Mittuniversitetet
MID SWEDEN UNIVERSITY

Lab 5 – Standard libraries and generic algorithms

Object-oriented programming in C ++

The use of generic algorithms and function objects.

Lab 5 – Standard libraries and generic algorithms

The purpose of the lab is to get practical experience of function object, templates, list, iterators, and generic functions from the STL.
Text written in italics are the names of the components used in the task.

Assignment
You will write a program to shuffle and process 20 random numbers in the range [1000-2000]. The type of numbers will be of either type int or double. The numbers will be stored in a `list`. Selection of data type (int / double) will be made first in the program. After the program has started, the list should be empty.

The content of the list should be able to treat as follows:

a) Fill the list with the new random numbers (`generate` with the appropriate function's object. Randomization should be done with components from the header file `<random>).`
b) Calculate the sum of all the numbers `(Accumulate)`
c) Calculate the average of all numbers `( iterators and function template)`
d) Search the first number in the interval [1500, 1900] (`find_if` with the appropriate factions object)
e) Divide any number by 2 (`transforme with the appropriate predefined function objects`)
f) Swap the first and last elem, the second largest and second last elem etc. `(Iter_swap)`
g) Search for the largest and smallest numbers `(max_element and min_element)`
h) sort the numbers in ascending order `(suitable form of sort)`
i) write the contents of the list to a file (`copy and ostream_iterator`)
j) empty the list
k) read values from the file to the list (`copy, istream_iterator and back_inserter`)
l) print all the numbers `(for_each)`

The above operations will be applied to the selected type, int or double.

The program will be menu driven.
The test program should utilize function's templates.
The code must not be **'doubled'** such that **'same'** function is written in an int version and a float version.

Tip: Since the user selected int or float creates a List <int> or List <float>. Let this be the argument to a function template from which the rest of the work is done:

```
template<typename T>
void myMainFunc<list<T> &theList) {
	Här är T den valda typen.
}
```