C++ program structure. Input and output. Common errors. Algorithm implementation

In Listing 1 the basic structure of a C++ program is given. The basic expressions are followed by comments – special parts of the program text which are placed after the symbol // or between the symbols /* */ and are ignored by the compiler.

```cpp
1  #include <iostream>   // stream input/output header
2
3  using namespace std;  // standard namespace
4
5  int main()            // every program has a main function which is unique
6  {
7      /*
8      place your code here
9      */
10     return 0;         // message to OS that the program executed successfully
11 }
```

Listing 1: Basic structure of a C++ program

Exercise 1. Rewrite the "Hello, World!" program consecutively omitting one of the basic expressions in Listing 1. For example, in your first attempt you have to omit only line 1, in the second attempt omit only line 3, etc. On each attempt try to compile the program. What are the compilation results?

Exercise 2. Listing 2 contains a program which asks the user for his name and prints out a greeting. Compile and run this program. Do not forget to hit the Enter key when you type your name.

```cpp
1  #include <iostream>
2  #include <string>                  // header for string data type
3
4  using namespace std;
5
6  int main()
7  {
8      cout << "Hello, what is your name?" << endl;
9      string user_name;         // string which will contain the user name
10     getline(cin, user_name);  // read whole line from the standard input
11     cout << "Hello, " << user_name << ", I am glad to meet you." << endl;
12     return 0;
13 }
```

Listing 2: name.cpp

Exercise 3. Modify the program in Listing 2 so it will ask a student for his name and faculty number. After that, the program will display the data, provided by the user.

Exercise 4. Write a program which reads two integer numbers from the standard input and displays their sum in the standard output. You can read an integer and store it in the computer memory using the code given in Listing 3.

```
1  int a;      // declare an integer variable
2  cin >> a;   // read an integer from the standard input
```

Listing 3: Read and store an integer

You can print out the sum of two integers a and b by:

```
1  cout << a + b << endl; // display the sum and a new line
```

Listing 4: Read and store an integer

Exercise 5. Do the following modifications of the program in Listing 2. What is the compiler response and the program execution? Each of the modifications must be done separately.

- Line 4: remove the semicolon. **Each statement ends in a semicolon.**

- Line 6: instead of main type Main. **C++ is *case-sensitive*.**

- Line 8: instead of name type game. **Logical errors are difficult to track down.**

Exercise 6. Write a program that prints the balance of an account that earns 5 percent interest per year after the first, second and third year. Follow the pseudocode given below:

Print("Initial balance? ")
Read(balance)
balance ← balance * 1.05
Print("First year: ") Print(balance)
balance ← balance * 1.05
Print("Second year: ") Print(balance)
balance ← balance * 1.05
Print("Third year: ") Print(balance)

- C++ is a *strictly typed* language. What type has to be the *variable* balance?

- You can read data from the standard input and place it in a variable using the entity cin, for example: cin >> balnce;

- Modify this program so it will ask the user for the interest in percentage as well.