



414078-HS2018-0 - C++ Programming II

EXERCISE-02

TABLE OF CONTENTS

1	Introduction	1
2	Phone Book (easy)	1
3	Billionaires of the World (not easy)	2
4	Submission	2

1 Introduction

This exercise of 414078-HS2018-0 will focus on the usage of STL containers, iterators and algorithms - in particular you'll work with map and pair in more detail. **Use STL whenever possible!**

2 Phone Book (easy)

Create a simple phone book with the content given in a vector. Choose the appropriate container.

```
vector<pair<string, string>>
vec{make_pair("Homer", "1111"),
    make_pair("Marge", "2222"),
    make_pair("Lisa", "3333"),
    make_pair("Magy", "4444"),
    make_pair("Bart", "5555")};
```

The program should ask the user to enter a name and present the number if exists and warn otherwise. Leave the phone book with Exit. Possible program flow:

```
Enter name (or Exit): Bart
Phone number: 5555
Enter name (or Exit): Homer
Phone number: 1111
Enter name (or Exit): Flenders
Name not in directory.
Enter name (or Exit): Exit
```

Change the program to allow multiple numbers per person, e.g. a mobile number. How can you achieve this?

3 Billionaires of the World (not easy)

Write a program that reads a list of billionaires and tells us the number of billionaires per country pointing out the richest person per country. The list is taken from Forbes 2018, *World's Billionaires* at <https://www.forbes.com/billionaires//list/> and has about 400 entries.

- ▶ Read the file containing the name, dollars in billions and the country which are separated by tabs. Create a class Billionaire and overload the istream » and ostream « operators to conveniently read the file into a vector and write the content to the output as shown in the code snippet.

```
ifstream stream("Forbes2018.txt");
if(!stream)
    cout << "WARNING: File not found!" << endl;

vector<Billionair> billionaires;
copy(istream_iterator<Billionair>(stream), istream_iterator<Billionair>(),
    back_inserter(billionaires));
copy(billionaires.begin(), billionaires.end(),
    ostream_iterator<Billionair>(cout, "\n"));
```

- ▶ Create a map which maps the country string to a pair. The pair contains a copy of the first billionaire of every country from the list and a counter to count the number of billionaires per country:

```
map<string, pair<const Billionair, size_t>> m;
```

- ▶ Now the key part: For each item in the vector try to emplace (m.try_emplace) a new pair for every country. If successful, the new item with count=1 is inserted into the map - it will be the richest person automatically as defined in the file. If not successful, increment the counter.
- ▶ The first lines of the output should look similar to:
Australia : 5 billionaires. Richest is Gina Rinehart with 17.4 B\$
Austria : 2 billionaires. Richest is Dietrich Mateschitz with 23 B\$
Belgium : 1 billionaires. Richest is Albert Frere with 6.2 B\$
Brazil : 9 billionaires. Richest is Jorge Paulo Lemann with 27.4 B\$
Canada : 7 billionaires. Richest is David Thomson with 25 B\$
Chile : 1 billionaires. Richest is Iris Fontbona with 16.3 B\$
China : 40 billionaires. Richest is Ma Huateng with 45.3 B\$
...
▶ Sort the map according to:
 - ▶ the richest person
 - ▶ the country with the most billionairesand print out the 5 first items.

4 Submission

Submit your source code (as a zip-file) to Ilias **before the deadline** specified in Ilias.