

Console game

Сложност: 10/10

Позволени езици: C++, C#, Java

1. Общо описание

Целта на проекта е да бъде създадена конзолна игра, подобна на TBS (Turn-based strategy) игрите. В играта ще се изправят една срещу друга две армии, като битките се развиват на ходове.

2. Описание на задачата

Трябва да бъде реализирана следната функционалност:

1. Класове представляващи различните единици в играта: абстрактен клас **Creature**, класове **Peasant, Footman, Archer, Griffon, Hero**

- В **Creature** да има:

- член-променливи:

- **damage** (int), **defense** (int) - defense определя с колко ще бъде намалена противникова атака
health (int), **mana** (int), **stamina** (int) - stamina определя колко квадратчета напред може да се придвижи дадено същество, **critChance** (int) - critChance определя възможността съществото да нанесе двоен damage при атака

- член-функции:

- **конструктори**

- **get**-ери и **set**-ери за член-променливите

- чисто виртуална функция **Attack**

- чисто виртуална функция **Move**

- **Peasant, Footman, Archer, Griffon, Hero** наследяват **Creature** и **override**-ват **Attack** и **Move**

2. Клас **Player**, който ще представлява играч. В него има текущото количество злато, което има играчът и масив от единици, които притежава (vector от `pair<Creature*,int>`). Началното злато на даден играч е 300g.

3. Клас **Shop**. В него трябва да има `map`, в който `key` е тип на единица, а `value` е цената на единицата (int). В магазина играчът да може да закупи единици, чрез златото, което има. В магазина единиците да има следните цени: **Peasant - 30g, Footman - 90g, Archer - 50g, Griffon - 150g**. Играчът може да закупува безкраен брой единици, стига да има нужното злато.

4. Клас **Game**. В него трябва да има 6 основни неща:

- член-променлива за играча (**m_player**) и член-променлива за противника му (**m_enemy**) – и двете от тип **указател към Player**

- **конструктор** - в него трябва да бъдат заредени единиците, които са достъпни за закупуване и цените им в магазина. Също така към `m_enemy` трябва да бъдат добавени единици, за да може по-късно да участва в битка.

- **основно меню:**

- то да предоставя следните 3 команди:

- започване на игра

- посещаване на магазина

- излизане от играта

- меню за магазина:

- то да предоставя следните 3 команди:

- изкарване на всички единици, които могат да бъдат закупени, както и техни цени
- закупуване на определен брой единици. Командата да е **buy**

<peasant|archer|footman|griffon> <amount>. Например командата **buy griffon 10** ще закупи 10 грифона и ще ги добави към единиците на играча. Ако играчът няма достатъчно злато за покупката, да бъде изкарано съобщение колко злато не му достига за покупката.

- изкарване на наличното състояние злато, което има играчът

- меню за същинската игра

- на разположение да са командите:

- **преместване** на дадена единица на позиция на полето - да бъде реализирана чрез командата **move <source coords> <destination coords>**. Например ако на позиция (2,2) имаме грифон и искаме да го преместим на позиция (5,5), то командата ще е **move (2,2) (5,5)**. Тук трябва да се правят проверки дали **destination coordinates** не е заето от друга единица и дали единицата може да ходи толкова далеч

- **нападане** на противникова единица - командата да е **attack <source coords> <destination coords>**. Например ако на (4,4) имаме противников Footman, а на (2,2) имаме наш Peasant, с **attack (2,2) (4,4)** ще преместим Peasant-а на квадратче, което е на 1 единица разстояние (например (3,3)) и ще нападнем противниковия Footman.

- **принтиране** във файл текущото състояние на полето - как са разположени единиците:

- нашите единици да бъдат отбелязани така: P за Peasant, A за Archer, F за Footman, G за Griffon, H за Hero.

- противниковите единици да бъдат отбелязани така: EP за Peasant, EA за Archer, EF за Footman, EG за Griffon, EH за Hero.

- под таблицата да бъде изкарано колко единици има от всеки тип в момента на полето
Например: P 10, A 14, F 20, G 3, H 1, EP 20, EA 40, EF 30, EG 8, EH 1

- **приключване** на текущия ход. Когато приключим хода, противниковият играч трябва да изиграе хода си и да разположи своите единици, след което отново ние сме наред.

5. Игралното поле се представя чрез матрица **10x10**, като нашата и противниковата армията се разполагат върху нея. За целта трябва да бъде създаден клас **Battlefield**, който ще представлява полето. Героят на играча и противниковият герой са извън бойното поле.

6. Единиците имат различна stamina - колко квадратчета далеч могат да се придвижат, в радиус от текущата позиция: Peasant – 3, Footman – 4, Archer – 3, Griffon – 5, Hero не може да се мести по полето

7. Единиците имат различен обхват на атака, в радиус от текущата им позиция:

Peasant – 1, Footman – 1, Archer – може да напада всяка противникова единица по полето, независимо от разстоянието, Griffon – 2, Hero – всяка противникова единица

8. Единиците имат следният damage: Peasant – 1, Footman – 3, Archer – 2, Griffon – 4, Hero – 2
На игралното поле на 1 квадратче се разполагат всички единици от един тип. Когато нападнат те правят комбиниран damage. Ако на (2,3) имаме група от 10 Footman, който нападнат противникова група от 5 Peasant на (3,3), то те ще направят 30 damage и той ще бъде разпределен на групата, като се вземе в предвид кръвта и защитата на всеки Peasant. Накрая бройката загинали Peasant-и ще бъдат приспаднати от групата.

9. Освен, че играчът губи умрелите по време на игра единици, ако след края на дадена битка е загубил играта - губи и 100g. Ако е спечелил - печели 200g.

Играта приключва когато един от играчите остане без единици на полето.

10. Трябва да бъде реализиран базов изкуствен интелект, така че противникът да разполага единиците си по полето и да се бие равностойно с играча. На един ход единиците на противниковия играч изпълняват действие една след друга. На един ход една единица може да изпълни само едно действие - преместване или атака. Аналогично за нашите единици трябва да бъде отбелязано, че една единица вече е изпълнила действие и ние вече не можем да я използваме.

11. Окончателно в main функцията трябва да имате следния код:

```
int main()
{
    Game game;
    game.Run();
    return 0;
}
```

Абсолютното следване на структурата на кода и примерните стойности, описани в отделните точки не е задължително. За някой от характеристиките (например **health** и **defense** на Creature) трябва да измислите собствени стойности.

Допълнително в играта могат да бъдат добавени магии, които могат да изпълняват единиците. Проявата на творчество ще бъде оценена подобаващо.



Reference: Heroes of Might and Magic