

Written Homework #1

CS 162: Introduction to Computer Science

Alves Silva, Otavio Augusto
PSU ID: 902840168

Create an algorithm

1. I must be responsible with my timetable.
 - 1.1 I must not be late.
2. I must have interested about the class.
 - 2.1 I have to pay attention in teacher explanation.
 - 2.2 I must have focus and leave things that can take my attention during the class.
3. I must be organized.
 - 3.1 I need take notes with key words and important topics of the lecture.
 - 3.2 I need start do my homework some days before the due day.
4. I must start think like a programmer.
 - 4.1 He must make an algorithm before start the code.
 - 4.2 He should ask oneself how to solute the problem.
 - 4.3 He should try to do a code that everybody can understand, even a person who doesn't know anything about programming.
5. I must read the books, documentation, and everything that can help me to improve my knowledge about the language and algorithms.
6. If I have some doubts about the lecture.
 - 6.1 I should read about it in the book.
 - 6.2 I can go to the FAB and see the assistants to clear my doubts.
7. If I have some programs with my program.
 - 7.1 I must try resolve alone first and think where I did a mistake.
 - 7.2 I should read about it in the book.
 - 7.3 If even after these steps I didn't find my answer, I can go to the FAB and see the assistants to clear my doubts.

Ethics

To explain in my words what is “computer ethics”. I need first explain what is “ethics”. Basically, ethic is a set of moral rules that a person or a community have in their life. Therefore, “Computer ethics” are a group of moral rules that people linked to this area of knowledge need to have at the moment that they are programming, building, creating and using anything which has relation with computers.

For example, when a programmer are developing software that will possess some private information of user. The programmer must have an extremely care with the security of his software because if any information is stolen in reason of flaws in the software, it will be a fault of the programmer. When a programmer are coding, he must have consciousness that even good software can be used to do bad things. For example,

software was developed by the police company to access private information of criminals in social network. However, the software was stolen and when was programmed the programmer didn't make an access protection, so any one can use even without permission. Another important example is when someone uses paid software illegally. In other words, without the permission of the creator and without pay anything to use.

Style

A code must have some characteristics and styles that make its readability and maintainability. A code has readability when it can be read easily by anyone, even if this person does not know anything about programming. Therefore, the code must be easy and uncomplicated.

For example, I want a code to calculate the volume of a rectangle solid, if I could choose between code A and code B. Probably I will choose a style of coding like the code A. You can see the difference bellow.

```
1 #include <iostream>
2 using namespace std;
3
4 // Program to calculate the volume of a rectangle
5 int main(){
6     // Variables of the rectangle measures
7     int length;
8     int width;
9     int height;
10    int volume;
11    cout << "CALCULATOR OF VOLUME OF A RECTANGLE\n";
12    // Receiving measures of the user
13    cout << "Please, enter the length: ";
14    cin >> length;
15    cout << "Please, enter the width: ";
16    cin >> width;
17    cout << "Please, enter the height: ";
18    cin >> height;
19    // Calculing the volume
20    volume = length * width * height;
21    // Showing the result
22    cout << "The volume of this rectangle is: " << volume << "\n";
23 }
```

Code A

```
1 #include <iostream>
2 using namespace std;
3
4 int main(){
5     int l;
6     int w;
7     int h;
8     int v;
9     cout << "VOLUME OF A RECTANGLE\n";
10    cout << "Lenght: ";
11    cin >> l;
12    cout << "Width: ";
13    cin >> w;
14    cout << "Height: ";
15    cin >> h;
16    v = l * w * h;
17    cout << "Volume: " << v << "\n";
18 }
19
```

Code B

Both codes do the same thing, but the code A can be read and understood more clearly because the functions of his variables are clear and the comments help the user who would be reading the code or the programmer in the future.

Another important concept of a code is his maintainability. The maintainability of a code is the ability to be cohesive and clear, even with the increase of code lines over time. Furthermore, having the ability to be changed without all the code needs to be restarted from zero.

C++

I took these three different keywords, which are “cout”, “if”, and “while”, from the book. The keyword “cout” is a stream object in C++ to output information in the screen. It is the most common keyword in C++ because almost all the codes need this stream object to inform or to request something from the user. To be used the “count” must be used together with the insertion operator (<<). For example, to print a string we use the following line of the code:

```
cout << "HelloWorld!";
```

The “count” has the same function in Python. However in Python, this keyword is called “print” and doesn’t need the insertion operator, but it uses brackets. For example, to print the same string in Python we use the following line of the code:

```
print ("HelloWorld!")
```

Another C++ keyword is the “if”. This keyword is used to execute a statement if a condition is met. With this, the code has the ability to maintain an operational flow. To check if a number is greater than zero we would use the keyword if to help us. For example:

```
if(number > 0)
{
cout <<"The number "<< number << " is greater than zero";
}
```

In Python, the keyword “if” has the same function and is called by the same away. However, it isn’t required brackets in the condition, but for start the statement that will be executed we use a colon after the finish of the condition and the statement must have a correct indentation because the indentation shows when the statement begins and finishes. For example:

```
if number > 0:
    print ("The number ", x, " is greater than zero.")
```

The keyword “while” is statement used to do loops or to repeat a statement for a number of times until the condition is fulfilled. In other words, the loop will happen until the moment that the condition is not truer. For example, if we have a problem to decrease a number until zero, it can be done with the help of the keyword “while”. For example:

```
int n = 5;
while(n>=0)
{
    --n;
}
```

In Python, the keyword “while” has the same use and function. However, the syntax is a little bit different. Like the keyword “if”, the condition expression of the “while” doesn’t need necessarily have brackets, but for start the statement that will be retreaded we use a colon after the finish of the condition and the statement must have a correct indentation because the indentation shows when the statement begins and finishes. For example:

```
n =5
while n >= 0:
    n -= 1
```

References

- MALIK, D.S., C++ Programming: From Problem Analysis to Program Design, Sixth Edition, 2013.
- C++ documentation
 - <http://www.cplusplus.com/doc/>
- Python documentation
 - <https://www.python.org/doc/>
- Readability and maintainability of a code
 - <http://www.cs.pomona.edu/classes/cs181f/slides/maintain.pdf>