09/10/2017

Programming Languages
Specifications for
Assignments

Submission

- Assignments submitted after submission deadline (at most 2 days late) will be evaluated over 50. Do not send any e-mail 3 days after submission deadline.

- Collaboration on any assignment is strictly prohibited. Submitted assignments are automatically checked for similarities. Infractions will be given a zero for the entire assignment.

- Assignments must be submitted by e-mail. Every student must send his/her assignment to the following e-mail address.

  amac@yildiz.edu.tr

- **Subject of the e-mail** must contain course name, Assignment # and student number in specified format written below;

  Example Subject :

  BLM2541_Assignment_1_15011035

Content

1. An archive (zip, rar) file which contains only **ONE report file** and **ONE or MORE source files**. The question number indicates the number of source files.

2. The report file should include

- 1. Question - A brief description for each question
- 2. Solution - An explanation for each solution
- 3. Analysis - Numerous (at least 5, if necessary more) screenshots to show that your program runs correctly
- 4. Source Code - The source code (use Notepad++ or the equivalent to have colored codes)

About Source Code

The source file must include **comments** which explain the code.
The code should be **well-designed**.

About Comments

- Important algorithmic parts must be commented.
- You must write a heading comment describing you and the program
- In addition, you must write doxygen compatible comments for projects. Meaning that every function must have a comment on it starting with /** ending with */ having a brief description,
  - **@param** statements for every parameter and **@return** statement for the return values

****see example codes at the bottom**** Writing these comments improves readability of the code and enables to use document generation software (e.g. doxygen) for automatically generating documentation of the code.

If you want to learn further:
information about comments
information about documentation comments
information about doxygen and doxygen comments
doxygen website

3. The name of the submitted files must contain student number and question number. (q1 refers to "Question 1")

Example: zip/rar filename : 15011035.zip

The zip/rar should contain

pdf filename:	15011035.pdf
Source filename:	15011035_q1.c
Source filename:	15011035_q2.c

ASSIGNMENT EVALUATION CRITERIA

- Format
- Code
  - Readability (variable names, code organization)
  - Reliability (correctness of the code)
  - Effective use of the C Language
  - Modularity (effective use of functions)
- Analysis (Screenshots of different cases)
- Report Content and Organization

ATTENTION

- Do not use double filename extensions. For example, xxx.zip.rar, xxx.c.exe, … etc. Otherwise your mail will be blocked by spam filter and your assignment will not be evaluated.

- Assignments that don't comply with submission rules will **be evaluated over at most 80. It can decrease up to 50.**

EXAMPLE CODE-I

```
/**
@file
BBG2 spring2013 assignment 1.

A program that checks whether the number
is even or odd.
It prints 1 for odd numbers, 0 for even numbers

@author

Name:                   Bart Simpson

Student no:             08011001

Date:                   11/02/2013

E-Mail:                 bart_simpson@gmail.com

Compiler used:          GCC

IDE:                    CodeBlocks

Operating System        Windows 7
*/

#include<stdio.h>

/**
    Main function. Reads the number and prints the result
*/


int main(){

    int number,result;
    // reading the number:
    scanf("%d",&number);
    // checking whether the number is odd or even:
    if((number%2)==0){
        result=0;
    }
    else{
        result=1;
    }
    // output the result:
    printf("%d\n",result);

    return 0;
}
```

EXAMPLE CODE-II

```
/**
@file
BBG2 spring2013 assignment3.

A program that adds and multiplies two given numbers.

@author

Name:                   Homer Simpson

Student no:             08011002

Date:                   11/04/2012

E-Mail:                 homer.simpson@yahoo.com

Compiler used:          GCC

IDE:                    Netbeans 7.2.1

Operating System        Linux
*/

#include<stdio.h>

/**
    function which gives the sum of two integers.
    @param a first number
    @param b second number
    @return sum of a and b
*/
int add(int a, int b){
    int c
        c=a+b;
    return c;
}
/**
    function which multiplies two integers.
    @param a first number
    @param b second number
    @return multiplication of a and b
*/
int multiply(int a,int b){
    int c;
       c=a*b;
    return c;
}
/**
    Main function. Reads the numbers from the user.
    Then it outputs the sum and multiplication of two inputs.
*/
int main(){

    int a,b,sum,result;

    printf("Enter two numbers");

    scanf("%d",&a);
    scanf("%d",&b);


    // call add function to sum the given two numbers
    sum = add(a,b);

    // call multiply function to multiply the given two numbers
    result multiply(a,b);
```

```
    // output for sum operation
    printf("The result of addition is = %d", sum);

    // output for multiplication process
    printf("The result of multiplication is = %d",result);

    anykey();
    return 0;
}
```