30.9.2011

Supporting Material for Teachers

Explaining basic computational concepts I

Algorithms...

What is an algorithm?

Algorithm is a predetermined series of instructions for carrying out a task in a finite number of steps.

In a way, an algorithm is a step by step description of a solution to the given problem. An algorithm must have a clear starting and stopping point.

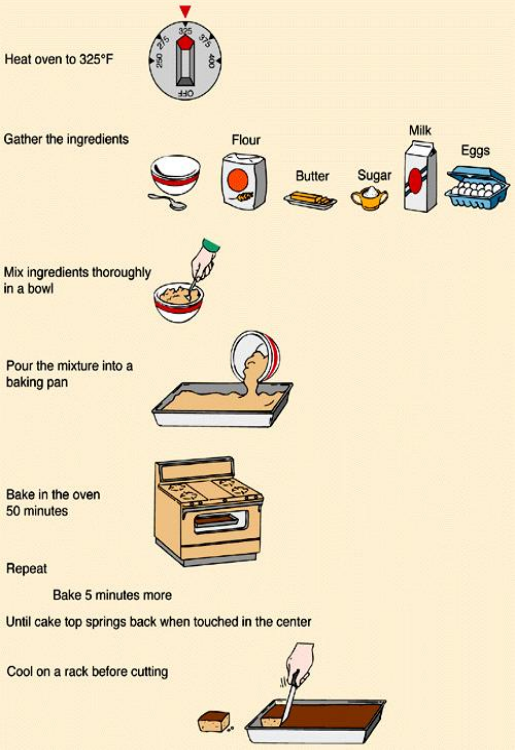
Composing elegant algorithms, that are **simple** and require the **fewest** steps possible -- is one of the principal challenges in programming.

How are they expressed?

Algorithms can be expressed in any language, from natural languages like English or French, to symbolic languages like charts and to programming languages like FORTRAN, C++ etc.

Is their any connection with real life?

We use algorithms every day. For example, a recipe for baking a cake is an algorithm.



Many applications that we use in everyday life are based on algorithms. In example: Automatic kiosks for bus and train tickets, 24 h bank machines (ATM), systems for receiving orders in restaurants, café and snacks vending machines and more

indices

It is either true or false...

A computer is capable of logic, or what we humans might call primitive 'reasoning'.

Using logic, a computer can compare values, and make decisions based on that comparison. Programmers use this 'logic' to write programs. The computer evaluates each argument/expression to true or false. In other words, it understands two states: true or false.

Example of logic expressions:

The week has 7 days. **(true)**

The day has 24 hours. **(true)**

Ice melts at 0 C degrees **(false)**

8 is bigger than 9. **(false)**

15– 5 equals to 10. **(true)**

10kgr equals to 10 gr. **(false)**

9 is an integer. **(true)**

This is not a guide for teachers. **(false)**

cMinds is about the development of analytical thinking. **(true)**

The Czech republic takes part in the cMinds project. **(true)**

cMinds project has Australian participants. **(false)**

The 2nd meeting of cMinds took place in Trondheim. **(true)**

Can you think of more sentences like these?

Conditional constructs

Conditional constructs are structures consisted of commands. These commands will be executed upon a condition (meaning a logic expression) that evaluates to true or false.

If the condition is true, a set of commands will be executed. Otherwise, if the condition is false, another set of commands, possibly embodied in the 'else part of the structure' will be executed .

A conditional construct takes two basic forms: IF statement and IF-then -else statement.

In the less complex form, if the *condition* evaluates to true, *statement_1* is executed. If the condition evaluates to false, statement_1 is not executed and the execution flow is transferred outside the body of the conditional construct.

```
IF (condition) then
Statement 1
```

Bear in mind that more than one statement can be placed in the body of the conditional statement.

In the more complex form if the *condition* evaluates to true, *statement_1* is executed; *statement_2* is executed only in case the *condition* evaluates to false.
After the execution the flow is transferred outside the body of the conditional construct.

```
IF (condition) then
Statement 1
Else
Statement 2
```

Again, more than one statement can be placed in the branches of the conditional statement.

From theory to practical examples...

```
IF (time = 08h30) then
Ring the school bell
```

What will happen if the time is 8h30?

If the time is 8h30 the condition evaluates to true. Thereby the school bell will be heard.

```
IF (weather= rainy) then
{
Take an umbrella
}
Take your bag
Lock the door
```

What actions will take place if the weather is rainy?

The avatar will take an umbrella and it will take its bag and it will lock the door.

What actions will take place if the weather is sunny?

The avatar will only take its bag and lock the door.

```
x = 0;
a = 0;
b = 1;
IF (x = 0) then{
   a= 1;
   b= 2;
}
x = a + b
```

Which is the value of x after the execution of the script?

Initially, x equals to 0.

The logic expression/condition is true so the statements in the IF– construct are executed. Thereby, a equals to 1 and b equals to 2. The flow is being transferred outside the conditional construct. x takes a new value. It is not zero anymore; it is 1 + 3. x equals to 3.

```
x = 0;
a = 0;
b = 1;
IF (x > 0) then{
   a= 1;
   b= 2;
}
x = a + b
```

Which is the value of x after the execution of the script?

Initially x equals to 0.

The logic expression/condition is false so the statements in the IF– construct are not executed. The flow is being transferred outside the conditional construct.

x takes a new value. It is not zero anymore; it is 0 + 1. x equals to 1.

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.