

# Zadání zápočtového projektu

## *Gremlins Hunter*

### 1 Story

Naším cílem je vytvořit si menší, jednoduchou a snad zábavnou hru na motivy filmu Gremlins (1984). Pokud jste tento film neviděli, je to sice možná škoda, ale i tak se bez toho obejdete.

Žil byl jednou jeden člověk (Vy) a ten jednoho dne v jednom obchodě ukradl krásné, roztomiloučké a velice hodné a poslušné zvířátko rodu Mogwai a pojmenoval ho Gizmo. Gizmo však není obyčejné zvířátko, není z tohoto světa, ale vy jste si k němu ukradli také uživatelský manuál, takže víte, na co si máte dávat pozor, a tedy Vás nic nezaskočí. Pokud Gizmo přijde do styku s vodou, rozmnoží se a přivede na svět dalších několik tvorů, které si pro jednoduchost nazveme gremlini. Pokud se Gizmo v noci nají, sám se změní v gremlina. Gremlini jsou zvířátka, která jsou oproti Gizmovi velice zlobivá a to pro to, že rády ubližují, a když ublíží moc, tak našeho hlavního hrdinu zabijí.

Ale jak už to tak bývá, tak veškerá snaha být opatrný a dodržovat vše, jak by se mělo, je k ničemu, když přijde den blbec. Jednoho deštivého dne, když jste se v zimě již za tmy vraceli ze školy z předmětu Objektové programování zpět na kolej za svým zbožňovaným Gizmem, zasáhla kolej menší sprška meteoritů, které sice nikomu neublížili, ale nadělali dírký do stropů, kterými déšť protéká až do prvních pater. Dírký jsou tak malé, že jimi nikdo nepropadne, ale voda jimi stéká jedna báseň. Současně jeden z meteoritů shodil a rozbil klíčku, ve které byl Gizmo uzavřen (taková zlí jste, zavírat tohle roztomilé zvířátko do klece). Gizmo se ze strachu z meteoritů schoval do koutu místnosti, přičemž na něj během cesty kápala kapička vody, která způsobila zrod prvního gremlina. Nový přírůstek jakožto první akt své existence vypíchl Gizmovi očička a při vši své nešikovnosti je vypíchl také sobě (toto mělo, naštěstí pro Vás, také pozitivní výsledek a to ten, že každý další potenciální potomek si vždy po zrodu sám vypíchne oči a tedy Vás neuvidí). Mimo to se na koleji různě válí všelijaké jídlo a jelikož je již tma, Gizmo se nesmí najíst!

Vaším úkolem je zachránit Gizma a odstranit všechny gremliny. Pokud se Vám nepodaří zachránit Gizma, stejně budete muset odstranit všechny gremliny, jinak odstraní oni Vás.

Pro zjednodušení si představíme koleje ve 2D prostoru jakožto jednolitou plochu, na které se celá akce bude odehrávat.

### 2 Souhrn

Naše hrací pole je rozměru 14x19 (14 řádků, 19 sloupců, parametr může být před spuštěním změněn). Sloupce hracího pole jsou identifikovány znaky anglické abecedy A-Z, řádky jsou identifikovány čísly 1-15. Máme 3 druhy mimozemšťanů (Gizmo, Gremlin a TurboGremlin) a 2 druhy pozemšťanů (Human, Hunter). Každý z těchto živých tvorů má přiřazený svůj počet životů, který se nastaví při jejich vytvoření a sníží se o 1 každým zraněním. Každý takovýto tvor má dále svůj stav a to v pořádku, zraněn a zabit.

Na hracím poli se může na jednotlivých polích nacházet voda, potrava, nebo živý tvor a to v libovolné kombinaci (tedy i více najednou, počet není omezen). Každé políčko hracího pole má 30% šanci, že bude tvořeno vodou (číslo se může dle požadavku změnit). Na hracím poli se dále bude nacházet 1x potrava (umístěna náhodně tak, aby každá pozice měla stejnou pravděpodobnost umístění). Na hracím poli budou dále umístění ještě 2 lidé (Human), kteří budou spát a tedy se nebudou dále hýbat. Levý horní roh a pravý dolní roh pole bude suchý a bez potravy. Levý horní roh bude počáteční pozice Gizma a jednoho gremlina. Pravý dolní roh bude Vaše počáteční pozice (pozice umístění Huntera). Všechna ostatní pole budou prázdná a nic se na nich vyskytovat nebude.

### 3 Struktura projektu

Vytvořte v programovacím jazyku Java v rozhraní Eclipse projekt `GremlinsHunter_uzivatelskejmeno`. Třidu nesoucí funkci `main` pojmenujte `Hunt`.

V projektu vytvořte balík (pagkage) pod názvem `aliens`. Balík `aliens` bude obsahovat 3 třídy (`Gizmo`, `Gremlin`, `TurboGremlin`).

V projektu vytvořte balík (pagkage) pod názvem `humans`. Balík `humans` bude obsahovat 2 třídy (`Human`, `Hunter`).

V defaultním balíku vytvořte třídu `Gremlins`, `Humans` a `BattleField`.

V projektu vytvořte 2 rozhraní (`Informujici`, `Akcní`).

#### 3.1 Třída Hunt

Třída `Hunt` obsahuje mimo metody `main` také metodu `vykresli`. Metoda `vykresli` má jeden vstupní parametr typu třídy `BattleField`. Metoda překreslí na konzoli aktuální stav na hracím poli včetně identifikátorů sloupců a řádků. Následující tabulka definuje, jaký znak se vykreslí, je-li na dané pozici v hracím poli daný objekt/předmět. Pokud se na dané pozici nenachází nic, zobrazí se mezera (prázdný znak). Je-li na daném poli více objektů, vykreslí se znak toho s vyšší vahou dle následující tabulky.

váha	objekt/předmět	znak	Popis znaku
1	voda	~	vlnovka
2	potrava	!	vykřičník
3	Gremlin	X	X
4	TurboGremlin	Y	Y
5	Human	M	M
6	Gizmo	O	O
7	Hunter	H	H

#### 3.2 Balík aliens

Balík `aliens` obsahuje definici tříd reprezentující jednotlivé mimozemské příšerky, se kterými se lze na poli setkat. Třídy jsou blíže popsány v **příloha1**. Ve třídě `Gizmo` musí být implementováno rozhraní `Informujici` a `Akcní`, ve všech jeho potomcích metody rozhraní překryjte, bude-li to třeba.

#### 3.3 Balík humans

Balík `humans` obsahuje definici tříd reprezentující jednotlivé lidské bytosti, se kterými se lze na poli setkat. Třídy jsou blíže popsány v **příloha1**. Ve třídě `Human` musí být implementováno rozhraní `Informujici` a `Akcní`, ve všech jeho potomcích metody rozhraní překryjte, bude-li to třeba.

#### 3.4 Třída Gremlins

Třída `Gremlins` obsahuje atribut `creatures`. Atribut `creatures` je pole obsahující všechny mimozemské tvory, které se nachází na hracím poli. Rozměr tohoto pole je tedy shodný s počtem těchto stvůr. Třída implementuje rozhraní `Informujici` a `Akcní`.

### 3.5 Třída Humans

Třída Humans obsahuje atribut `mankinds`. Atribut `mankinds` je pole obsahující všechny pozemské lidské tvory, které se nachází na hracím poli. Rozměr tohoto pole je tedy shodný s počtem těchto pozemšťanů. Třída implementuje rozhraní `Informující` a `Akční`.

### 3.6 Rozhraní Informující

Rozhraní `Informující` obsahuje metodu `vypisStav`, která vypíše na konzoli stav dané bytosti (v případě objektů v balíku `aliens` a `humans`), nebo souhrn stavu celé rasy (v případě třídy `Humans` a `Gremlins`).

Rozhraní dále implementuje metodu `pocetZivotu` která vrátí počet životů bytosti (v případě objektů v balíku `aliens` a `humans`), nebo součet životů všech bytostí celé rasy (v případě třídy `Humans` a `Gremlins`).

### 3.7 Rozhraní Akční

Rozhraní `Akční` implementuje metodu `pohyb`, která zajistí přesun objektu na jinou pozici.

Rozhraní `Akční` implementuje metodu `utok`, která má vstupní parametry souřadnic, kam je útok směřován. Metoda `utok` provede útok na objekt, z něhož byla vyvolána (je to tedy metoda typu „útočí na mě“) a vrátí počet životů, který daný objekt při útoku ztratil. Dojde-li k zásahu, objektu se odečte 1 život a zajistí se nastavení správného stavu objektu (v pořádku, poškozen, mrtev). Leží-li na daném poli (na souřadnicích útoku) více objektů (tato možnost není proti našim pravidlům), týká se útok každého takového objektu (objekty se tedy navzájem nekryjí).

## 4 Poznámky

### 4.1 Typografie

Úhlové závorky `>` uvedené v příkladech textového výstupu na konzoli nejsou součástí výstupu, pouze představují začátek řádky konzole.

### 4.2 Třídy

Každá třída může obsahovat libovolné množství dalších (pomocných) metod, atributů a tříd.

### 4.3 Zadané hodnoty parametrů

Počítejte s tím, že zadané hodnoty parametrů mohou být na požádání změněny. Jedná se například o rozměr hracího pole, počet jednotek, % pravděpodobnosti apod. Operujte s těmito hodnotami tak, aby v případě požadavku na jejich změnu bylo tuto změnu třeba provést pouze na jednom místě (nemusí být všechny pohromadě, ale aby každý parametr bylo třeba změnit jen na jednom konkrétním místě v kódu).

### 4.4 Atributy

#### 4.4.1 Poloha Objektů

Atributy `x` a `y` ve třídě `Gizmo`, `Human`, atd. odpovídají souřadnicím a určují polohu bytosti. Poloha bytosti je omezena na kladný kvadrant se zajištěním toho, aby se jednotka vždy nacházela na hracím poli (omezení v **příloha1**).

#### 4.4.1.1 Vytváření jednotek (konstruktory)

Je-li zadán konstruktor s parametry mimo uvedené meze, zajistěte nastavení atributů na nejbližší okraj této meze.

## 5 Výstupy

Pro lepší pochopení uvedeme příklad konkrétního výstupu. Mějme hrací pole 7x7, v poli máme 1x Gizmo, 1x Hunter, 2x Human, 4x Gremlin, 1xTurbogremlin, 1x potrava a 6x voda. Jsou-li v poli ještě nějaké jiné jednotky, jsou překryty dle pravidel. Výstup na konzoli metodou `vykresli` je pak následující.

```
> ABCDEFG
>01OX M ~
>02 ~~
>03 * XX
>04 ~~X
>05 M
>06 ! Y
>07 ~ H
```

## 6 Metoda Main a další specifikace

V metodě `Main` zajistěte vytvoření instancí tříd `Gremlins`, `Humans` a `BattleField` a jejich naplnění dle úvodního zadání.

Po naplnění zobrazte na konzoli plánec celého hracího pole (metoda `vykresli`).

Metoda `Main` dále zajistí cyklické střídání akcí rasy `Gremlins` a `Humans`. Každá jedna akce každé rasy je vyvolána její metodou `pohyb`, která vyvolá metodu `pohyb` pro každého člena dané rasy. Nejprve se tedy pohnou všechny bytosti jedné rasy a poté všechny bytosti druhé rasy. V rase `Gremlins` se všechny bytosti pohybují náhodně o jedno pole. V rase `Humans` se pohybuje pouze `Hunter` a to podle uživatelského vstupu na konzoli (0 = stát, 2 = dolů, 4 = vlevo, 8 = nahoru, 6 = vpravo). Všechny bytosti se umí pohybovat jen o jednu pozici. Jakmile by se nějaká bytost měla pohnout mimo přípustné hrací pole, bude automaticky posunuta po obvodu pole s 50% pravděpodobností po směru a 50% proti směru hodinových ručiček o jedno políčko.

Rasa `Humans` má první tah.

Po každém tahu obou ras se na konzoli vykreslí stav na poli a vypíše se souhrnný stav jednotlivých ras.

Každý pohyb každé bytosti (mimo `Gizmo`) je současně útokem, pokud na dané pozici je bytost jiné rasy. Každý tvor rasy `Gremlins` po úspěšném útoku na druhou rasu sám ztratí 1 život. `Hunter` má 6 životů. Pokud tvor rasy `Gremlins` ukončí svůj pohyb na vodě, vytvoří na daném místě dalšího tvora třídy `Gremlin` (tento se bude moci hýbat až následující kolo). Pokud `Gizmo` ukončí svůj pohyb na poli s potravou, zahyne a na jeho místě vznikne `TurboGremlin`.

Hra končí a vyhráváte, jakmile `Hunter` pozabíjí všechny `Gremlins`. Prohráváte, jakmile zemře `Hunter`. Za každého zabitého tvora rasy `Gremlins` získáváte 1 bod, pokud `Gizmo` zahyne, ztrácíte 25 bodů. Výsledný počet bodů se zobrazí pouze, pokud `Hunter` přežije. Z toho všeho tedy vyplývá, že `Gizmo` zahyne při umístění na pole s potravou a je zachráněn, pokud se nachází na poli s `Hunterem` (v takovém případě se již na poli nezobrazuje a nevykonává žádnou akci).

## 7 Odevzdání

Celý projektový adresář zabalte do archivu (**zip**) a odešlete na e-mailovou adresu [cernyf@vscht.cz](mailto:cernyf@vscht.cz) a jako kopii na [oop.java@seznam.cz](mailto:oop.java@seznam.cz). Jako předmět e-mailu uveďte **OOP\_JAVA\_Projekt**. E-mail bez podpisu budu bez jakéhokoliv upozornění mazat a budu ho považovat za nedoručený.

Projekt je nutno odevzdat **nejpozději do čtvrtka 22. 12. 2011 do 16:00**. V případě odevzdání do soboty 10. 12. 2010 do 12:00 je možno po domluvě přijít na předtermín 16. 12. asi od 15:00.

Ač se mají projekty odevzdávat vždy kompletní, upozorňuji, že je lepší odevzdat alespoň nějaký projekt, než žádný. Pokud byste se rozhodli dělat projekt na poslední chvíli a poslat ho nekompletní, uveďte projekt alespoň do spustitelného stavu (pokud nějaké funkce nebudou pracovat správně, zakomentujte je). Dále pak mějte na paměti, že se jedná o samostatný projekt, ne skupinový. V případě duplicitních projektů nebude podstatné, kdo ho vytvořil a kdo je ten, komu ho dotyčný dal, takže se sdílením své práce buďte skutečně opatrní (předtermíni dostanou známku zapsanou do indexu teprve, až budu mít v ruce všechny projekty). Pokud byste chtěli stáhnout a použít nějaký již vypracovaný projekt z internetu, zklamu Vás, je zbytečné to hledat.