```
/*****************************************************************************/
/*HW02_Onur_Sezer_121044074.c                                                */
/*_________________________                                                  */
/*Written by Onur Sezer on September 30, 2014                                */
/*                                                                           */
/*Description:                                                               */
/*____________                                                               */
/*Program dosyadan alinan degerleri  kullanip sonuclari dosyaya              */
/*yazma islemi yapar.                                                        */
/*Inputs:                                                                    */
/*args.txt ve in.txt deki degerleri alir                                     */
/*                                                                           */
/*Outputs:                                                                   */
/*out.txt ye sounuclari yazar                                                */
/*****************************************************************************/

/*---------------------------------------------------------------------------*/
/*                            Includes                                       */
/*---------------------------------------------------------------------------*/
#include<stdio.h>
#include<stdlib.h>

/*---------------------------------------------------------------------------*/
/*                          Function Prototypes                              */
/*---------------------------------------------------------------------------*/
/*************************************************/
/*Fonksiyon belirli formulle agirlik analiz eder */
/*input: double weight,height                    */
/*output:Cıkan sonucu return yapar               */
/*************************************************/
int weight_analyzer(double weight,double height);
/*****************************************************************************/
/*Fonksiyon fileden alinan noktalarin hangi sekli ifade ettigini bulur       */
/*Inputs:                                                                    */
/*double p1_x, p1_y, p2_x, p2_y, p3_x, p3_y, p4_x, p4_y                      */
/*                                                                           */
/*Outputs:                                                                   */
/*Bulunan sekli return yazar                                                 */
/*****************************************************************************/
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);
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);
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);
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);
/*********************************************************************/
/*Fonksiyon en büyük ikinci ve ücüncü sayinin toplamamini bulur  */
/*input: FILE* fptr_in  'den degerleri alir                        */
/*output:FILE* fptr_out 'a sonucu yazar                            */
/*********************************************************************/
void sum_of_2nd_and_3nd( FILE* fptr_in, FILE* fptr_out);
/*********************************************************************/
/*Fonksiyonlar Gregorian ve Hijri takvime göre yas hesaplar      */
/*input: int year,month,day                                        */
/*output:sonucu return yapar                                       */
/*********************************************************************/
double age_calculator( int year, int month, int day);
double age_calculator_hijri( int year, int month, int day);
/*******************************************/
/*Fonksiyon belirli toplam islemini yapar     */
/*input: int m                               */
/*output:sonucu return yapar                 */
/*******************************************/
double compute_serie( int m );


int main() {
    /*START_OF_MAIN*/
    double weight;
    double height;
```

```
    int ret_val ;

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

    int year, month, day;
    double age_gre;
    double age_hijri;

    int m;
    double serie_val;
    /*END_OF_VARIABLES*/

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

    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 );

    ret_val = weight_analyzer( weight, height);

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


    /*////////////////////////////////////////////////////////////////*/
    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 );
    /*////////////////////////////////////////////////////////////////*/


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

    age_gre = age_calculator(year, month, day);
    age_hijri = age_calculator_hijri (year, month, day);

    fprintf (pFile_out, "\n%s\n%.2f\n%.2f\n", "Your age results(Gregorian, Hijri):", age_gre,
age_hijri);
    /*////////////////////////////////////////////////////////////////*/


    /*////////////////////////////////////////////////////////////////*/
    fscanf (pFile_args, "%d", & m );
    serie_val = compute_serie( m );
    fprintf (pFile_out, "%s\n%f\n", "compute_serie result:",serie_val);
    /*////////////////////////////////////////////////////////////////*/

    return 0;

}
```

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

    double healthy;

    healthy=weight/(height*height);

    if(healthy<18.5)
        return 1;
    else if(healthy<25)
        return 2;
    else if(healthy>25)
        return 3;

}
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){

    if(check_line(p1_x,p1_y,p2_x,p2_y,p3_x,p3_y,p4_x,p4_y) == 1)
        return 1;
    if(check_triangle(p1_x,p1_y,p2_x,p2_y,p3_x,p3_y,p4_x,p4_y) == 1)
        return 2;
    if(check_quadrilateral(p1_x,p1_y,p2_x,p2_y,p3_x,p3_y,p4_x,p4_y) == 1)
        return 3;

}
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){
    double m;

    if( ((p2_y-p1_y)/(p2_x-p1_x)) == (m=((p3_y-p2_y)/(p3_x-p2_x))) && (m == ((p4_y-p3_y)/(p4_x-p3_x))))
        return 1;
    return 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){

    double m;

    if( ((p2_y-p1_y)/(p2_x-p1_x)) == (m=((p3_y-p2_y)/(p3_x-p2_x))) != (m == ((p4_y-p3_y)/(p4_x-p3_x))))
        return 1;

    if( ((p2_y-p1_y)/(p2_x-p1_x)) == (m=((p4_y-p2_y)/(p4_x-p2_x))) != (m == ((p4_y-p3_y)/(p4_x-p3_x))))
        return 1;

    if( ((p2_y-p1_y)/(p2_x-p1_x)) == (m=((p4_y-p3_y)/(p4_x-p4_x))) != (m == ((p3_y-p2_y)/(p3_x-p2_x))))
        return 1;

    if( ((p3_y-p2_y)/(p3_x-p2_x)) == (m=((p4_y-p2_y)/(p4_x-p2_x))) != (m == ((p2_y-p1_y)/(p2_x-p1_x))))
        return 1;

    return 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){

    double m;

    if( ((p2_y-p1_y)/(p2_x-p1_x)) != (m=((p3_y-p2_y)/(p3_x-p2_x))) && (m != ((p4_y-p3_y)/(p4_x-p3_x))))
        return 1;
    return 0;

}

void sum_of_2nd_and_3nd( FILE* fptr_in, FILE* fptr_out){

    int num1,num2,num3,num4;
    int sayi1,sayi2,sayi3,sayi4; /*Degiskenler degistiginden bu degiskenlere esitlendi*/

    int smallest,largest,result,sum;
```

```
    fscanf(fptr_in,"%d%d%d%d",&num1,&num2,&num3,&num4);
    sum=num1+num2+num3+num4;

    sayi1=num1; sayi2=num2; sayi3=num3; sayi4=num4;

    smallest=num1;

    if(num2 < smallest)
        smallest=num2;
    if(num3 < smallest)
        smallest=num3;
    if(num4 < smallest)
        smallest=num4;

    largest=sayi1;

    if(sayi2 > largest)
        largest=sayi2;
    if(sayi3 > largest)
        largest=sayi3;
    if(sayi4 > largest)
        largest=sayi4;

    result=sum-(smallest+largest);/*2. ve 3. en buyuk sayi bulunur*/

    fprintf(fptr_out,"%d",result);

}
double age_calculator( int year, int month, int day){

    /*Sistem tarihi 25/9/2014 */
    int s_year=2014 ,s_month=9 ,s_day=25 ;
    int result;
    int leap_year;

    leap_year=(s_year-year)/4;/*Subatin 29 cektigi yıllar toplami bulunur*/

    year=s_year-year;
    day=abs(day-s_day);
    month=abs(month-s_month);

    switch(month){
    case 1 :
            result=day;
            break;
    case 2 :
            result=day;
            result=result+(365-306);
            break;
    case 3 :
            result=day;
            result=result+(365-275);
            break;
    case 4 :
            result=day;
            result=result+(365-245);
            break;
    case 5 :
            result=day;
            result=result+(365-214);
            break;
    case 6 :
            result=day;
            result=result+(365-184);
            break;
    case 7 :
            result=day;
            result=result+(365-153);
            break;
    case 8 :
            result=day;
```

```
            result=result+(365-122);
            break;
    case 9 :
            result=day;
            result=result+(365-92);
            break;
    case 10 :
            result=day;
            result=result+(365-61);
            break;
    case 11 :
            result=day;
            result=result+(365-31);
            break;
    case 12 :
            result=day;
            result=365-result;
            break;
    }


    result=result+leap_year;

    return year+((result+1)/365.0);

}
double age_calculator_hijri( int year, int month, int day){
    /*Sistem tarihi 25/9/2014 */

    int s_year=2014 ,s_month=9 ,s_day=25 ;
    double age_gre,result,kalan;
    int rest;

    rest=s_year-year;

    age_gre = age_calculator(year, month, day);
    kalan=age_gre-rest;/*Yasin kusurati bulunur*/

    result=(365*rest)+(365*kalan);/*Toplam yasadigi gün sayisi bulunur*/

    result=result/354.0;/*Hijri takvime gore yasadigi yil bulunur*/
    return result;
}

double compute_serie( int m ){

    double result;


    result=(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
    m=m+1;
    result=result+(m-1.0)/(m*m*m*m*(2*m+1));
```

```
    return result;

}
/*##############################################################################*/
/*               End of HW02_Onur_Sezer_121044074                               */
/*##############################################################################*/
```