

Praktikum ‚Objektorientierte Programmierung‘

Aufgabenblatt 9

Aufgabe 2:

Einige von Ihnen kennen die beiliegende Datei `flights.csv` mit Informationen über Linienflüge noch aus dem Datenbanken-Praktikum. Eine beispielhafte Zeile lautet:

```
2013,1,1,542,540,2,923,850,33,"AA",1141,"N619AA","JFK","MIA",160,1089,5,40,2013-01-01 05:00:00
```

Entwickeln Sie eine Funktion

```
std::string extract_airline(const std::string& text)
```

die aus einer Zeile der Datei die Fluglinie ermittelt. Im Beispiel ist dies AA. Testen Sie Ihre Funktion.

Hinweis: Für die Lösung der Aufgaben können die beiden Methoden `find` und `substr` hilfreich sein. Im folgenden Beispiel sehen Sie, wie man sie nutzen kann.

```
std::string text="hello-hfu";  
assert(text.find("-") == 5);  
assert(text.substr(6,10) == "hfu");
```

Aufgabe 3:

Entwickeln Sie eine Funktion

```
std::map<std::string, int> create_frequencies(const std::string& filename)
```

der Sie den Namen der Datei `flights.csv` übergeben können und die eine Map liefert, die für jede Fluglinie die Anzahl der Flüge dieser Fluglinie liefert, die in der Datei `filename` enthalten sind. Insbesondere muss der folgende Test bestanden werden:

```
std::string filename="flights.csv";  
auto airlines=create_frequencies(filename);  
assert(airlines["AA"]==32729);  
int size=airlines.size();  
assert(size==18);
```

Hinweis: eine Datei kann man in C++ wie folgt lesen:

```
std::ifstream in(filename);  
assert(in.is_open());  
std::string line;  
while (in >> line){  
    std::cout<<line<<std::endl;  
}  
in.close();
```

Aufgabe 3:

- a. Was kann man an der Funktion `create_frequencies` verbessern? Einen Hinweis - eigentlich schon fast die Antwort - liefert die folgende Teilaufgabe.
- b. Ändern Sie `create_frequencies` so, dass sie keinen Wert liefert, sondern einen geeigneten Smart-Pointer. Passen Sie auch den Test aus Aufgabe 2 an.