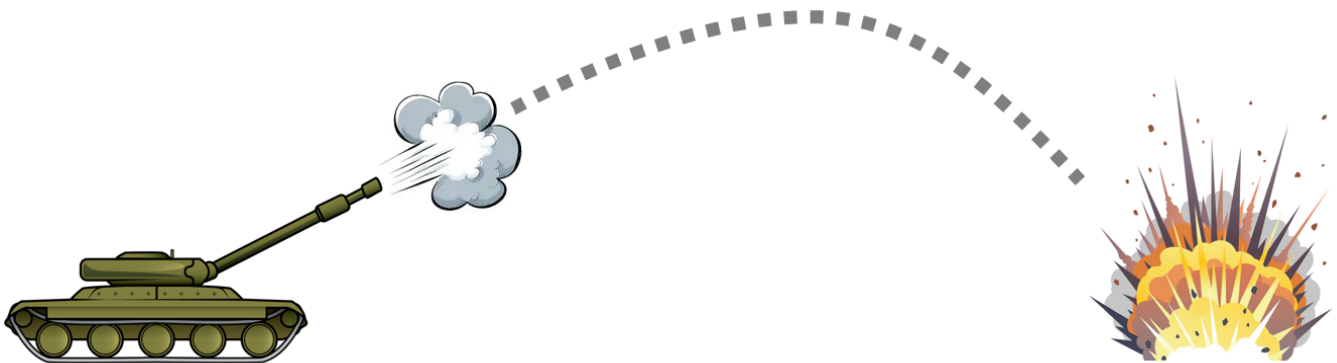


Zadanie 1

Nezabudnite zadanie nahrať do AIS do príslušného miesta odovzdania (odovzdajte len zdrojový .c súbor).

Dôležité informácie

- **Termín odovzdania:** 3. týždeň (3. – 5. marec 2020) osobne na vašom cvičení
- **Počet bodov:** 8
- **Prezentácia Z1**
- **English version**



Krátky popis

Cieľom zadania je naprogramovať v jazyku C jednoduchú interaktívnu konzolovú hru simulujúcu súboj 2 tankov. Jeden z tankov je ovládaný hráčom z klávesnice a súper je riadený umelou inteligenciou (PC protihráč). Hráč zvolí uhol výstrelu a počiatočnú rýchlosť (silu), dôjde k výstrelu a vypočíta sa dostrel (t.j. X-ová súradnica, kde projektil dopadne). Ak projektil zasiahne PC protihráča, hra končí. V opačnom prípade je na ťahu PC protihráč, ktorý náhodne generuje uhol a silu výstrelu (snažte sa predovšetkým o vytvorenie umelej inteligencie). A týmto spôsobom sa strieda hráč a PC protihráč až pokiaľ nedôjde k zničeniu niektorého z tankov.

Na začiatku hry sa tanky náhodne umiestnia na hernej plošine (vygeneruje sa ich X-ová súradnica). Herná plošina má dĺžku určenú tak, aby sa jej vykreslenie zmestilo do konzolového okna (ideálne určená globálnou konštantou pomocou `#define`, napr. `#define ARENA_LENGTH 80`). Tanky je potrebné vhodne vizualizovať, napr. pomocou vhodne zvolených symbolov `1` - hráč a `2` - PC protihráč. Rovnako je vyžadovaná vizualizácia dopadu projektilu, t.j. nakreslenie symbolu v mieste dopadu/výbuchu.

V zadaní sa bude kontrolovať jednak požadovaná funkcionálnosť, ale aj kvalita návrhu, t.j. prehľadnosť hlavnej funkcie `main`, rozčlenenie zdrojového kódu do funkcií (napr. oddelenie hernej logiky od kreslenia konzolovej grafiky),



Ukazka (video)

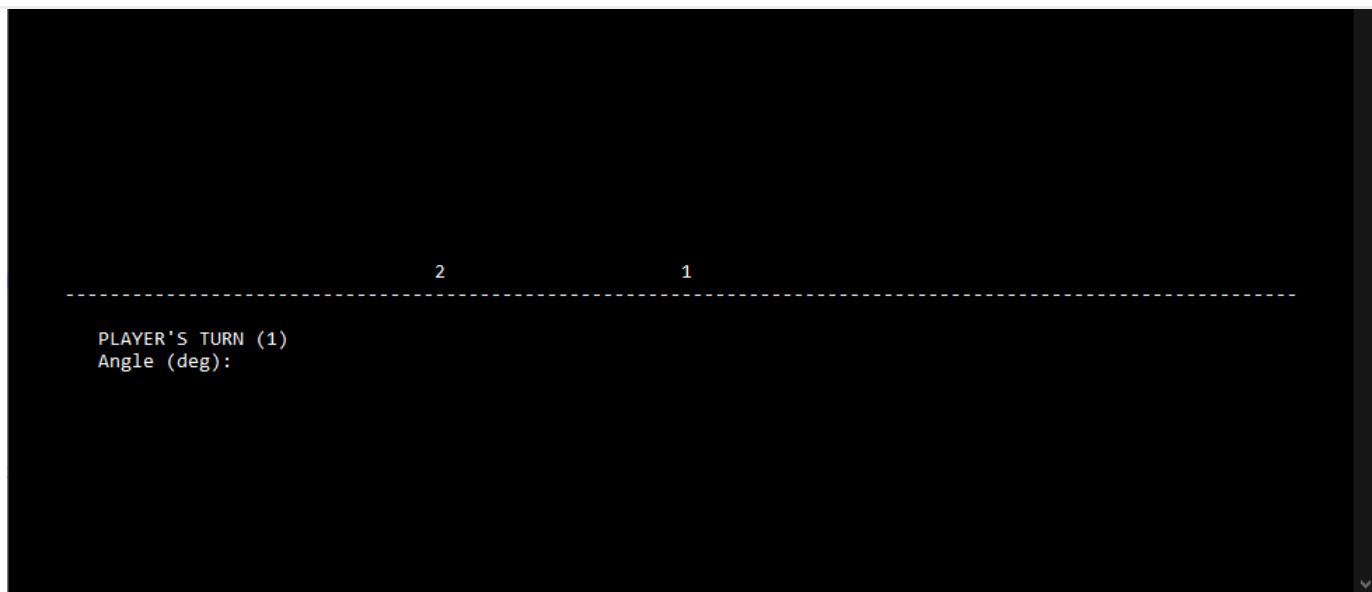
0:00 / 0:29



Hlavné aspekty hry

Inicializácia

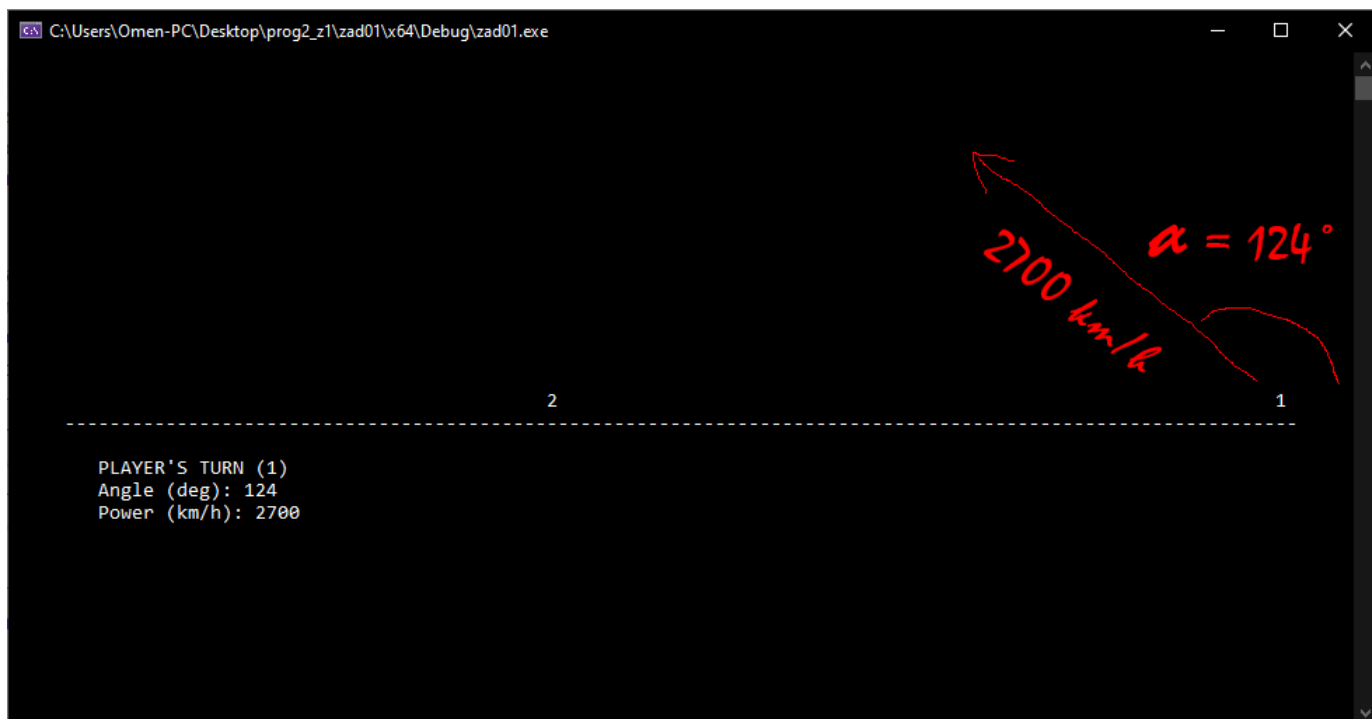
Hra začína rozmiestnením tankov na hernej plošine. Rozmiestnenie je náhodné. Na tento účel je potrebné si napísať vlastnú funkciu na generovanie náhodného čísla vo zvolenom intervale (napr. šírka intervalu bude zodpovedať šírke konzolového okna). Funkcia na generovanie náhodného čísla sa dá použiť aj v neskorších fázach zadania, napr. pri generovaní parametrov strely PC protihráčom. Treba si dávať pozor, aby nevznikla situácia, že vygenerovaná X-ová súradnica tanku bude presahovať rozmer hernej plošiny a taktiež treba zabrániť tomu, aby neboli obidva tanky vygenerované na rovnakom mieste (t.j. rovnaká X-ová súradnica). Pri každom spustení programu sa musia tanky vygenerovať na inej náhodnej pozícii (treba použiť funkciu `srand`, zavolať ju na začiatku programu vo funkcii `main`).



Obr. Náhodné umiestnenie tankov na začiatku hry (1 je hráč, 2 je PC protivráč). Tank ovládaný hráčom sa môže vyskytnúť vľavo/vpravo od PC protivráča.

Interakcia s hráčom

Tanky sa budú v streľbe striedať. Ako prvý začne hráč. Manuálne z klávesnice zadá parametre streľby: uhol a počiatočnú rýchlosť (silu) výstrelu. Zvoľte si jednotky, ktoré sa budú používať pri zadávaní vstupu (oznáňte to hráčovi výpisom na obrazovku). Netreba ošetrovať neplatné vstupy. Na zadávanie vstupov z klávesnice použite funkciu scanf.

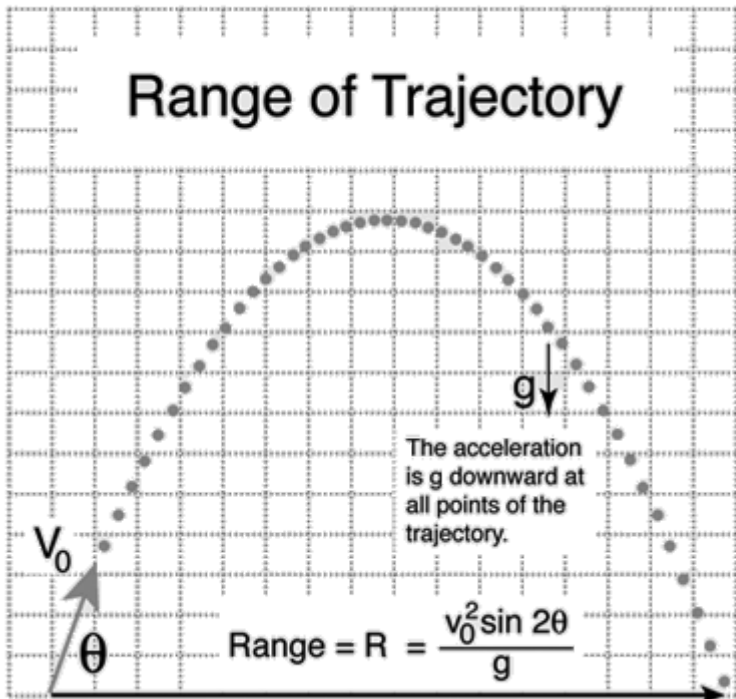


Obr. Zadávanie parametrov strely z klávesnice (platí pre hráčov tank).

Výpočet dostrelu

g

- v_0 je počiatočná rýchlosť výstrelu (alebo aj sila) udávaná v m/s (pozor treba previesť rýchlosť ak ste ju zadávali v km/h)
- g je tiažové zrýchlenie (počítame s hodnotou $9,81 m/s^2$)
- α je uhol výstrelu vyjadrený v stupňoch (pozor matematická funkcia `sin` z hlavičkového súboru `math.h` očakáva uhol vyjadrený v radiánoch)



The basic motion equation

$$x = v_{0x} t$$

can be used to find the range. By symmetry, the total time of flight is equal to twice the time at the peak:

$$t_{\text{range}} = 2t_{\text{peak}} = \frac{2v_{0y}}{g}$$

This gives:

$$R = \frac{2v_{0x} v_{0y}}{g}$$

$$R = \frac{2v_0^2 \sin \theta \cos \theta}{g}$$

$$R = \frac{v_0^2 \sin 2\theta}{g} \quad \text{Calculation}$$

using the trig identity:
 $\sin 2\theta = 2\sin \theta \cos \theta$.

Obr. Dráha projektilu (prevzaté z hyperphysics.phy-astr.gsu.edu)

Výpočet dostrelu musí byť podľa vyššie uvedeného vzťahu implementovaný v samostatnej funkcii. Nezabudnite si napísať aj vlastné funkcie na prevod jednotiek (km/h na m/s, stupne na radiány).

Umelá inteligencia PC protihráča

PC protihráč by mal vykazovať istú úroveň inteligencie, t.j. jeho tank by mal mieriť kanónom na hráčov tank (uhol výstrelu) a pri výstrele použiť adekvátnu silu (aby nebola napríklad prestrelená šírka hracej plochy). Inými slovami, PC protihráč sa musí snažiť trafiť váš tank. Body za túto časť zadania budú odzrkadľovať kvalitu vášho návrhu.

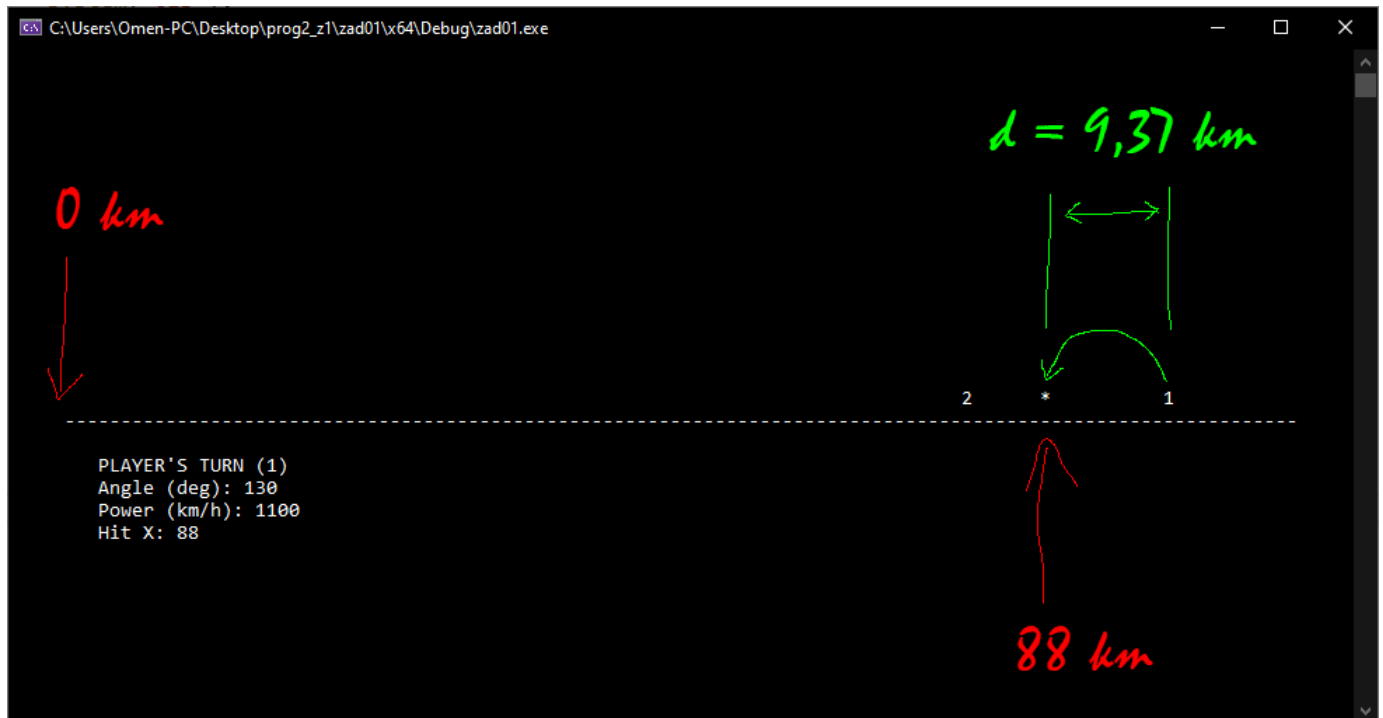
Kreslenie hry

Dôležitou časťou tohto zadania je kreslenie hernej grafiky do konzolového okna. Dbajte na prehľadnosť a vizuálnu prívetivosť vašej grafiky. Kreslenie konzolovej grafiky musí byť oddelené od zvyšku hernej logiky, t.j. musíte na to vytvoriť samostatnú funkciu (napr. `printGame`). Túto kresliacu funkciu budete v priebehu hry opakovane volať na prekresľovanie aktuálneho stavu hry.

Minimálne požiadavky na kreslenie hernej grafiky:



Zobrazenie dopadu projektilu (po vystrelení sa v konzole zobrazí bod dopadu, výška a v správnej vzdialenosti prepočítanej na rozmery vašej hernej grafiky)



Obr. Vizualizácia dopadu projektilu v konzole (s pomocnou infografikou). Červené šípky znázorňujú X-ové pozície počiatku súradnicovej sústavy a bodu dopadu. Zelenou farbou je vyznačený dostrel.

Vymazanie obsahu konzoly

Multiplatformová funkcia `clearScreen` na vymazanie obsahu konzoly. Môžete ju použiť na aktualizáciu hernej grafiky.

```
1 void clearScreen() {
2     #if defined(WIN32) || defined(_WIN32) || defined(__WIN32__) || defined(__NT__)
3         system("cls");
4     #elif defined(__APPLE__)
5         system("clear");
6     #elif defined(__linux__)
7         system("clear");
8     #else
9     #error "Unknown OS"
10 #endif
11 }
```

Ukončenie hry

Hra končí v momente, keď vystrelený projektil zasiahne súperov tank. Na to, aby ste tank zničili nemusíte trafiť presnú súradnicu tanku - môžete si určiť určitý interval/zónu, do ktorej keď projektil dopadne, tak tank bude zničený, napr. deštrukčná zóna v okruhu +/- 1 km od miesta dopadu projektilu.



```
1 #define _CRT_SECURE_NO_WARNINGS
2 #define _USE_MATH_DEFINES
3 #define ARENA_LENGTH 80 // sirka hry
4 #define G 9.81 // tiažove zrýchlenie
5 #include <stdio.h>
6 #include <stdlib.h>
7 #include <time.h>
8 #include <math.h>
9
10 // Funkcia na generovanie nahodneho cisla v intervale <a,b>
11 // pozn. v pripade potreby mozete upravit na typ 'double'
12 int randomNumber(int a, int b) {
13     // todo
14 }
15
16 // Funkcia na kreslenie hernej grafiky
17 // vstupy:
18 //     * t1 - pozicia tanku 1 (hrac)
19 //     * t2 - pozicia tanku 2 (PC)
20 //     * hit - pozicia dopadu projektilu
21 //     * arena_length - sirka hracej plochy
22 // pozn. volba parametrov funkcie je na vas
23 void printGame(int t1, int t2, int hit, int arena_length) {
24     // todo
25 }
26
27 // Funkcia na prevod km/h na m/s
28 double kmh2ms(double kmh) {
29     // todo
30 }
31
32 // Funkcia na prevod stupnov na radiany
33 double deg2rad(double deg) {
34     // todo
35 }
36
37 // Funkcia na vypocet dostrelu (v metroch)
38 // vstupy:
39 //     * alpha - uhol vystrelu (radiany)
40 //     * v0 - pociatocna rychlost vystrelu (m/s)
41 double shotDistance(double alpha, double v0) {
42     // todo
43 }
44
45 // Funkcia na realizáciu tahu hraca
46 // pozn. volba parametrov funkcie/urcenie navratoveho typu je na vas
47 double humanTurn() {
48     // todo
49     // vo funkcii hrac zada parametre strely (uhol a rychlost)
```



```
53// Funkcia na realizáciu tanku PC protihráča
54// pozn. voľba parametrov funkcie/určenie návratového typu je na vás
55double pcTurn() {
56    // todo
57    // vo funkcii sa vygenerujú parametre strely (uhol a rýchlosť)
58    // funkcia vráti X-ovu súradnicu dostrelu (vzhľadom na použitý súradnicový systém
59}
60
61// Hlavná herná funkcia
62// vstupy:
63//     * t1 - pozícia tanku 1 (hráč)
64//     * t2 - pozícia tanku 2 (PC)
65// výstup:
66//     funkcia návratovou hodnotou signalizuje, ktorý hráč vyhral
67int game(int t1, int t2) {
68    printGame(...); // prvé vykreslenie hry bez dopadu projektilu
69    while (...) { // hra prebieha pokiaľ sú obidva tanky nezničené
70        humanTurn(...); // ťah hráča
71        printGame(...); // kresba aktuálneho stavu hry
72        pcTurn(...); // ťah PC
73        printGame(...); // kresba aktuálneho stavu hry
74    }
75    return ...; // vráti sa ID hráča, ktorý vyhral
76}
77
78int main()
79{
80    int t1, t2; // pozície tankov
81    srand((unsigned int)time(NULL));
82    t1 = randomNumber(...); // náhodná pozícia tanku 1
83    t2 = randomNumber(...); // náhodná pozícia tanku 2
84    int winner = game(...); // start hry
85    // vypísanie kto vyhral
86    return 0;
87}
```

Vašou úlohou je rozdeliť si kód celého zadania do vhodných logických celkov, t.j. vlastných funkcií. Kvalita architektúry vašej implementácie (návrh a používanie vlastných funkcií) bude zohľadnená počas bodovania.

Bodovanie

Body sú rozdelené nasledovne:

- **[1b]** Jednoduchosť/prehľadnosť funkcie `main`. Funkcia `main` by mala byť krátka a úsporná (minimum) kódu. V `main` funkcii postačí vytvoriť premenné reprezentujúce X-súradnice tankov, inicializovať ich na náhodné platné hodnoty (zvolaním funkcie na generovanie náhodného čísla vo zvolenom intervale) a zavolať hlavnú hernú funkciu (čítaj ďalej).
- **[2b]** Hlavná herná funkcia (s názvom napr. `game`, `runGame`, atď.). Táto funkcia bude zavolaná z funkcie `main`. Hlavná herná funkcia bude riadiť priebeh celej hry a volať ďalšie pomocné funkcie. Bude obsahovať herný cyklus, v



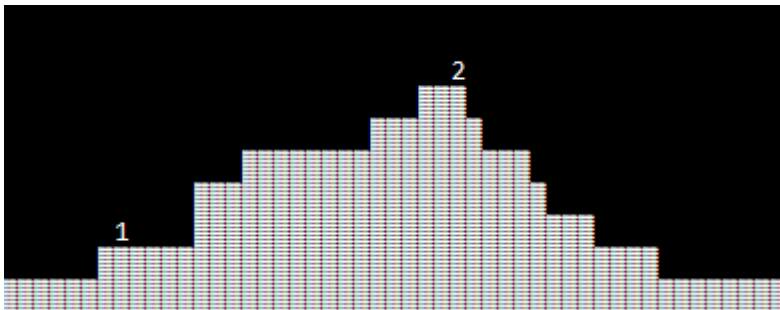
dostrel projektilu a vráti ho.

- **[1b]** Funkcia na vygenerovanie náhodného čísla vo zvolenom intervale. Túto funkciu využijete pri generovaní počítačových súradníc tankov na začiatku hry a taktiež pri generovaní parametrov strely PC protihráča.
- **[2b]** Funkcia na kreslenie hernej grafiky do konzoly (napr. s názvom `printGame`). Táto funkcia bude zodpovedná za formátované a vizuálne prívetivé vykreslenie aktuálneho herného stavu (t.j. pozície tankov, terén a miesto dopadu projektilu).

Extra body

Za extra snahu a námahu môžu byť udelené extra body po zvážení cvičiacim. Extra funkcionálnosť môže byť napr.:

- nerovnosť terénu, na ktorom sú umiestnené tanky



- pohyb tankov (napr. riadený z klávesnice)
- pokročilá vizualizácia v konzole (výbuchy, realistická dráha projektilu, naklonenie kanónu tanku)
- ostatné:
 - zdravie tankov
 - damage rôznych typov projektilov
 - rádius poškodenia tanku
 - rôzne atribúty tankov ako napr. štít, dočasné zosilnenie zbrane, možnosť strieľať viackrát za sebou
 - rôzne predmety predmety rozmiestnené na hernej ploche so špecifickým významom pre hru

Zdroje

- [Projectile Motion Calculator](#)
- [Projectile Motion Wikipedia](#)

Veľa úspechov pri programovaní zadania.

Built with [Grav CMS](#)

[Open Publishing Space](#) package by [hibbittsdesign.org](#)

© Programovanie 2. Pavol Marák. Ústav informatiky a matematiky FEI STU. 2020.