GIT CSE102 HW02

Fall 2014

Due Date

02.10.2014, 23:59

Implement the functions described below.

1. Weight Analyzer: Write a function that computes Body mass index of the user. Your function should get weight, height information as parameter. Computed BMI using the following formula;

$$\mathrm{BMI} = \frac{\mathrm{mass(kg)}}{(\mathrm{height(m)})^2}$$

Finally, determine the category of the user based on the following table (Your function should return corresponding integers);

1=Underweight	less than 18.5
2=Healthy	from 18.5 to 25
3=Overweight	Over 25

Function Header:

```
int weight_analyzer( double weight, double height );
```

2. Write a function that takes 4 (2D) points as parameter and determines whether the points can construct a line, a triangle or a quadrilateral (Your function should return 1 for line, 2 for triangle and 3 for quadrilateral). Check for each geometric shape in a separate function.

Function Headers:

```
int geo_analyzer( double p1_x, double p1_y, double p2_x, double
p2_y, double p3_x, double p3_y, double p4_x, double p4_y);
```

```
//returns true or false (1,0)

int check_line(double p1_x, double p1_y, double p2_x, double
p2_y, double p3_x, double p3_y, double p4_x, double p4_y);

//returns true or false (1,0)

int check_triangle(double p1_x, double p1_y, double p2_x,
double p2_y, double p3_x, double p3_y, double p4_x, double
p4_y);

//returns true or false (1,0)

int check_quadrilateral (double p1_x, double p1_y, double p2_x,
double p2_y, double p3_x, double p3_y, double p4_x, double
p4_y);
```

3. Write a function that reads four unsigned digits from a text file and writes sum of second and third largest numbers to an output text file.

   Input file format (in.txt):
   4 2 1 5
   Output file format (out.txt):
   3

   Function Header:

   ```
   void sum_of_2nd_and_3nd( FILE* fptr_in, FILE* fptr_out);
   ```

4. Write an age calculator which takes birth date of a person as parameter and returns the age of the person.

   Function Header:

   ```
   double age_calculator( int year, int month, int day);
   ```

   For example your function will return 14.74 for the parameters; 2000, 01, 01 (for the current day 25/09/2014)

   Write another function that computes age of a person in the Hijri calendar. For example the age of the same person would be 15.20 in the Hijri calendar.

   Function Header:

   ```
   double age_calculator_hijri( int year, int month, int day);
   ```

   You will use `age_calculator` function in `age_calculator_hijri` function.
   Hint: Assume that a year in the Hijri calendar is 354 and in the Gregorian calendar is 365.

5. Write a function that computes and prints the following equation for *n* (m <= *n* <= m+10);

$$\sum_{n=m}^{m+10} \frac{n-1}{n^4(2n+1)}$$

Your function should get *m* as a parameter.

Function Header:

```
double compute_serie( int m );
```

Notes:

- Don't use any loop structure.
- You should submit 1 files;
  - main.c
- Add all files into a folder and compress it for submission. The folder name must be your student id.
- Upload soft copy of your homework to Moodle course web page
- Submit hard copy of your assignment to Teaching Assistant within 24 hours after the soft copy submission deadline.
- Don't forget to test your code in the provided Linux virtual machine.
- Obey good programming rules (Indentation, Documenting, Well Commenting etc.)

Your main function must be in following format;

```
int main(){

     FILE* pFile_args;
     FILE* pFile_in;
     FILE* pFile_out;

     double weight;
     double height;

     pFile_args = fopen ("args.txt","r+");
     pFile_in = fopen ("in.txt","r+");
     pFile_out = fopen ("out.txt","w+");

     ////////////////////////////////////////
     fscanf (pFile_args, "%lf", &weight);
     fscanf (pFile_args, "%lf", &height);

     int ret_val = weight_analyzer(weight, height);

     fprintf (pFile_out, "%s\n%d\n", "weight_analyzer result:", ret_val);
     ////////////////////////////////////////



     ////////////////////////////////////////
     double p1_x, p1_y;
     double p2_x, p2_y;
     double p3_x, p3_y;
     double p4_x, p4_y;

     fscanf (pFile_args, "%lf %lf", &p1_x, &p1_y);
     fscanf (pFile_args, "%lf %lf", &p2_x, &p2_y);
     fscanf (pFile_args, "%lf %lf", &p3_x, &p3_y);
     fscanf (pFile_args, "%lf %lf", &p4_x, &p4_y);

     ret_val  = geo_analyzer(p1_x, p1_y, p2_x, p2_y, p3_x, p3_y, p4_x,
     p4_y );

     fprintf (pFile_out, "%s\n%d\n","geo_analyzer result:", ret_val);
     ////////////////////////////////////////



     ////////////////////////////////////////
     fprintf (pFile_out, "%s\n","sum_of_2nd_and_3nd result:");
     sum_of_2nd_and_3nd(pFile_in, pFile_out);
     ////////////////////////////////////////



     ////////////////////////////////////////
     int year, month, day;

     fscanf (pFile_args, "%d %d %d", &year, &month, &day);

     double age_gre = age_calculator(year, month, day);
```

```
    double age_hijri = age_calculator_hijri (year, month, day);


    fprintf (pFile_out, "\n%s\n%lf\n%lf\n ", "Your age
    results(Gregorian, Hijri):", age_gre, age_hijri);
    ////////////////////////////////////////




    ////////////////////////////////////////
    int m;

    fscanf (pFile_args, "%d", &m);

    double serie_val = compute_serie(m);


    fprintf (pFile_out, "%s\n%lf\n", " compute_serie result:",
    serie_val);
    ////////////////////////////////////////



}
```

Your `args.txt` file must be in following format;

```
80 1.8
1.0 5.0
1.0 6.0
1.0 6.0
2.0 5.0
2010 1 1
10
```