed odpověz, až poté čti dále)


- Můžou nastat dvě varianty:

První varianta je, že meloun je v dosahu Honzy. V takovém případě bude signál pokračovat přes „Ano“ do bloku „Zastavit Chůzi“ a Honza se zastaví

Druhá varianta je, že meloun v dosahu není. V takovém případě bude signál pokračovat přes „Ne“. Jelikož za výstupem „ne“ není žádný blok, scénář si myslí, že je hotovo a již nebude provádět žádné bloky

- Nyní se podívej na jedno z dalších využití bloku „Když“. Pohlédni na obrázek vpravo a pokus se vysvětlit, jak toto zapojení funguje



OBRÁZEK 69 57 ZAPOJENÍ PODMÍNKY S DOTAZEM, ZDA JE POSTAVA NA DOSAH S POUŽITÍM OPAKOVÁNÍ


- Uprav svůj scénář tak, aby odpovídal zapojení na obrázku
- Jak se nyní Honza chová a proč se tak chová?


- Pokus se vymyslet, proč je v tomto zapojení použit blok „Čekej“. Svůj názor prober s ostatními skupinami a až se shodnete na nějakém názoru, zkonzultujte tento názor s vyučujícím

## POZNÁMKY PRO VYUČJÍCÍHO

Podkapitola „Detekce melounu“ seznamuje žáka s možností zapojení bloku „Když“ jakožto detektoru přiblížení k položce.

Žáci mohou tuto podkapitolu dělat dvěma způsoby. První (a ten častější) je ten, že si žáci nedokáží poradit s úkolem, který jim zadává zastavit Honzu u melounu po vyhodnocení, že je meloun na dosah. Druhá varianta je ta, že žáci přijdou na to, že je potřeba vytvořit smyčku, ale nebudou vědět, že mají použít blok čekej. V obou případech vyučující vyzve žáky k tomu, ať tento úkol přeskočí a pokračují dále. Další body v zadání jsou vytvořeny k tomu, aby tento problém vysvětlily.

Vyučujícímu bude předložena teorie, proč je nutné využít blok opakování. Vyučující by měl o této teorii s žáky debatovat a postupně odhalovat pravý důvod. Ten je takový, že bez použití časové funkce je uskutečněna nekonečná smyčka v jednom okamžiku a program nemůže pokračovat.

Možné odpovědi na otázky:

- *Prohlédni si tento nový blok. Jak by mohl fungovat v kombinaci s blokem „Když“?*

Dá informaci, zda se Honza nachází poblíž určité položky či ne.

- *Co se nyní stane? (Napřed odpověz, až poté čti dále)*
  - *Viz následující body v zadání*
- *Nyní se podívej na jedno z dalších využití bloku „Když“. Pohlédni na obrázek vpravo a pokus se vysvětlit, jak toto zapojení funguje*

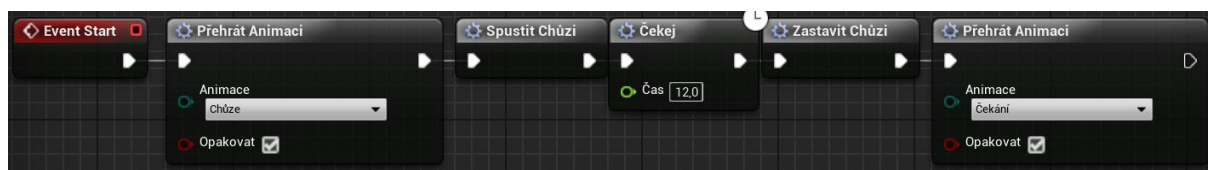
Pokud Honza stojí u melounu, scénář zastaví chůzi

Pokud ne, chvíli se čeká a následně se znovu opakuje dotaz

- *Jak se nyní Honza chová a proč se tak chová?*

Honza jde k melounu a u něj zastaví,

protože je vyhodnoceno, že se Honza nachází u melounu ve chvíli, kdy se tam nachází



OBRÁZEK 70 SCÉNÁŘ PRO CHŮZI HONZI K MELOUNU

## SBÍRÁNÍ POLOŽEK

- Nyní zajisti, aby si Honza vzal meloun pomocí bloku „Sebrat položku“
- Teď zaříd', aby se Honza po přiblížení se k melounu (ještě před sebráním melounu) rozhlédl
- Obrat' svou pozornost na blok ze seznamu „Functions“ s názvem „Přehrává se animace?“. S ním v kombinaci s blokem „Když“ zajisti, aby se meloun sebral až potom, co se Honza rozhlédne. Nezapomeň se opět zamyslet nad otázkami, co se stane, pokud bude podmínka splněna a naopak
- Následně zaříd', aby Honza došel k rybám, a sebral sedm ryb.
- Nyní zajisti, aby po sebrání ryb Honza došel ke Klaudivii.
- Ve chvíli, co k ní dojde zajisti, aby Honza řekl, co všechno sebral pomocí bloku „Vypsat nesené položky“
- Vytvoř svůj vlastní seznam položek a vyšli Honzu na nákup pro Klaudivii

## POZNÁMKY PRO VYUČUJÍCÍHO

*V podkapitole „Sbírání položek“ je trénováno použití bloku „Když“ jakožto detektoru.*

*Problém mohou žáci mít s využitím bloku „Přehrává se animace?“. Tento problém si však většinou žáci vyřeší v rámci jednotlivých skupin.*

*Pokud žáci nepoužijí opakování pro sebrání sedmi ryb, je vhodné jim připomenout, že taková možnost existuje.*

*Poslední úkol je určen pro nejrychlejší skupiny. Není potřeba, aby jej splnily všechny.*

## 5.7. Část 6. Parametry

### Cíl výuky

- Žák dokáže přidat do vlastních bloků parametry a následně je využívat

### Témata lekcí

- Vylepšování scénáře pro pohyb Klaudie
- Vylepšování Kvízu

### 5.7.1. Lekce 1. Rozhýbat dámu na úrovni

Orientační časová dotace: 35 minut

Využitá úroveň: Pohyb

V lekci žáci upravují scénář Klaudie s využitím parametrů.

#### Jednotlivé aktivity

##### VYLEPŠENÍ SCÉNÁŘE

- Otevři složku „Pohyb“, a v ní spust' soubor „Pohyb“
- Spust' soubor „Klaudie\_Pohyb“
- Prohlédni si scénář, který jsi kdysi vytvořil/a. Vybav si, jak funguje.
- Popřemýšlej, jak by se dal tento scénář vylepšit. Své domněnky prodiskutuj s ostatními skupinami a následně s vyučujícím
- Uprav scénář podle těchto nápadů
- Pokud jsi tak neučinil/a, vytvoř čtyři nové bloky, jeden pro každou část chůze tak, aby ve scénáři byly pouze 4 bloky (krom bloku Event Start)
- Co by nyní bylo potřeba udělat, aby Klaudie po obejití prvního čtverce obešla druhý, ale se vzdáleností stran 700?

- Co by bylo potřeba, aby následně obešla další čtverec se vzdáleností stran 1000?

- Kolik nových bloků by bylo potřeba?



- Existuje způsob, díky kterému ti na všechny tyto i další případy postačí pouze jeden blok

### POZNÁMKY PRO VYUČUJÍCÍ

Podkapitola „Vylepšení scénáře“ je zaměřena na schopnost žáků upravit scénáře aplikováním prvků jako jsou cykly či vlastní bloky. Podkapitola je zde přidána z důvodu tvorby podkladů pro následující části lekce.

Vyučujícím budou předneseny návrhy na vylepšení scénáře. Mezi nimi by měla být především aplikace opakování a tvorba vlastních bloků. Vyučující může s žáky na toto téma debatovat.

Otázky této podkapitoly jsou přidány pro nastínění problému bloků bez parametrů.

Možné odpovědi na otázky:

- Co by nyní bylo potřeba udělat, aby Klaudie po obejití prvního čtverce obešla druhý, ale se vzdáleností stran 700?

Vytvořit další bloky

- Co by bylo potřeba, aby následně obešla další čtverec se vzdáleností stran 1000?

Opět vytvořit další bloky

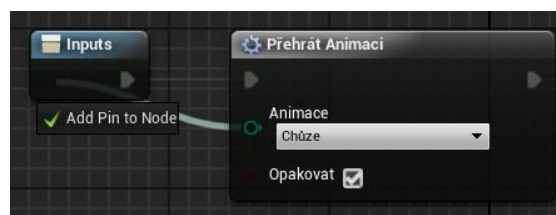
- Kolik nových bloků by bylo potřeba?

8

### PARAMETRY

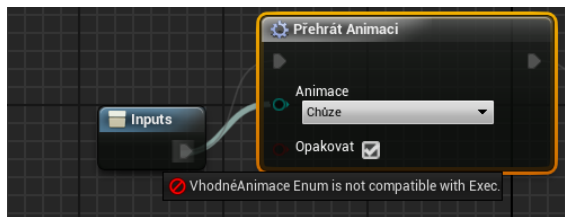
- Smaž ze seznamu „Macros“ tři ze čtyř bloků které jsi vytvořil. (klikni na něj v seznamu a stiskni tlačítko Delete)
- Poslední blok otevři dvojklikem
- Jak bys přeložil/a slovo „Inputs“?

- Nyní chytň zobáček Animace u bloku „Přehrát animaci“ a přetáhni jej do bloku „Inputs“ podobně, jako vidíš na obrázku vpravo. Dej si pozor na to, že myš můžeš pustit až ve chvíli, co se na kurzorem objeví zelená fajfka. Toho docílíš tak, že s kurzorem najedeš kamkoliv do prostoru bloku „Inputs“ kromě zobáčku



OBRÁZEK 71 VYTVÁŘENÍ PARAMETRU

- S pomocí takto upraveného tvého bloku sestav scénář, který zařídí, aby Klaudie šla čtvrtinu cesty normálně, čtvrtinu cesty s tancem, čtvrtinu cesty s rozhlížením se, čtvrtinu cesty šťastně a pokaždé se otočila doleva (aby šla do čtverce stejně jako v předchozím případě). Nepoužívej při tom žádné jiné bloky než tvůj blok



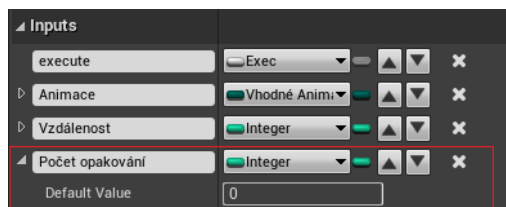
OBRÁZEK 72 CHYBNÉ VYTVOŘENÍ PARAMETRU

- Co se změnilo napojením zobáčku „Animace“ do bloku „Inputs“?

- Nyní stejným způsobem napoj do bloku „Inputs“ zobáček „Vzdálenost“ z bloku „Jít dopředu“
- Uprav scénář tak, aby jeden úsek šla Klaudie 200 jednotek, druhý 400, třetí 600, a čtvrtý 800
- Nakonec připoj i zobáček „Počet Opakování“ z bloku „Opakovat“
- Co se bude dít, když tuto hodnotu budeme měnit od 0 do 6? Odpověz a následně ověř

- V tuto chvíli se zobáček z bloku „Inputs“ jmenuje stejně jako zobáček z bloku „Opakovat“, tedy „Počet opakování“. Vymysli lepší název pro tento zobáček, který by opravdu vystihoval to, co se stane, když se změní jeho hodnota

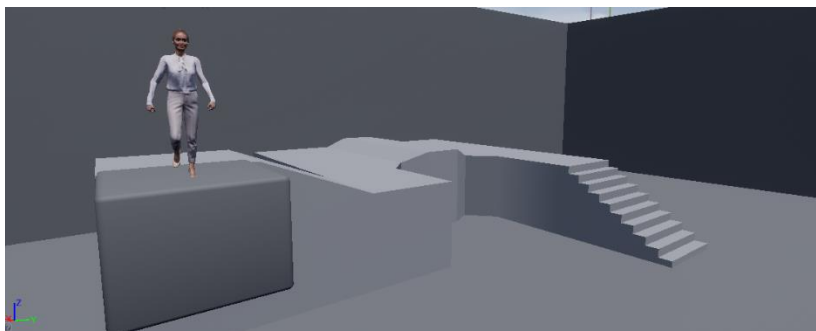
- Nyní klikni na blok „Inputs“
- Nyní se ti v pravé části obrazovky objeví seznam pojmenovaný „Inputs“ se všemi zobáčky, které obsahuje. Změň název zobáčku „Počet opakování“ na název, který jsi vymyslel/a



OBRÁZEK 73 DETAIL PARAMETRŮ

- Pokud chceš, můžeš kliknout na zobáček vedle tohoto názvu, a tím změnit, jakou hodnotu bude tento zobáček mít při přetažení ze seznamu „Macros“ do scénáře místo nuly. Stejně tak můžeš změnit i ostatní zobáčky

- Pouze za pomoci tvého bloku sestav scénář, ve kterém Klaudie dojde až na konec plošiny, jež je v mapě podobně jako na obrázku vpravo

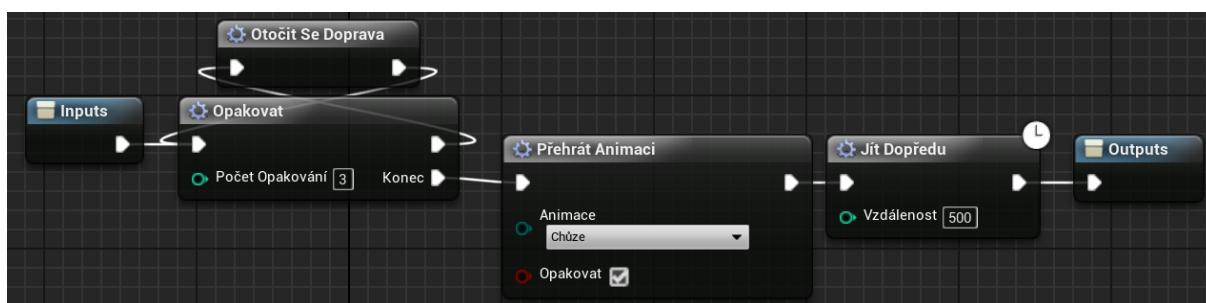


OBRÁZEK 74 UKÁZKA KLAUDIE NA PLOŠINĚ

### POZNÁMKY PRO VYUČUJÍCÍ

V podkapitole „Parametry“ jsou žáci seznámeni s možnostmi vstupních parametrů v jejich vlastních blocích. Tyto poznatky jsou průběžně trénovány.

Žáci mohou mít problémy v případě, že v předchozí podkapitole špatně sestavili svůj blok. Pokud se tak stane, vyučující může žákům pomoci vymyslet, jak by scénář takového bloku mohl vypadat.



OBRÁZEK 75 MOŽNÉ SCHÉMA SCÉNÁŘE BLOKU

Žáci mohou mít problém s pojmenováním parametrů. Je vhodné je v takové chvíli navést, aby popsali to, co daný parametr provádí za změny. Podle toho žáci určí název.

Možné odpovědi na otázky:

- Jak bys přeložil/a slovo „Inputs“?

Vstupy

- Co se změnilo napojením zobáčku „Animace“ do bloku „Inputs“?

Mohu měnit animaci přímo ze scénáře

- Co se bude dít, když tuto hodnotu budeme měnit od 0 do 6? Odpověz a následně ověř

Klaudia se otočí doprava podle tohoto čísla

## 5.7.2. Lekce 2. Vylepšujeme

Orientační časová dotace: 25 minut

Využitá úroveň: Cesta\_S\_Klaudii

V lekci žáci upravují scénář Klaudie s využitím parametrů.

### Jednotlivé aktivity

- Otevři složku „Cesta“, a v ní spust' soubor „Cesta\_S\_Klaudii“
- Spust' soubor „Klaudie\_Cesta“
- Uprav scénář tak, aby se jeho funkce nezměnila, ale skládal se pouze z bloků „Event Start“, „Přehrát Animaci“, „Opakovat“ a jednoho tvého bloku, který můžeš vytvořit z jakýchkoliv bloků, jež má tato verze Klaudie k dispozici. Podmínkou je, že výsledný scénář nesmí být delší než 9 bloků a kratší než 5 bloků

#### POZNÁMKY PRO VYUČUJÍCÍHO

*Tato lekce nutí žáky přemýšlet nad možnými způsoby rozčlenění scénáře na bloky s parametry. Je vhodná pro vyhlášení jakožto soutěže mezi jednotlivými skupinami. Kritéria vyhodnocení mohou být nejrychlejší čas, nejpřehlednější scénář, nejméně použitých bloků. Opět je podstatné ocenit i pomalejší skupiny.*

## 5.7.3. Lekce 3. Kvíz V2.

Orientační časová dotace: 60 minut

Využitá úroveň: Prázdnota

V lekci žáci vytvářejí blok, s jehož pomocí lze rychle zadávat a vyhodnocovat otázky.

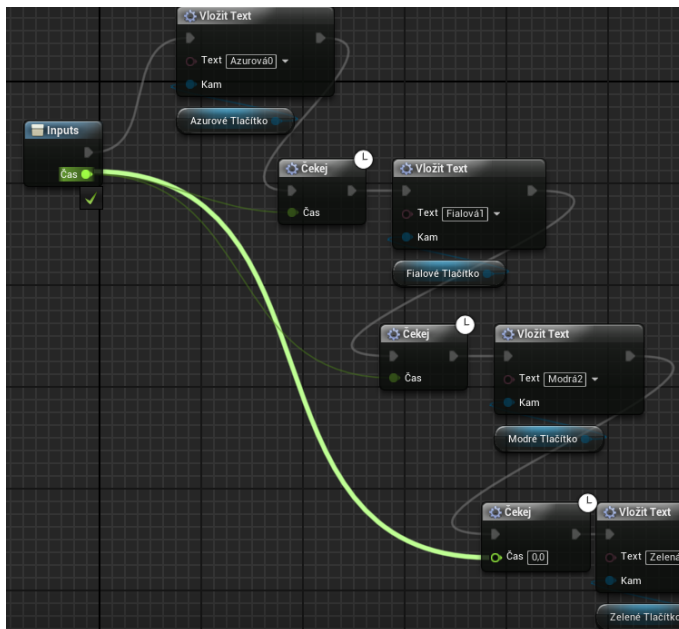
### Jednotlivé aktivity

#### SDÍLENÉ VSTUPY

- Otevři složku „Chat“, a v ní spust' soubor „Prázdnota“
- Spust' soubor „Chat“
- Nalezni blok pro vkládání textu do tlačítek, jenž jsi zde kdysi vytvořil/a
- Uprav tento blok tak, aby se mohl čas mezi objevením textu prvního a druhého tlačítka měnit přímo ve scénáři

- Kolik takto nazelenalých zobáčků by bylo potřeba, kdyby si chtěl/a ze scénáře nastavovat všechny časy se stejnou hodnotou? (př. Všechny časy na 1s nebo 2s)

- Správná odpověď je, že stačí pouze jeden. Použijeme totiž vlastnost těchto propojení. Existuje pravidlo, které říká, že pokud se nejedná o bílé propojení (to propojení, které udává jednotlivé kroky scénáře), lze z jednoho zobáčku, který dává informace vytáhnout libovolné množství propojení. Napoj tedy všechny „Časy“ ze všech bloků „Čekej“ do tohoto zobáčku z bloku „Inputs“ podobně, jak je tomu na obrázku



OBRÁZEK 76 SDÍLENÍ VSTUPŮ V BLOKU

- Zobáčky lze odstranit kliknutím na blok „Inputs“ a následným stisknutím křížku vedle daného zobáčku v nabídce.



OBRÁZEK 77 DETAIL PARAMETRU

- Co se stane, když ve scénáři tomuto bloku nastavíš 0, pak 0,5, a poté 2? Odpověz a následně ověř

- Jak nyní upraviš tento blok, když budeš chtít těmto tlačítkům přiřadit i slova (každému tlačítku jiné slovo)? Napřed promysli, následně udělej a otestuj

- Vhodně přejmenuj zobáčky v bloku „Inputs“

### POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Sdílené vstupy“ ukazuje možnosti využívání jednoho vstupu, pro více bloků.

Při postupu v podkapitole bude po žácích vyžadováno, aby nejprve zapojili parametry bez jejich sdílení. Až následně je jim představen tento koncept. Některé skupiny budou schopné sdílet parametry intuitivně, již před představením konceptu.

*Při dokončování podkapitoly je vhodné, aby vyučující zkontroloval, jak si jednotlivé skupiny počínaly. Případně může vyučující žáky vyzvat k tomu, aby mu jejich blok demonstrovali.*

*Možné odpovědi na otázky:*

- Kolik takto nazelenalých zobáčků by bylo potřeba, kdyby si chtěl/a ze scénáře nastavovat všechny časy se stejnou hodnotou? (př. Všechny časy na 1s nebo 2s)*

5;1

- Co se stane, když ve scénáři tomuto bloku nastaviš různé hodnoty? Odpověz a následně ověř*

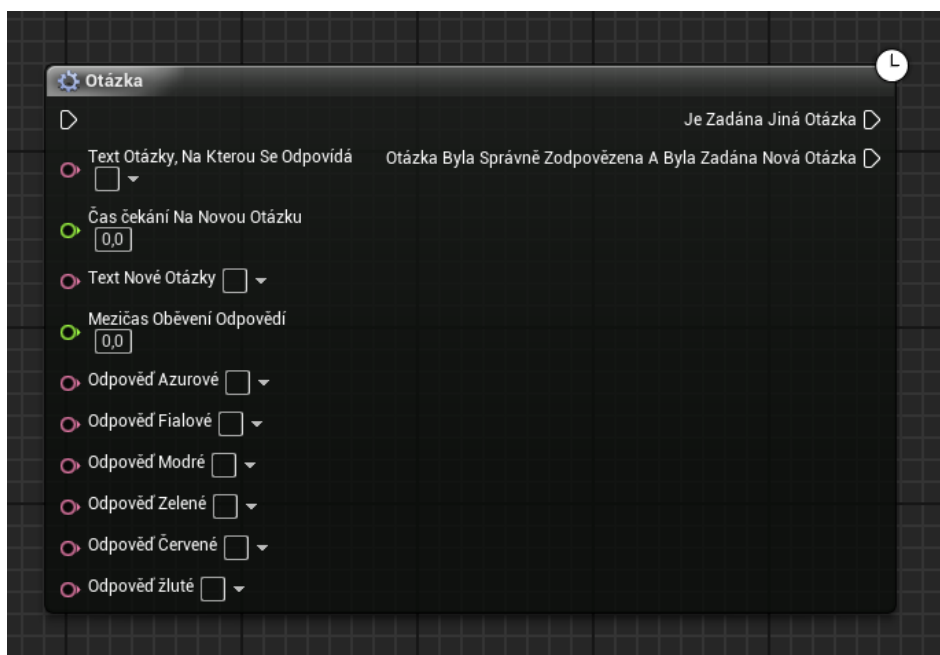
Všechny časy objevování budou mít stejnou délku podle této hodnoty

- Jak nyní upraviš tento blok, když budeš chtít těmto tlačítkům přiřadit i slova (každému tlačítku jiné slovo)? Napřed promysli, následně udělej a otestuj*

Každému slovu vytvořím nový zobáček

#### BLOK OTÁZEK

- Odpoj všechny otázky, jež jsi doposud stvořil/a
- Následně zařid', aby se po stisknutí tlačítka „Play“ do chatu zadala otázka a na všech tlačítkách byla nějaká odpověď
- Vytvoř blok, který nejdříve zkontroluje, zda se odpovídá na otázku, pro kterou je určen. Pokud ne, nic neudělá, jen pošle signál ven z bloku. Pokud ano, pochválí tě za stisknutí správného tlačítka. Následně čeká dobu, kterou mu ve scénáři nastaviš. Poté zadá novou otázku, kterou mu zadáš ze scénáře. Následně všem tlačítkům nastaví možné odpovědi, které budou zadávány taktéž ve scénáři (neostýchej se využít i blok, který jsi již vytvořil/a). Na závěr pošle signál z bloku ven jiným zobáčkem než v případě, kdy tento blok nebyl určen pro tuto otázku. Možný výsledný vzhled bloku můžeš vidět na obrázku níže.



OBRÁZEK 78 MOŽNÝ VZHLED BLOKU OTÁZKY

- Otestuj tento blok se třemi otázkami

#### POZNÁMKY PRO VYUČUJÍCÍHO

*V této podkapitole žáci vytvářejí blok, který je již poměrně komplexní.*

*Část skupin bude mít s řešením tohoto úkolu problémy. Při takovémto problému je vhodné, aby vyučující s žáky prošel jednotlivé kroky, které povedou k realizaci bloku. Vyučující by však neměl vymýšlet samotný scénář, měl by se pouze dotazovat na to, co musí po čem následovat, aby situace fungovala správně.*

*Je důležité testovat tento blok pro více otázek. Při tomto testování žáci přemýšlejí o tom, kam má být blok zapojen a v jakém pořadí. Často žáci přijdou s vlastním řešením, které je správné, a neměli by za toto řešení být trestáni. Místo toho by měli dostat pochvalu, ať už daný problém vyřešili jakkoliv.*

*Po splnění úkolů by žáci měli dostat volnost k tomu, aby si vzájemně vytvářeli kvízy. Zvýší to jejich motivaci věnovat se programování.*

## 5.8. Část 7. Volané události

### Cíl výuky

- Žák dokáže vytvářet a používat volané události pro komunikaci mezi objekty

### Témata lekcí

- Otáčet se pomocí budov a postav
- Předávat si informace mezi postavami

### 5.8.1. Lekce 1. Nákup za chodu

Orientační časová dotace: 45 minut

Využitá úroveň: Nákup

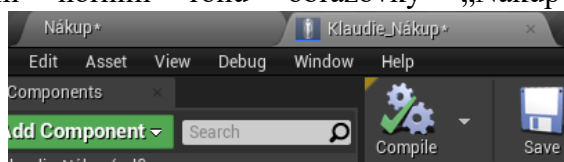
V lekci žáci vytváří volanou událost, jež slouží k otáčení postavy

### Jednotlivé aktivity

#### ZÁLOŽKY

- Otevři složku „Nákup“, a v ní spusť soubor „Nákup“
- Spusť soubor „Klaudie\_Nákup“
- Povšimni si dvou tlačítek v levém horním rohu obrazovky „Nákup“ a „Klaudie\_Nákup“. Otestuj, co dělají.

- Tato tlačítka plní podobnou funkci jako záložky ve webovém prohlížeči (Chrome, Edge, ...).



OBRÁZEK 79 LIŠTA ZÁLOŽEK

Na každé z nich je nějaká stránka. V našem případě tlačítko „Nákup“ skrývá stránku, na které je mapa nákupu spolu s hlavním menu a tlačítko „Klaudie\_Nákup“ skrývá stránku, která dává Klaudii instrukce

- Nyní spusť soubory „Brána“ a „Honza\_Nákup“
- Popiš, co která záložka obsahuje


#### POZNÁMKY PRO VYUČUJÍCÍHO

V podkapitole „Záložky“ jsou žákům představeny záložky, které zde přepínají mezi vhledy jednotlivých postav a chatu.



Většina žáků již tyto záložky používala a zná je, avšak nebyly použity pro více objektů. Je proto dobré tuto problematiku s žáky řešit.

Možné odpovědi na otázky:

- Popiš, co která záložka obsahuje

Každá záložka, kromě záložky Nákup, obsahuje scénáře pro danou postavu .
--

Záložka Nákup obsahuje hlavní menu; vzhled mapy; seznam úkolů; ...
--

#### SBĚR POLOŽEK

- Klikni na tu záložku, která obsahuje Honzovy scénáře
- Vytvoř nový blok, po jehož stisknutí Honza dojde k položce, zastaví u ní a následně ji sebere. Využij k tomu „Inputs“
- Nyní vytvoř scénář, který v tomto pořadí sebere dýni, rybu, rajče a houbu pouze za použití tvého nového bloku. Pokud se ti to podaří, informuj o tom vyučujícího. Pokud narazíš na problém, čti dále
  - Patrně se objevil problém, že po sebrání ryby Honza šel dále a spadnul do bezedné propasti. Je proto nutné zajistit, aby se Honza před koncem cesty otočil. Tento problém by šel vyřešit způsobem, ve kterém bude Honza neustále kontrolovat, zda došel k poslední položce a následně se otočil. Nicméně si ukážeme jiný způsob, při kterém Honza nebude muset nic kontrolovat. Kontrolovat to místo něj bude v tomto případě brána na konci obchodu

#### POZNÁMKY PRO VYUČJÍCÍHO

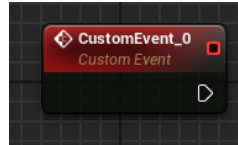
Podkapitola „Sběr položek“ je určena k nastínění problému, který se bude v lekci řešit.

Žáci budou vyzváni ke splnění úkolu, který bez znalosti volaných událostí není možné splnit. Úkol sice informuje žáky o tom, že při nesplnění úkolu mají pokračovat dále, avšak je možné, že část žáků tuto instrukci neuposlechne. Vyučující by je měl v takovémto případě naorientovat.

## VYTVORENÍ VOLANÉ UDÁLOSTI

- Nyní se vrať zpět na Honzovu záložku
- Klikni zde pravým tlačítkem myši někam do prázdna podobně, jako když jsi zajišťoval, abys mohl/a ovládat kostlivce
- Napiš text „custom event“ a stiskni Enter

- Ve scénáři se objevil nový blok rudé barvy jménem „CustomEvent\_0“, podobný všem blokům, ze kterých startuje scénář. Pojmenuj jej

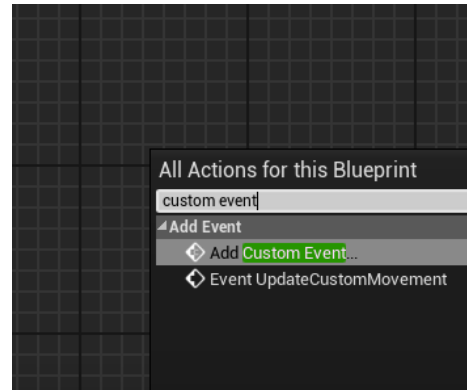


OBRÁZEK 81 VOLANÁ UDÁLOST

„Otočit Honzu“ Při přejmenovávání musí být název bloku označen bílým pozadím tak, jak je tomu na obrázku. Toho docílíš třemi rychlými kliknutími na název bloku

- Za tímto blokem vytvoř jednoduchý scénář, který nastaví Honzův úhel na -180
- Stiskni tlačítko „Compile“ umístěné v levém horním rohu

- Stisknutím tohoto tlačítka dávaš najevo, že jsi vytvořil blok „custom event“, neboli blok, který může kdokoliv zavolat



OBRÁZEK 80 VYTVORENÍ VOLANÉ UDÁLOSTI



OBRÁZEK 82 VOLANÁ UDÁLOST, PŘEJMENOVÁVÁNÍ



OBRÁZEK 83 TLAČÍTKO COMPILE

## POZNÁMKY PRO VYUČUJÍCÍHO

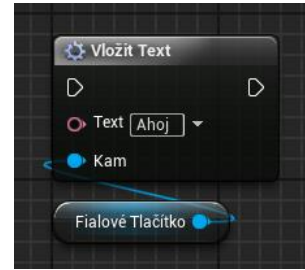
Podkapitola „Sběr položek“ je vytvořena jakožto návod pro vytvoření volané události.

Žáci mohou mít problém samotnou událost do scénáře přidat. UE4 naneštěstí nepodporuje jednodušší způsob vkládání volaných událostí. Je proto možné žákům při vkládání události pomoci.

## VOLÁNÍ UDÁLOSTI

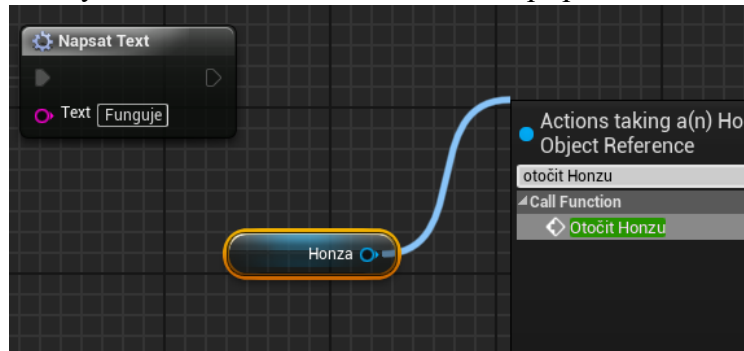
- Klikni nyní na záložku pro bránu
- Z bloků, které má brána k dispozici vytvoř scénář, který bude neustále hlídat, zda není Honza na dosah brány. Zda tvůj scénář funguje můžeš ověřit blokem „Napsat text“
- V seznamu „Variables“ je blok „Honza“ Přetáhni jej do blízkosti tvého scénáře stejným způsobem, jakým jsi přetahoval/a bloky „Fialové\_tlačítko“ apod.

- Tento blok je takovým odkazem na Honzu podobně, jako blok „Fialové\_tlačítko“ je odkazem na opravdové fialové tlačítko. Kdysi jsi pomocí tohoto bloku v kombinaci s blokem „Vložit text“ nastavoval, jaké bude fialové tlačítko obsahovat text, viz obrázek. V případě bloku Honza je to podobné. Tím že použiješ blok „Honza“, můžeš Honzovi posílat informace stejným způsobem



OBRÁZEK 84 VKLÁDÁNÍ TEXTU

- Před chvílkou jsi v Honzových scénářích vytvořil blok „Otočit Honzu“. Nyní jej využijeme. Z bloku „Honza“ vytáhni do prázdna propojení
- Následně napiš název bloku, se kterým chceme komunikovat, v našem případě s blokem „Otočit Honzu“ viz obrázek



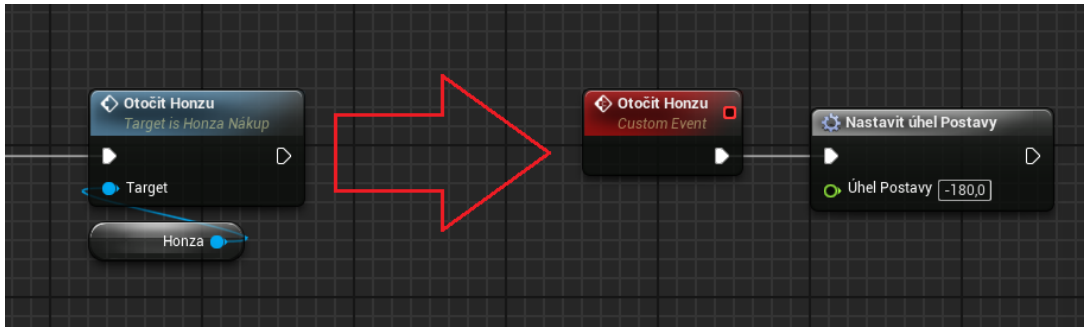
OBRÁZEK 85 VÝBĚR HONZOVÝCH UDÁLOSTÍ

- Stiskni Enter
- Nový blok, který se ti objevil napoj do scénáře tak, aby se aktivoval v tu chvíli, co bude Honza na dosah brány

- Otestuj, co se stane
- Pokus se odpovědět, jak vlastně blok „Otočit Honzu“ funguje

--

- V základu funguje prostě. V Honzových scénářích je vytvořen blok, který může být kdykoliv a odkudkoliv aktivován. Rozdíl oproti bloku, který se spustí po stisknutí tlačítka, je ten, že místo podmínky „zapni se po stisknutí tlačítka“, je zde podmínka „zapni se, až tě někdo (něco) zavolá z jiného scénáře“. Tím, že Honza vstoupil do blízkosti brány, aktivoval blok „Otočit Honzu“, který je aktivován v Honzových scénářích. To je důvod, proč je k bloku, který je volán, připojen blok „Honza“. Je totiž umístěn v Honzových scénářích



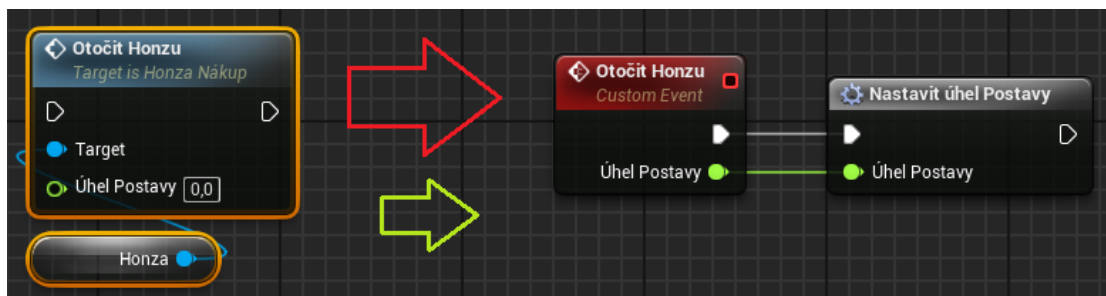
OBRÁZEK 87 PRINCIP VOLANÝCH UDÁLOSTÍ 1

- Nyní zatáhni za zobáček „Úhel postavy“ a propoj jej s blokem „Otočit Honzu“ podobně, jako jsi propojoval/a zobáčky s blokem „Inputs“



OBRÁZEK 86 VYTVOŘENÍ PARAMETRU

- Vždy, když provádíš změny v červených blocích, stiskni po provedení změn tlačítko „Compile“
- Podívej se do scénářů brány, jaké se provedly změny
- Zajisti, aby se Honza opět otáčel tak, jako před způsobem této změny
  - Úplně stejným způsobem, jakým se posílá propojení bílé čáry (signalizace, jsi na řadě. Spust' se) lze posílat i propojení jakékoliv jiné, ať už je to číslo, slovo, či něco jiného. Z tohoto pohledu se dá říci, že bloky „Custom Event“ jsou velice podobné blokům „Inputs“



OBRÁZEK 88 PRINCIP VOLANÝCH UDÁLOSTÍ 2

- Podobným způsobem zajisti otočení i na druhé straně obchodu. Využij k tomu Klaudivii. Neboj se opět použít blok „Otočit Honzu“, který jsi již vytvořil/a
- Na závěr zajisti, aby po sebrání všech předmětů Honza došel ke Klaudivii, vypsál položky, které nese a padl vyčerpáním

## POZNÁMKY PRO VYUČJÍCÍHO

Podkapitola „Volání události“ ukazuje způsob, jakým lze volat události.

V prostředí UE4 se jedná již o poměrně složitou záležitost, proto žáci nemusí hned pochopit, jak volání událostí funguje. Vyučující se může pokusit vysvětlit způsob fungování volaných událostí svými slovy. Může také žáky odkázat na odrážku pod jedinou otázkou této podkapitoly a debatovat o ní.

Při vytváření scénáře pro Klaudii mohou žáci vytvářet novou událost. V této chvíli to sice není zapotřebí, ale procvičí si tím způsob tvoření událostí. Proto je vhodné nechat žáky své řešení dokončit, a až poté jim předvést možnost volání jedné události z více míst.

U otázky není nutně vyžadována správná odpověď. Otázka je zde umístěna z důvodu zamýšlení nad problematikou volání událostí.

Možné odpovědi na otázky:

- Pokus se odpovědět, jak vlastně blok „Otočit Honzu“ funguje.

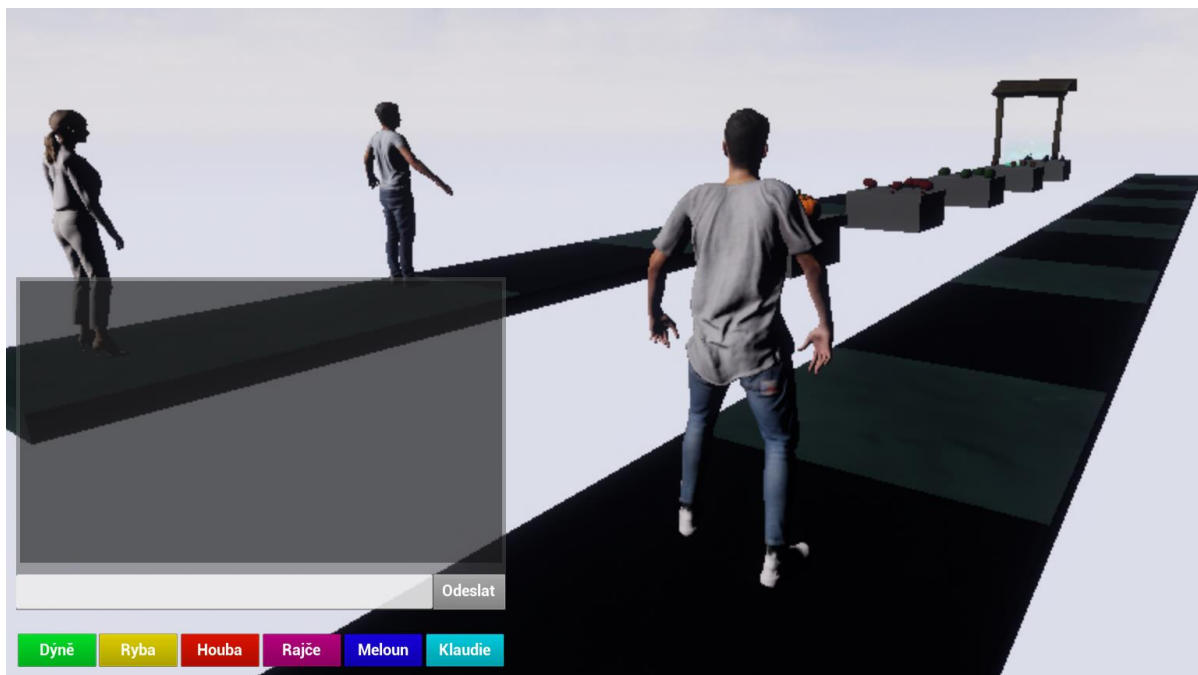
Posílá signál Honzovi, že má aktivovat tento blok.

## 5.8.2. Lekce 2. Nákup na vyžádání

Orientační časová dotace: 50 minut

Využitá úroveň: Nákup\_V2

V lekci žáci propojují stisknutí tlačítek s Honzovým chováním



OBRÁZEK 89 UKÁZKA ÚROVNĚ NÁKUP\_V2

### Jednotlivé aktivity

#### PROHLÍDKA MAPY

- Otevři složku „Nákup“, a v ní spusť soubor „Nákup\_V2“
- Stiskni tlačítko „Play“
  - V tomto režimu světa je navíc přidán chat. Tento chat však neobsahuje žádné scénáře, jež jsi vytvořil/a
  - Otáčet se s Emanuelem (s postavou, kterou ovládáš) můžeš podržením levého tlačítka myši a následným pohybem kurzoru

#### NÁKUP PŘES TLAČÍTKA

- Zajisti, aby po stisknutí „Play“ Honza stál na místě a nic nedělal
- Vytvoř nový scénář s použitím bloku „Custom Event“ který po zavolání dovede Honzu k dýni, u které se zastaví, a kterou následně Honza sebere.

- Nyní zajisti, aby zelené tlačítko obsahovalo text „Dýně“ a po jeho stisknutí se:
  1. Do chatu napsal text „Honza jde pro dýni“
  2. Honza vyrazil sebrat dýni (Odkaz na Honzu nalezněš v seznamu „Variables“ kliknutím na zobáček u „Postavy“)
- Zajisti, aby se po sebrání dýně napsalo do chatu „Předmět sebrán“
- Podobným způsobem nastav i zbylá tlačítka tak, aby při stisknutí jiného tlačítka Honza sebral jinou položku (Po stisknutí tlačítka s textem „Meloun“ Honza došel pro meloun, ...). Nevytvářej při tom žádné nové volané události
- Poslední volné tlačítko nastav tak, aby po jeho stisku Honza došel ke Klaudii a následně jí předal předměty, které Honza nasbíral. Využij k tomu blok „Vzít si položky“, který má Klaudie v seznamu bloků. (Nyní již můžeš použít nové volané události)
- Zajisti, aby po předání předmětů Klaudie vypsala předměty, jež jí Honza předal pomocí bloku „Vypsat nesené předměty“
- Otestuj pro různé předměty

#### POZNÁMKY PRO VYUČUJÍCÍHO

*Podkapitola „Nákup přes tlačítka“ trénuje aplikaci událostí.*

*Při vytváření volaných událostí žáci zapominají na stisknutí tlačítka „Compile“, díky čemuž následně nevidí svou událost v nabídce.*

*Při prvním testování volané události žáci většinou využívají scénář Event Start. Někteří žáci však nebudou vědět, jak svůj scénář otestovat. Je proto vhodné pokládat cílené otázky tak, aby žákům došlo, jaké mají možnosti.*

*Žákům může plnění jednotlivých úkolů trvat déle, či se na nich mohou zastavit. Podkapitolka je vytvořena tak, aby žáci museli přemýšlet, jak jednotlivé události využít. V případě, že se žáci zastaví na nějakém problému, vyučující může žáky navést na správné řešení pomocí cílených otázek, neměl by však za žáky vymýšlet kód.*

## 5.9. Část 8. Proměnné

### Cíl výuky

- Žák dokáže vytvářet a používat proměnné

### Témata lekcí

- Počítání zodpovězených úloh
- Komunikace postav

### 5.9.1. Lekce 1. Počty

Orientační časová dotace: 50 minut

Využitá úroveň: Prázdnota

V lekci žáci vytváří proměnnou, se kterou se seznamují. Postupně vytvářejí čítač správně zodpovězených otázek.

### Jednotlivé aktivity

#### TYPY ZOBÁČKŮ

- Doposud jsi používal/a 7 barev zobáčků. Popiš, co která barva znamená (pokud si nejsi jistý/á, zapátrej v předchozích lekcích)




OBRÁZEK 90  
VYUŽÍVANÉ  
ZOBÁČKY

- Představ si situaci, ve které budeme chtít počítat, kolik otázek se ti podařilo uhodnout. K tomuto účelu by se hodil blok, který v sobě ukrývá nějakou hodnotu (v našem případě číslo), kterou lze měnit. Takový blok naštěstí existuje



## POZNÁMKY PRO VYUČJÍCÍHO

V podkapitole „Typy zobáčků“ mají žáci za úkol popsat, jakou hodnotu či příkaz v sobě skrývají jednotlivé barvy zobáčků.

V této podkapitole je vhodné, aby vyučující s žáky diskutoval nad možnostmi jednotlivých typů zobáčků, popřípadě aby žákům dopomohl pochopit, co který zobáček ukrývá. Tuto pomoc je vhodné dělat formou procházení předešlých lekcí a hledáním zobáčků stejné barvy.

Žáci mohou mít problém s rozeznáním zobáčků typu float a integer. Tento rozdíl je vhodné demonstrovat na bloku „Čekej“ a bloku „Opakovat“ spolu s dotazováním, proč je důležité, aby tyto bloky měly své typy takto.

Žáci mohou tápat nad otázkou, jakou hodnotu v sobě ukrývá tmavě zelený zobáček (enum). Pokud je tento problém na úrovni celé třídy, lze tento problém řešit kolektivně. Vyučující může po žácích chtít slyšet nápady, na které následně patřičně reaguje.

Možné odpovědi na otázky:

- Doposud jsi používal/a 7 barev zobáčků. Popiš, co která barva znamená (pokud si nejsi jistý/á, zapátrej v předchozích lekcích)

= udává pořadí, v jakém se bloky spouští
= obsahuje, zda je podmínka splněna nebo ne
= obsahuje určitý seznam slov; seznam položek; seznam věcí
= obsahuje celé číslo
= obsahuje jakékoliv reálné číslo
= obsahuje slova; text
= obsahuje postavu; odkaz na postavu; scénáře postavy; ...

## PŘÍPRAVA PRO PROMĚNNÉ

- Otevři složku „Chat“, a v ní spust' soubor „Prázdnota“
- Spust' soubor „Chat“
- Vytvoř scénář, který po stisknutí tlačítka „Tlačítko\_odeslat“ (šedé tlačítko) napíše do políčka „Napsaný\_text“ text „Počet správných otázek“
  - Zanedlouho zjistíš, že se místo tohoto textu bude psát počet správně zodpovězených otázek

## VYTVORENÍ PROMĚNNÉ

- Nyní se zaměř na seznam „Variables“. Přelož toto slovo

- Vedle názvu tohoto seznamu je tlačítko „+“. Klikni na něj



OBRÁZEK 91 SEZNAM VARIABLES

- Objevil se nový blok. Pojmenuj jej „Číslo“

- Všimni si, že okolo názvu tohoto nového bloku je červená čárka. Barva této čárky říká, jaký tip hodnoty se v tomto bloku skrývá (číslo, text, ano/ne, ...). Rozeznávání je stejné, jako u zobáčků.



OBRÁZEK 92 VZHLED PROMĚNNÉ

- Jaký typ hodnoty by v tuto chvíli mohl blok „Číslo“ obsahovat ?

### POZNÁMKY PRO VYUČUJÍCÍHO

*Při přejmenování proměnné žáci mohou mít problém, že si nepamatují postup přejmenování bloků (trojklik na jejich reprezentaci v daném seznamu). V tomto případě může vyučující žákům poradit.*

*Možné odpovědi na otázky:*

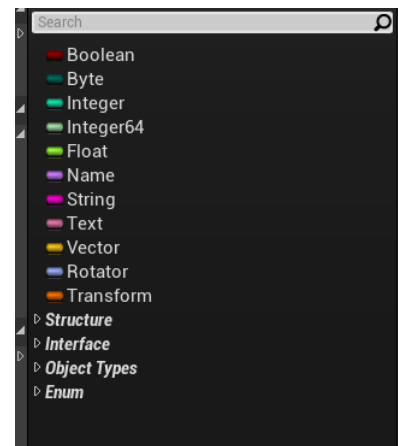
- Nyní se zaměř na seznam „Variables“. Přelož toto slovo

- Jaký typ hodnoty by v tuto chvíli mohl blok „Číslo“ obsahovat ?

### ZMĚNA DATOVÉHO TIPU

- Pro náš případ potřebujeme, aby tento blok obsahoval nějaké číslo. Klikni myší na červenou čárku umístěnou vedle tohoto bloku a z nabídky vyber tu barvu, o které výš, že je to barva označující celé číslo, viz obrázek vpravo

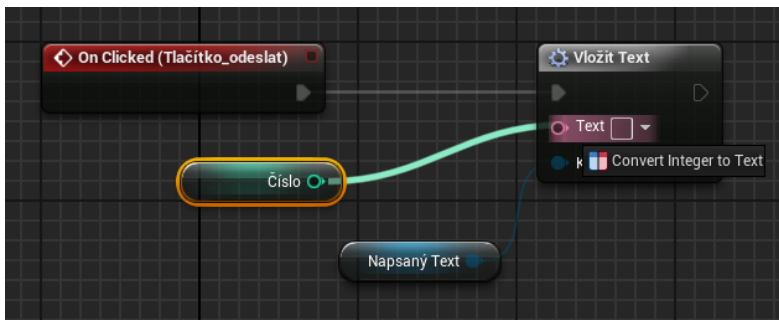
- Všimni si, že čárka u bloku změnila barvu



OBRÁZEK 93 NABÍDKA VÝBĚRU DATOVÉHO TIPU

## ČTENÍ PROMĚNNÝCH

- Přidej tento blok do blízkosti scénáře po stisknutí tlačítka „Tlačítko\_odeslat“ stejně, jako jsi přidával/a bloky odkazující na tlačítka (zvol možnost „Get Číslo“ )
- Propoj blok „Číslo“ s blokem „Vložit text,“ stejně, jako vidíš na obrázku níže
  - Mezi těmito bloky se objevil blok nový s názvem „ToText (integer)“ Tento blok má za úkol přetvořit číslo na text, který vyžaduje blok „Vložit text“



OBRÁZEK 94 PŘIPOJOVÁNÍ PROMĚNNÝCH

- Otestuj, co se nyní napíše po stisknutí tlačítka „Tlačítko\_odeslat“?

- Napsala se 0 a to z toho důvodu, že 0 je základní hodnota, kterou číselné bloky obsahují. Aby se vypsalo jiné číslo než 0, je potřeba hodnotu bloku změnit

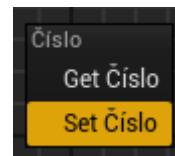
## POZNÁMKY PRO VYUČUJÍCÍHO

Možné odpovědi na otázky:

- Otestuj, co se nyní napíše po stisknutí tlačítka „Tlačítko\_odeslat“?

## ZÁPIS DO PROMĚNNÉ

- Přidej na konec scénáře blok „Číslo“, tentokrát ovšem se zvolenou možností „Set Číslo“
- Libovolně nastav hodnotu tohoto bloku na jiné číslo než 0
- 2x stiskni tlačítko „Tlačítko\_odeslat“ Jaká čísla se při jednotlivých stisknutí vypíšou a proč ?

- Zajisti, aby se sebou zvolené číslo objevilo již při prvním stisknutí tlačítka. Zajisti to pouze za použití tohoto scénáře

## POZNÁMKY PRO VYUČJÍCIHO

V podkapitole „Zápis do proměnné“ se žáci seznamují s variantou proměnné „set proměnná“.

Žáci mohou mít problém zodpovědět otázku. Vyučující může pomoci tím, že s žáky popíše, co se ve kterém momentu scénáře stane. Většina žáků pouze po tomto popisu nalezne odpověď.

Možné odpovědi na otázky:

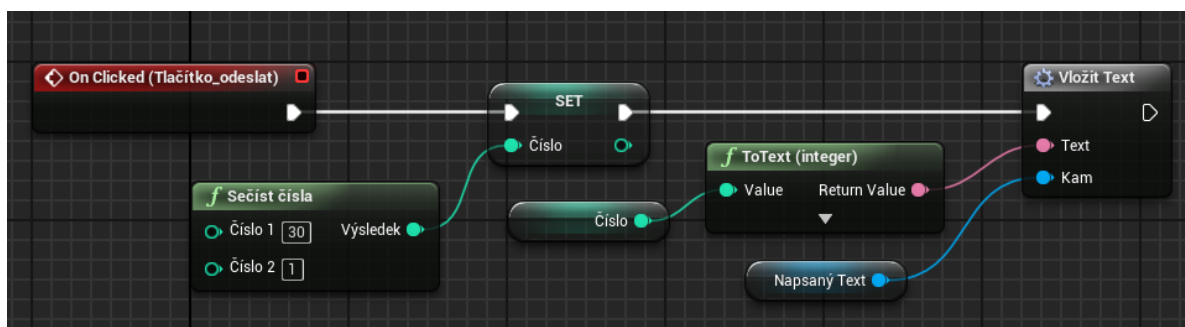
- 2x stiskni tlačítko „Tlačítko\_odeslat“. Jaká čísla se při jednotlivých stisknutí vypíší a proč ?

0 a „jiné číslo“;

protože číslo se změní až na konci scénáře

## SČÍTÁNÍ

- Obrat' svou pozornost na seznam „Functions“. Tento seznam obsahuje složku „Matematika“, ve které je umístěn blok „Sečíst čísla“. Připoj a nastav jej podle obrázku níže.



OBRÁZEK 95 ZAPOJENÍ BLOKU SEČÍST ČÍSLA

- Jaké číslo se vypíše po aktivování scénáře (napřed odpověz následně otestuj) ?

- Připoj do bloku „Sečíst čísla“ blok „Číslo“ tak, jak to vidíš na obrázku níže. Co se stane po opakovaném klikání na tlačítko „Tlačítko\_odeslat“ (Napřed odpověz, poté otestuj) ?



OBRÁZEK 96 ZAPOJENÍ PROMĚNNÉ JAKOŽTO ČÍTAČE

## POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Sčítání“ ukazuje způsob, jakým lze zapojit proměnnou jakožto čítače.

Někteří žáci mohou mít problém pochopit takovéto zapojení. Vyučující opět může pomoci tím, že s žáky popíše, co se ve kterém momentu scénáře stane. Většina žáků pouze po tomto popisu pochopí.

Možné odpovědi na otázky:

- Jaké číslo se vypíše po aktivování scénáře (napřed odpověz následně otestuj) ?

31

- Připoj do bloku „Sečíst čísla“ blok „Číslo“ tak, jak to vidíš na obrázku. Co se stane po klikání na tlačítko „Tlačítko\_odeslat“ (Napřed odpověz, poté otestuj) ?

1,2,3,4, ...; číslo vždy o jedno větší než předchozí

## POČÍTÁNÍ OTÁZEK

- Uprav tvůj blok, jenž zadává a vyhodnocuje otázky tak, aby po zodpovězení otázky a následné pochvle napsal, na kolikátou otázku jsi správně odpověděl/a. Využij k tomu blok „Číslo“
- Vhodněji přejmenuj blok „Číslo“. Snaž se, aby jeho název ukazoval to, co počítá
- Uprav scénář tak, aby se vypisovala celá věta říkající pořadí otázky. (Př: Počet zodpovězených otázek: [x]). Dopomoz si blokem „Spojit texty“

## POZNÁMKY PRO VYUČUJÍCÍHO

Podkapitola „Počítání otázek“ dává žákům za úkol, využít čítače v praxi.

Rychlost postupu žáků nyní závisí na tom, jak přehledně tvořili svůj scénář v lekci Kvíz V2. Díky tomu se žáci učí i tomu, aby své scénáře tvořili co možná nejpřehledněji.

## 5.9.2. Lekce 2. Finální nákup

Orientační časová dotace: 45 minut

Využitá úroveň: Nákup\_V2

V lekci žáci řeší složitější zapojení s využitím proměnných

### Jednotlivé aktivity

- Otevři složku „Nákup“, a v ní spust' soubor „Nákup\_V2“
- Zajisti, aby se pokaždé, co Honza sebere nějaký předmět, napsal do chatu celou větou počet předmětů, jež Honza právě nese
- Zajisti, aby byl tento počet správný i potom, co Honza předá předměty Klaudii
- Zajisti, aby po předání předmětů Klaudii chat celou větou napsal, kolik předmětů Klaudie nese. Otestuj i případ, ve kterém Honza donese Klaudii předměty dvakrát.

#### POZNÁMKY PRO VYUČUJÍCÍHO

*Žáci mohou mít problém zopakovat postup vytvoření proměnné. V takovém případě je možné žákům poradit.*

*Žáci se budou během plnění úkolů setkávat s komplikacemi různého druhu. Pokud se jedná o technické komplikace, vyučující může žákům pomoci s řešením (př. vytvoření události). Při problémech, které jsou spojeny se sestavováním scénáře, by měl vyučující pouze nastítnit situaci a cílenými otázkami žáky navést ke správnému řešení.*

## 6. Závěr

Popsal jsem prostředí Unreal Engine 4 a přitom jsem potvrdil, že se jedná o herní engine, který je jeden z nejpopulárnějších na trhu. Můžou za to mimo jiné možnosti, které toto prostředí poskytuje, ať už velké množství součástí, z nichž každá plní svůj jedinečný účel, či licence vhodná pro začínající i pokročilé vývojáře. Velkou výhodou UE4 je i možnost programovat skrze systém Blueprint, který podporuje vizuálního skriptování i psaní kódu pomocí jazyka C++. Použití tohoto systému jsem demonstroval na vzorovém projektu, který mimo jiné obsahuje použití událostí, či způsoby tvorby cyklů.

Porovnal jsem prostředí UE4 s prostředím Scratch a MakeCode z pohledu vhodnosti k výuce programování. Prostředí UE4 mělo oproti Scratch a MakeCode mnoho nevýhod. To je prioritně způsobeno tím, že UE4 nebyl prioritně zamýšlen k výuce programování. Přesto však díky systému Blueprint a některým jeho unikátním vlastnostem je možné tento systém při výuce programování využít.

Vytvořil jsem osnovu výuky, podle níž je možné vést výuku programování v prostředí UE4. Dále jsem vytvořil sadu úloh, která seznamuje žáky s prostředím UE4 a učí je základním prvkům programování, jako jsou například cykly, či události. Tato sada úloh je vytvořena tak, že v ní žáci dokáží postupovat samostatně, bez nutnosti výkladu vyučujícího. Vyučující je zde spíše pro podporu žáků než jakožto zprostředkovatel znalostí.

Dále jsem vytvořil podklady pro vyučujícího, zahrnující vzorový projekt obsahující vypracované scénáře, které žáci dostávají za úkol vypracovat, a metodické listy, obsahující zadání pro žáky spolu s poznámkami k úlohám.

Ověřováním při výuce jsem tyto materiály dále vylepšoval odstraňováním nedostatků. Zároveň jsem si ověřil, že podle mnou vytvořené metodiky lze vyučovat žáky programování.

Je otázkou, jestli se prostředí Unreal Engine 4 stane předním programem pro výuku programování. Je možné, že v budoucnu mohou základní školy mnohem více dbát na výuku programování a že se vytvoří mnoho odvětví této výuky, například formou dobrovolných aktivit, které budou mít za cíl vytvořit něco komplexního, například kompletní videohru. V takovém případě by má práce mohla být užitečná, jelikož je metodika v ní realizována v prostředí, které je světově uznávané pro vývoj her a mnohého dalšího.

Navázat na mou práci se může několika způsoby. Lze do mé práce přidat úlohy, zaměřené na čtení kódu, či jiné typy úloh, které v mé sadě nejsou obsaženy v dostatečném počtu. Lze navázat i vylepšením grafické stránky mé sady úloh, která by zahrnovala různé efekty, či jejich větší barevnost. Dále je možné podrobně otestovat mou práci při výuce programování a případně ji ze získaných poznatků dále rozvíjet. Je možné zkoumat, které prvky programování dělají žákům v UE4 problémy a na jaké prvky je tedy třeba se zaměřit. Dále je také možné zkoumat vhodnost metodiky pro výuku informatického myšlení a případně práci z nalezených nedostatků poupravit.



## Seznam použité literatury a zdrojů

- [1] LEE, J., *Learning Unreal Engine Game Development* [online]. [cit. 2020-03-05]. Packt Publishing Limited, 2016. Dostupné z WWW: [https://subscription.packtpub.com/book/game\\_development/9781784398156/1/ch011v11sec10/the-history-of-unreal-engine](https://subscription.packtpub.com/book/game_development/9781784398156/1/ch011v11sec10/the-history-of-unreal-engine)>.ISBN: 9781784398156
- [2] Bibliografická citace. *Unreal Engine 4 - games* [online]. [cit. 2020-04-12]. Dostupný z WWW: <<https://www.giantbomb.com/unreal-engine-4/3015-2551/games/>>
- [3] FARRIS, J., *Forging new paths for filmmakers on "The Mandalorian"* [online]. [cit. 2020-04-12]. Dostupný z WWW: <<https://www.unrealengine.com/en-US/blog/forging-new-paths-for-filmmakers-on-the-mandalorian>>
- [4] Bibliografická citace. *Co je informatické myšlení?* [online]. [cit. 2020-05-02]. Dostupný z WWW: <<https://www.imysleni.cz/informaticke-mysleni/co-je-informaticke-mysleni>>
- [5] QUIGLEY, E., *ADDIE: 5 Steps To Effective Training* [online]. [cit. 2020-05-02]. Dostupný z WWW: <<https://www.learnupon.com/blog/addie-5-steps/>>
- [6] Bibliografická citace. *Unreal Engine 4 Documentation* [online]. [cit. 2020-03-05]. Dostupné z WWW: <<https://docs.unrealengine.com/en-US/index.html>>
- [7] MCMANUS, S., *Jak se naučit programovat v 10 lekcích*. Svojtka&Co., 2017. ISBN: 978-80-256-2046-5
- [8] VANÍČEK, J., et al. *Programování ve Scratch pro 2. stupeň základní školy* [online]. Jihočeská univerzita v Českých Budějovicích. [cit 2020-03-05]. Dostupné z WWW: <<https://www.imysleni.cz/ucebnice/programovani-ve-scratchi-pro-2-stupen-zakladni-skoly>>
- [9] Bibliografická citace. *Microsoft MakeCode for BBC micro:bit, Documentation* [online]. [cit 2020-04-15]. Dostupné z WWW: <https://makecode.microbit.org/about>
- [10] Bibliografická citace. *MakeCode Languages: Blocks, Static TypeScript and Static Python*. [online]. [cit 2020-04-15]. Dostupné z WWW: <<https://makecode.com/language> >
- [11] PECINOVSKÝ, J., *Zásady správné výuky programování* [online]. [cit 2020-03-05]. Dostupné z: <<http://www.ceskaskola.cz/2001/09/rudolf-pecinovsky-zasady-spravnevyuky.html>>

- [12] ČÁPKA, D., *ITnetwork.cz* [online]. [cit 2020-03-05]. Dostupné z WWW:  
<<https://www.itnetwork.cz/it-e-learning>>

## Přílohy

1. Pracovní listy pro žáky
2. Projekt\_Programování: složka souborů, obsahující projekt UE4, ve které žáci plní úkoly ze svého zadání
3. Projekt\_Programování\_Vzor: složka souborů, obsahující projekt UE4, ve které jsou vytvořeny vzorové scénáře jednotlivých lekcí

Tyto přílohy jsou z důvodu jejich velikosti součástí pouze elektronické verze práce na paměťové kartě, kde je též možné nalézt bakalářskou práci ve formátu PDF.