

Praktikum ‚Objektorientierte Programmierung‘

Aufgabenblatt 11

Wie in der Vorlesung besprochen, empfehle ich die Compiler-Flags `-Wall` und `-Wextra`. Um möglichst im Einklang mit der Vorlesung zu sein, sollten Sie dafür sorgen, dass Ihr Compiler den C++14-Standard unterstützt. In keiner der Aufgaben darf der *globale* Namensraum genutzt werden. Definieren Sie einen eigenen Namensraum. Die Anweisung `using namespace` darf nicht genutzt werden.

Aufgabe 1 ‚08 Die Container der STL‘:

Entwickeln Sie eine Funktion

```
void sort(std::vector<Position>& positions)
```

die den Vektor sortiert. Dabei darf selbstverständlich keine Sortierfunktionen aus einer Bibliothek genutzt werden. Um die Sache interessant zu gestalten, ist die explizite Nutzung von Vergleichsoperatoren nicht zulässig. Nutzen Sie den Datentyp `std::map`.

Aufgabe 2 ‚09 Vererbung‘ und ‚10 Polymorphie‘:

Für die Figuren eines Fantasy-Spiel entwickeln Sie eine (abstrakte) Klasse `Character` mit der folgenden Schnittstelle:

```
const std::string &getName() const  
bool isAlive() const  
void hit()  
virtual void applyWeapon(Character& character)
```

Jede Figur hat eine Zahl Lebenspunkten. Genau dann, wenn eine Figur keine Lebenspunkte mehr hat, liefert die Methode `isAlive` den Wert `false`. Mit der Methode `hit` wird die die Anzahl der Lebenspunkte um einen reduziert. Mit der Methode `applyWeapon` wird die Anzahl der Lebenspunkte der angegriffenen Figur reduziert. Die Zahl dieser der Lebenspunkte ist von der jeweiligen Figur abhängig. Beachten Sie aber, dass nur lebende Figuren angreifen können und angegriffen werden können. Eine Figur kann sich nicht selbst angreifen!

Es gibt die folgenden konkreten Figuren:

- **Dwarf**: Jeder Zwerg hat eine Axt. Das Gewicht der Axt liefert die Methode `getAxeWeight`. Der Hieb mit einer Axt kostet die angegriffene Figur drei Lebenspunkte.
- **Elf**: Jeder Elb hat Pfeile. Das Anzahl der Pfeile liefert die Methode `getArrows`. Jeder Treffer eines Pfeils kostet die angegriffene Figur zwei Lebenspunkte. Ein Elb kann selbstverständlich nur angreifen, wenn er noch Pfeile hat.
- **Orc**: Jeder Ork hat ein Schwert. Die Länge des Schwertes liefert die Methode `getSwordLength`. Jeder Treffer mit dem Schwert kostet die angegriffene Figur einen Lebenspunkt.

Erzeugen Sie einige Figuren, legen Sie sie in einem Vektor ab und lassen Sie sie solange gegeneinander kämpfen, bis nur noch eine lebt. Den Namen der überlebenden Figur geben Sie aus.

Kümmern Sie sich unbedingt um eine *redundanzarme* Implementierung.

