**ŽILINSKÁ UNIVERZITA V ŽILINE**

**FAKULTA RIADENIA A INFORMATIKY**

**INFORMATIKA 1**

Semestrálna práca

Samuel Vašečka

**2018**

Moja semestrálna práca nesie názov Battle ships, alebo po slovensky lodičky. Už z názvu sa dá poznať, že ide o známu počítačovú hru, ktorej základom je vyhrať nad protivníkom, a to tak, že mu pomocou strieľania zničíte všetky jeho lode, ktoré má rozostavené na hracej ploche, zatiaľ čo on sa snaží o to isté.

Moja semestrálna práca obsahuje 14 tried, pričom hlavná trieda *Aplikácia* pracuje so všetkými ostatnými triedami. (*Polia, Kontrola, Strielanie, Zapisovanie, Hraci, PosledneId, Tutorial, TopSkore, Top3Skore, Tabulka, RankHraca, Rank a Uvod* ) Hráči tu zadávajú vstupy, v podobe znakov a tak komunikujú s hrou, respektíve hre zadávajú pokyny. Celá hra teda prebieha v termináli, kde hráči zadávajú vstupy, ale zároveň aj dostávajú výstupy, ktoré ich navigujú ako pokračovať ďalej a nakoniec samozrejme aj ukážu výsledok hry.

Trieda **Polia** obsahuje 4 atribúty:

1. **VELKOST\_POLA** – typu int, pričom je static final a udáva veľkosť poľa v ktorom sú zapísané výsledky a zároveň aj veľkosť poľa, ktoré hráč vidí. Tento atribút má hodnotu 16, napriek tomu, že hracie pole (to, ktoré vidno) je iba veľkosti 10x10, a to z dôvodu, že keď sa kontroluje stav zasiahnutého políčka, kontrola zasahuje až 3 políčka dookola.
2. **CISLO** – typu int, pričom je static final a udáva hodnotu, ktorá sa pripočítava pri zadávaní nejakých súradníc hráčom z dôvodu, že pole je 16x16 ale tá časť, ktorú hráč vidí, začína až na indexe 3, 3. Tento atribút má hodnotu   
   (**VELKOST\_POLA** – 10)/2, čo sa rovná 3.
3. **pole** – typu int[][]. Toto pole je pole v ktorom sa zapisujú výsledky streľby a zadávania lodí.
4. **poleHracie** – typu String[][]. Toto pole je pole, ktoré hráč vidí a ukazuje mu aktuálny stav hry.

Trieda **Polia** obsahujenasledovné metódy:

1. **Konštruktor** priradí atribútu **pole** hodnoty, ktoré určia jeho veľkosť. Tieto hodnoty sú dve, a majú rovnakú veľkosť. (**VELKOST\_POLA)  
   Konštruktor** tiež priradí rovnaké hodnoty aj atribútu **poleHracie**.
2. Metóda **getCislo,** s návratovou hodnotou typu int,vráti hodnotu atribútu **CISLO**.
3. Metóta **plocha**, s návratovou hodnotou typu void, vykreslí do terminálu dvojrozmerné pole **pole.** (Pričom ho nevykreslí do jedného riadku, ale vytvorí pekný štvorec.
4. Metóda **kontrolaPlocha**, s návratovou hodnotou typu void, počíta koľkokrát sa v dvojrozmernom poli **pole** nachádzajú hodnoty, ktoré sa rovnajú -1 (predstavujú zničené časti lode). Táto metóda vracia dve hodnoty: 0 – ak ešte nie sú zničené všetky lode a 1 – ak sú zničené všetky lode.
5. Metóda **kontrolaPlochaSucet**, s návratovou hodnotou typu void, počíta koľkokrát sa nachádzajú v dvojrozmernom poli **pole**, hodnoty väčšie ako 1. (predstavujú rôzne typy lodí).
6. Metóda **getPlocha**, s návratovou hodnotou typu int, a parametrami int y a int x, vráti hodnotu, ktorá sa nachádza v dvojrozmernom poli **pole** na konkrétnom mieste y = riadok a x = stĺpec.
7. Metóda **setPlocha**, s návratovou hodnotou typu void, a parametrami int y, int x, int z, zmení v dvojrozmernom poli **pole** hodnpotu na konkrétnom mieste y = riadok a x = stĺpec za hodnotu z.
8. Metóda **plochaHracia**, s návratovou hodnotou typu void, a parametrom int zadavanie, vykreslí do terminálu dvojrozmerné pole **poleHracie**, pričom parameter zadavanie určuje, čo bude obsahovať pole na svojich pozíciách.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **Kontrola** obsahuje 5 atribútov:

1. **pole** – typu **Polia**, ktorý napĺňame objektami triedy **Polia**.
2. **lode1** – typu int, ktorý udáva počet lodí s dĺžkou jedno políčko.
3. **lode2** – typu int, ktorý udáva počet lodí s dĺžkou dve políčka.
4. **lode3** – typu int, ktorý udáva počet lodí s dĺžkou tri políčka.
5. **lode4** – typu int, ktorý udáva počet lodí s dĺžkou štyri políčka.  
   (Atribúty lode1 – 4, nie sú static final, pretože sa nesôr ich hodnota mení)

Trieda **Kontrola** obsahuje nasledovné metódy:

1. **Konštruktor,** s parametrom pole typu Polia, vytvorí s atribútom **pole** asociáciu s triedou **Polia**.   
   **Konštruktor** tiež priradí nasledovným atribútom hodnoty: lode1 = 4, lode2 = 3, lode3 = 2 a lode4 = 1.
2. Metóda **getLode1**, s návratovou hodnotou typu int, vráti hodnotu atribútu **lode1**.
3. Metóda **getLode2**, s návratovou hodnotou typu int, vráti hodnotu atribútu **lode2**.
4. Metóda **getLode3**, s návratovou hodnotou typu int, vráti hodnotu atribútu **lode3**.
5. Metóda **getLode4**, s návratovou hodnotou typu int, vráti hodnotu atribútu **lode4**.
6. Metóda **kontrola**, s návratovou hodnotou typu void a parametrami int stlpec a int riadok, kontroluje či hráč po streľbe nezničil celú loď. Ak hráč zničil celú loď, označia sa všetky políčka okolo lode, z dôvodu, že dve lode, nemôžu byť vedľa seba a zníži sa počet lodí danej lode.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **Strielanie** obsahuje 3 atribúty:

1. **pole** – typu **Polia**, ktorý napĺňame objektami triedy **Polia**.
2. **kontrola** – typu **Kontrola**, ktorý napĺňame objektami triedy **Kontrola**.
3. **hrac** – typu **Hraci**, ktorý napĺňame objektami triedy **Hraci**.

Trieda **Strielanie** obsahuje nasledovné metódy:

1. **Konštruktor**, s parametrami Polia pole a Hraci hrac, vytvorí s atribútmi **pole** a **hrac** asociácie s triedami **Polia** a **Hraci**. A s atribútom **kontrola** kompozíciu s triedou **Kontrola**.
2. Metóda **strielanieHrac**, s návratovou hodnotou typu void, ktorá zisťuje či sa na zadanej pozícii v dvojrozmernom poli **pole**, nachádza loď, ak áno označí ju ako zasiahnutú a skontroluje pomocou **kontrola**, či je daná loď už celá zničená.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **Zapisovanie** obsahuje 3 parametre:

1. **pocetLodi** – typu int, ktorí ukazuje koľko lodí ešte môže hráč zadať.
2. **pole** – typu **Polia**, ktorý napĺňame objektami triedy **Polia**.
3. **hrac** – typu **Hraci**, ktorý napĺňame objektami triedy **Hraci**.

Trieda **Zapisovanie** obsahuje nasledovné metódy:

1. **Konštruktor**, s parametrami Polia pole a Hraci hrac, vytvorí s atribútmi **pole** a **hrac** asociácie s triedami **Polia** a **Hraci**. A atribútu **pocetLodi** priradí hodnotu 10.
2. Metóda **zadavanieLodeHrac**, s návratovou hodnotou typu void, ktorá umožňuje hráčovi zadávať lode do dvojrozmerného poľa, pričom hráč zadáva smer lode, a jej súradnice, a metóda kontroluje, či sa lode náhodou neprekrývajú alebo nie sú hneď vedľa seba, pričom počet lodí, ktoré môže hráč zadať je **pocetLodi**.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **Hraci** obsahuje 4 atribúty:

1. **posledneId** – typu int, ktorý je static, čiže atribútom triedy a jeho hodnota je 100.
2. **iD** – typu int, ktorý udáva aké identifikačné číslo sa priradí konkrétnemu hráčovi.
3. **nickName** – typu String, ktorý si hráč zvolí sám ako svoje meno.
4. **cislo** –typu int, ktorému sa priradí hodnota identifikačného čísla posledného hráča.

Trieda **Hraci** obsahuje nasledovné metódy:

1. **Konštruktor**, s parametrami String nickName a PosledneId cisloId, vytvorí asociáciu s triedou **PosledneId**, pričom atribútu **cislo** priradí hodnotu metódou cisloId.**vyberZoSuboru**();, atribútu **posledneId** priradí hodnotu atribútu **cislo**, atribútu **iD** priradí hodnotu atribútu **posledneId** a atribútu **nickName** priradí parameter nickName.
2. Metóda **getNickName**, s návratovou hodnotou typu String, vráti hodnotu atribútu **nickName**.
3. Metóda **getId**, s návratovou hodnotou typu int, vráti hodnotu atribútu **iD**.
4. Metóda **toString**, s návratovou hodnotou typu String, vráti zadaný reťazec, ktorý obsahuje meno hráča a jeho identifikačné číslo.
5. Metóda **getTabulka**, s návratovou hodnotou typu String[], a parametrom cislo typu int vráti pole reťazcov, ktoré obsahuje meno hráča, identifikačné číslo a parameter cislo.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **PosledneId** obsahuje nasledobné metódy:

1. Metóda **vlozDoSuboru1**, ktorá je metódou triedy, s návratovou hodnotou typu void, a parametrom cislo typu int, zapíše do konkrétneho súboru číslo, ktoré získa z parametru cislo.
2. Metóda **vlozDoSuboru**, ktorá je metódou triedy, s návratovou hodnotou typu void, zapíše do konkrétneho súboru číslo 99. Táto metóda sa spúšťa ak je súbor prázdny aby pri čítaní nedošlo k chybe.
3. Metóda **vyberZoSuboru**, ktorá je metódou triedy, s návratovou hodnotou typu int, vráti prečítané číslo zo súboru (predstavuje identifikačné číslo posledného hráča) a spustí metódu **vlozDoSuboru1** pričom ako parameter zadá prečítané číslo + 1.



Trieda **Tutorial** obsahuje nasledujúce metódy:

1. Metóda **tutorial**, s návratovou hodnotou typu void, ktorá sa hráča spýta, či chce alebo nechce spustiť tutoriál, a ak chce, tak následne do terminálu vypíše pokyny k hre.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **TopSkore** je v asociácii s triedou **Hraci**.

Trieda **TopSkore** obsahuje nasledujúce metódy:

1. Metóda **vlozDoSuboru**, ktorá je metódou triedy, s návratovou hodnotou typu void, zapíše do konkrétneho súboru číslo 0. Táto metóda sa spúšťa ak je súbor prázdny aby pri čítaní nedošlo k chybe.
2. Metóda **vlozDoSuboru1**, ktorá je metódou triedy, s návratovou hodnotou typu void, a parametrom zoznam typu int[], zapíše do konkrétneho súboru čísla z parametru, ktoré predstavujú výsledky hráčov, ktorí vyhrali.
3. Metóda **vyberZoSuboru**, ktorá je metódou triedy, s návratovou hodnotou typu String, a parametrami int cislo a Hraci hrac1, vráti reťazec, ktorý vypíše do terminálu hodnotenie vyhratého hráča a spustí metódu **vlozDoSuboru1** pričom ako parameter použije zoznam typu int[], v ktorom sú prečítané čísla a parameter cislo.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **Top3Skore** je v asociácii s triedami **Tabulka** a **Hraci**.

Tried **Top3Skore** obsahuje nasledovné metódy:

1. Metóda **vlozDoSuboru,** ktorá je metódou triedy, s návratovou hodnotou typu void, ktorá do súboru, ak je prázdny, zapíše pripravené hodnoty.
2. Metóda **vlozDoSuboru1**, ktorá je metódou triedy, s návratovou hodnotou typu void, a parametrom topSkore typu int[], zapíše do konkrétneho súboru čísla z parametru, ktoré predstavujú identifikačné číslo a výsledky prvých 3 hráčov.
3. Metóda **vyberZoSuboru**, ktorá je metódou triedy, s návratovou hodnotou typu int[], a parametrami int cislo a Hraci hrac1, vráti pole čísel, ktoré predstavujú identifikačné číslo a výsledky prvých 3 hráčov a spustí metódu **vlozDoSuboru1** pričom do nej vloží pole čísel.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s vysokou spoľahlivosťou

Trieda **Tabulka** je v asociácii s triedou **Hraci**.

Trieda **Tabulka** obsahuje nasledujúce metódy:

1. Metóda **tabulka,** s návratovou hodnotou typu void a parametrami int[] topSkore1, Hraci hrac a int cislo, vypíše do terminálu tabuľku, ktorá obsahuje identifikačné číslo a výsledky prvých 3 hráčov a meno, identifikačné číslo a výsledok aktuálneho hráča, ktorý vyhral hru.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **RankHraca** je v asociácii s triedou **Rank**.

Trieda **RankHraca** obsahuje nasledujúce metódy:

1. Metóda **rankHraca**, s návratovou hodnotou typu Rank a parametrom cislo typu int, vráti, ktorí rank si získal vzhľadom na svoj výsledok, ktorí predstavuje parameter cislo.

Obrázok, na ktorom je text

Popis vygenerovaný s vysokou spoľahlivosťou

Trieda enum **Rank** obsahuje 20 atribútov:

1. KID,
2. CIVILIAN,
3. TRAINING\_PRIVATE,
4. PRIVATE\_FIRST\_CLASS,
5. SPECIALIST,
6. CORPORAL,
7. SERGEANT,
8. STAFF\_SERGEANT,
9. FIRST\_CLASS\_SERGEANT,
10. MASTER\_SERGEANT,
11. FIRST\_SERGEANT,
12. SERGEANT\_MAJOR,
13. COMMAND\_SERGEANT\_MAJOR,
14. SERGEANT\_MAJOR\_OF\_THE\_ARMY,
15. MAJOR\_ADMIRAL,
16. LIEUTENANT\_ADMIRAL,
17. ADMIRAL,
18. ADMIRAL\_OF\_THE\_ARMY,
19. LEGENDARY\_ADMIRAL,
20. CHUCK\_NORRIS;

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s veľmi vysokou spoľahlivosťou

Trieda **Uvod** obsahuje nasledujúce metódy:

1. Metóda **uvod**, s návratovou hodnotou typu void, ktorá do terminálu vypíše pripravené reťazce.
2. Metóda **zaver**, s návratovou hodnotou typu void, a parametrom hrac typu hraci, ktorá do terminálu vypíše pripravené reťazce.

Obrázok, na ktorom je snímka obrazovky

Popis vygenerovaný s vysokou spoľahlivosťou

Trieda **Aplikacia** je v kompozícii s triedami **Polia**, **Strielanie**, **Zapisovanie**, **Hraci**, **PosledneId**, **Tutorial**, **TopSkore**, **Top3Skore**, **RankHraca**, **Uvod**.

Trieda **Aplikacia** obsahuje nasledujúce metódy:

1. Metóda **main**, ktorá je metódou triedy, s návratovou hodnotou typu void, a parametrom args typu String[], využíva všetky predchádzajúce metódy a spája ich do jedného celku.

Obrázok, na ktorom je text, mapa

Popis vygenerovaný s veľmi vysokou spoľahlivosťou