

Praktikum ‚Objektorientierte Programmierung‘

Aufgabenblatt 8

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.

Für alle Klassen müssen Deklaration und Implementierung getrennt sein. Bei der Übersetzung werden nicht nur Fehler, sondern auch *Warnungen* und *Verbesserungshinweise* angezeigt. Ignorieren Sie das nicht, sondern nutzen Sie diese Hinweise, um Ihren Code zu verbessern. *Ergreifen Sie aber nur Maßnahmen, die Sie auch verstehen.*

Aufgabe 1:

Ändern Sie die Schnittstelle der Klasse `City`. Die Methoden `getPOI` und `setPOI` sind ab jetzt `private`. Die Schnittstelle soll jetzt den Zugriff über den Indexoperator ermöglichen. Recherchieren Sie dazu *sorgfältig*. Der folgende Code muss fehlerfrei ausgeführt werden. Die letzten beiden Zeilen bedürfen einiger Überlegung.

```
auto city=createCity();
assert(city[0]=="Sauron");
assert(city[1]=="Minas Morgul");
assert(city[2]=="Orodruin");
try {
    city[3];
    assert(false);
}catch (...){}
try {
    city[-1];
    assert(false);
}catch (...){}

city[0]="Gandalf";
assert(city[0]=="Gandalf");

try {
    city[3]="Gandalf";
    assert(false);
}catch (...){}
try {
    city[-1]="Gandalf";
    assert(false);
}
```

```
}catch (...){}  
  
const auto constCity=createCity();  
assert(constCity[0]=="Sauron");
```

Die Funktion `createCity` ist dabei wie folgt definiert:

```
hfu::City createCity(){  
    std::string morder = "Mordor";  
    std::string pois[]={ "Sauron", "Minas Morgul", "Orodruin"};  
    hfu::City result(morder, 47, 11, pois, 3);  
    return result;  
}
```

Aufgabe 2:

Ändern Sie die Klasse `City` so, dass der folgende Code fehlerfrei ausgeführt wird:

```
std::string pois_morder[]={ "Sauron", "Minas Morgul", "Orodruin"};  
hfu::City left("Mordor", 47, 11,pois_morder,3);  
std::string pois_arrakis[]={ "Arrakeen", "Desert"};  
hfu::City right("Arrakis", 23, 42,pois_arrakis,2);  
left=right;  
assert(left.getName()=="Arrakis");  
assert(left.getX()==23);  
assert(left.getY()==42);  
assert(left.getNumberOfPOIs()==2);  
assert(left[0]==pois_arrakis[0]);  
assert(left[1]==pois_arrakis[1]);
```