

Coursework 2: Basic Programming

Héctor Menéndez¹

AIDA Research Group
Computer Science Department
Universidad Autónoma de Madrid

October 18, 2013

¹based on the original slides of the subject

Index

- 1 Control sentences
- 2 Conditional Sentences
- 3 Repetitive Sentences

Index

- 1 Control sentences
- 2 Conditional Sentences
- 3 Repetitive Sentences

Programming

- Computers are machines that sequentially execute the program code.
- But usually programmer wants to execute a specific part of the code several times, or depending on some condition execute, or not, a block of code.
- Control sentences allow programmer to modify the execution order of the program creating loops or executing certain parts of the code.
- The control sentences are if-else, switch, for, while, do-while sentences.
- C allows the use unconditional jumps, these sentences are usually gotos sentences but in structured programming is not recommended to use them.

Index

- 1 Control sentences
- 2 Conditional Sentences**
- 3 Repetitive Sentences

if-else

- `if-else` sentence is used when the execution of a code block depends on the result of a logical sentence.
- The structure of an `if-else` sentence is the following:

Listing 1: `if-else` declaration

```
if( condition ) {  
  ...  
  ...  
}  
else if{  
  ...  
}  
else{  
  ...  
}
```

if-else

- else and else if are optional.

switch

- The `switch` sentence executes different blocks depending on the value of a variable.
- The structure of a `switch` sentence is the following:

Listing 2: switch declaration

```
switch (<varName>){  
  case <value1 >:  
    ...  
  break ;  
  case <value2 >:  
    ...  
  default :  
    ...  
}
```


switch

- The sentence `break` is used within the cases to avoid the execution of other blocks.
- The `default` case represents all values not described by previous cases and it is optional.

Index

- 1 Control sentences
- 2 Conditional Sentences
- 3 Repetitive Sentences**

for

- `for` sentence is used to repeat a code block several times. It is composed by three parts:
 - **Initialization**: this is the starting point of the loop.
 - **Condition**: while the condition is true, the code block is executed.
 - **Update**: changes the value of the variables used by the loop.

Listing 3: `for` declaration

```
for(<initialization >; <condition > ; <update > ){  
  ...  
}
```

for

- For `var1 = 1` to 10, do something and increment the variable one unit in each step.

Listing 4: for declaration

```
for(int var1=1; var1<10; var1++){  
  // Do something  
}
```

while

- The `while` sentence needs a logical sentence and the code block will be repeated until the logical sentence is false.

Listing 5: while declaration

```
while(<condition >){  
  ...  
}
```

do while

- If you want to run the bucle at least one time, you can use a do-while loop.

Listing 6: do-while declaration

```
do{  
  ...  
}  
while(<condition >);
```