**דוח מעבדה 2**

**שמות מגישים:**

סער ויקטור – 312392822

אילון בן סימון – 312162951

**תרגיל 1**

קוד התכנית:

#include<stdio.h>

#include<math.h>

#include<conio.h>

typedef struct Point //structure definition, point type (x, y)

{

float x, y;

}Point;

typedef struct Circle//structure definition, circle type o(x, y),radius

{

Point point;

float radius;

}Circle;

/\*functions declaration\*/

int pointincircle(Point c, Circle d);

int main()

{

Point a;

Circle b;

printf("Enter the coordinates of your point:");

scanf("%f %f", &a.x, &a.y);

printf("Enter the radius and the center of your circle:");

scanf("%f %f %f", &b.radius, &b.point.x, &b.point.y);

if (pointincircle(a, b))//If the function returns one point within the circle else the point is not within the circle

printf("The point is included in the circle");

else

printf("The point is not included in the circle");

getch();

return 0;

}

int pointincircle(Point c, Circle d)//A function that checks whether the radius is greater than the straightness connecting the two points

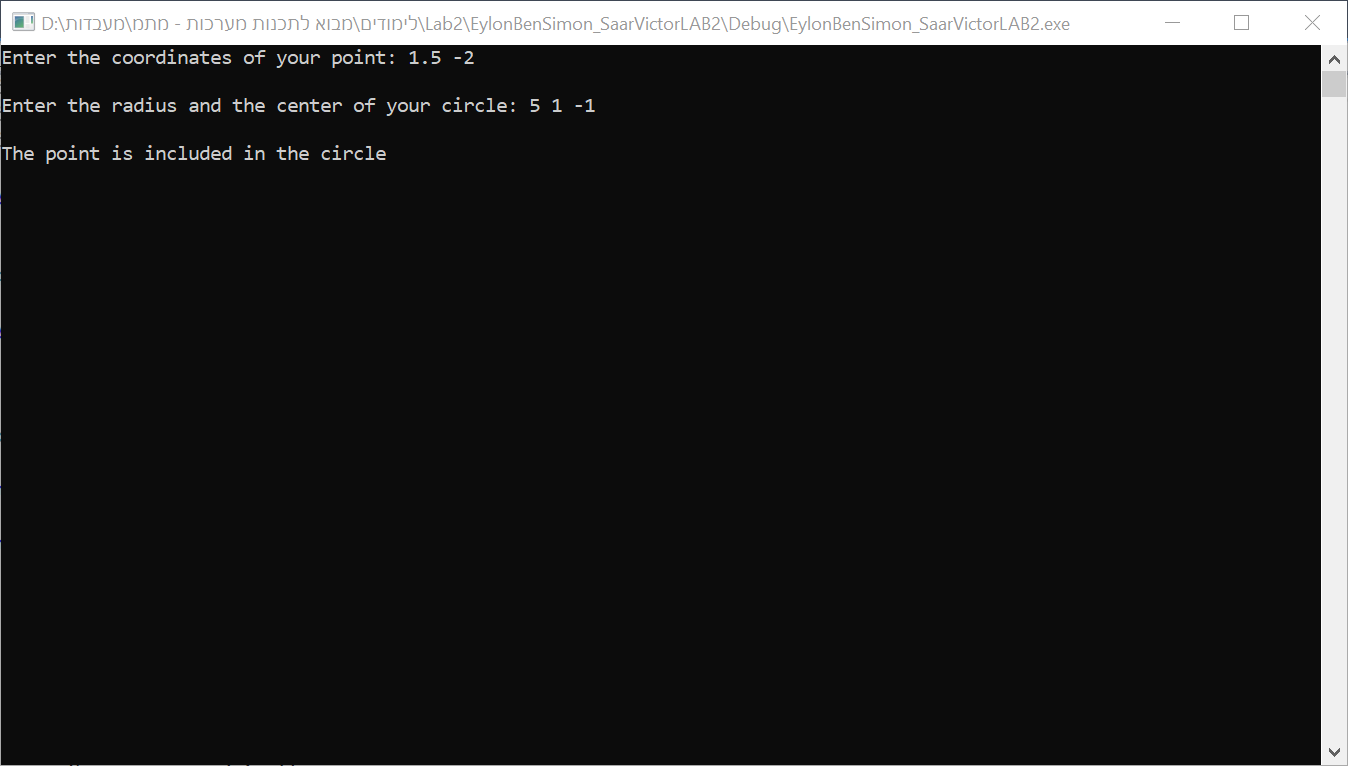
{

if (d.radius > sqrt((d.point.x - c.x)\*(d.point.x - c.x) + (d.point.y - c.y)\*(d.point.y - c.y)))

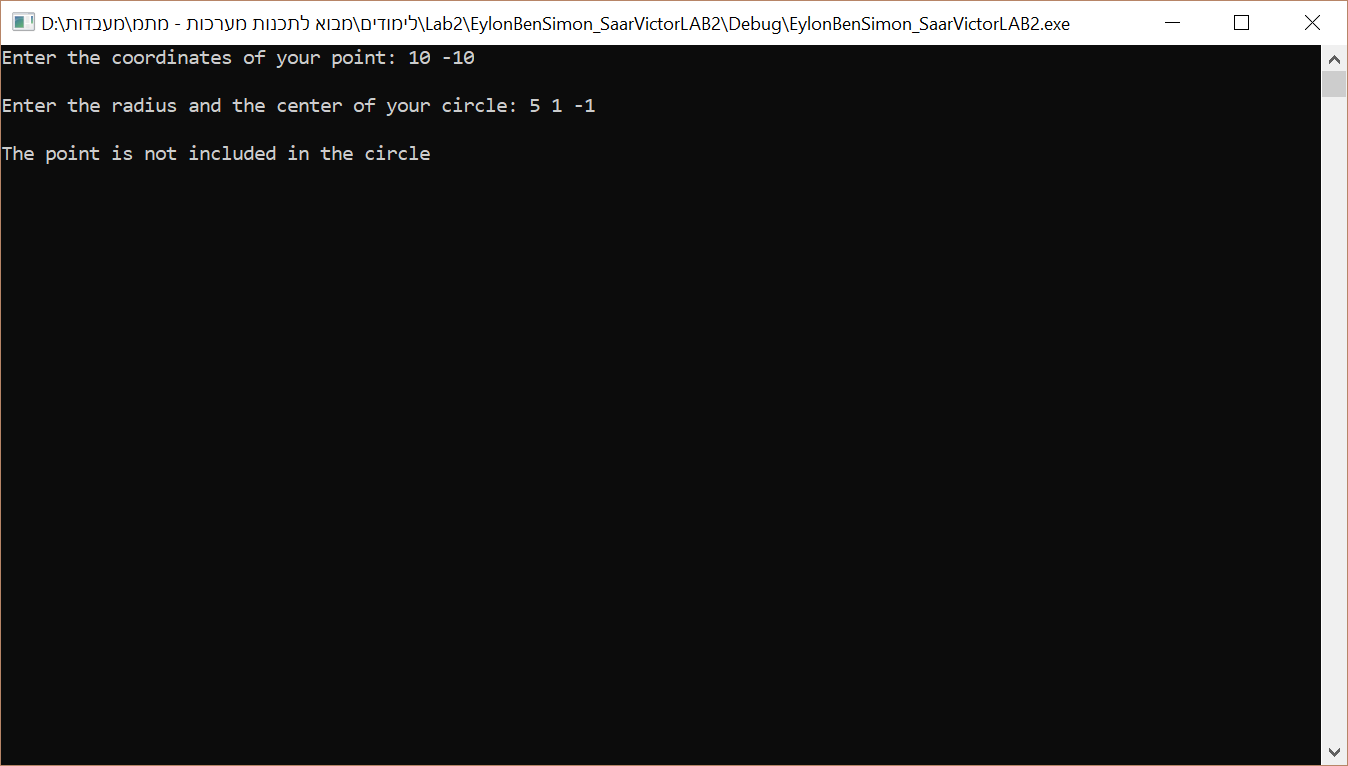
return 1;

return 0;

פלט 1 לדוגמא:



פלט 2 לדוגמא:



**תרגיל 2**

קוד התכנית:

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<conio.h>

typedef struct stud//structure definition, student type (name,mark,average)

{

char \*name;

int marks[4];

float avg;

}student;

/\*functions declaration\*/

student\* Create\_Class(int size);

void Print\_One(student \*s);

void Avg\_Mark(student \*s);

int main()

{

int size, i;

student \*arr;

printf("\nEnter the number of students:");

scanf("%d", &size);

arr=Create\_Class(size);

for (i = 0; i < size; i++)

{

if (arr[i].avg > 85)//a loop that checks whether the student average is greater than to 85 if it is sent to print else goes on

Print\_One(&arr[i]);

}

for (i = 0; i < size; i++)//loop which frees the allocation of space

free(arr[i].name);

free(arr);

getch();

return 0;

}

void Print\_One(student\* s)//a function that prints the student name and average marks

{

printf("\nThe average of %s is %.1f\n",s->name, s->avg );

}

void Avg\_Mark(student\* s)//A function that calculates the student average marks

{

int i;

float sum=0;

for (i = 0; i < 4; i++)

sum += s->marks[i];

s->avg = (sum / 4.0);

}

student\* Create\_Class(int size)//a function which inserts values into an array of structures according to the struct 'student'

{

char temp[50];

int i, j;

student\* arr;

arr = (student \*)malloc(sizeof(student)\*size);//dynamic memory allocation

if (arr == NULL)//check if the memory allocation was successful

{

printf("Error! memory not allocated.");

exit(0);

}

for (i = 0; i < size; i++)

{

printf("\nEnter your name:");

scanf("%s", temp);

arr[i].name = (char\*)malloc(strlen(temp) + 1);

if (arr[i].name == NULL)//check if the memory allocation was successful

{

printf("Error! memory not allocated.");

exit(0);

}

strcpy(arr[i].name,temp);

printf("\nEnter your marks:");

for (j = 0; j < 4; j++)

scanf("%d", &arr[i].marks[j]);

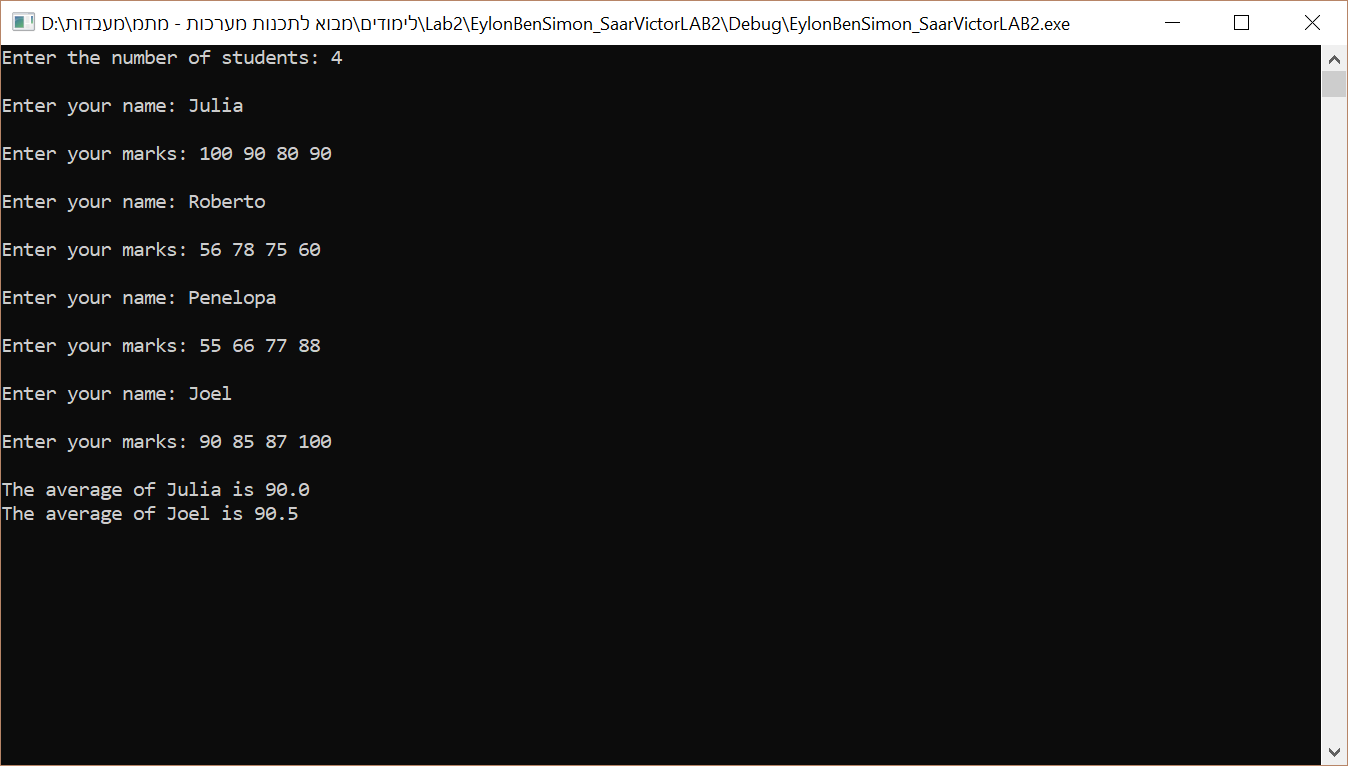
Avg\_Mark(&arr[i]);

}

return arr;

}

פלט לדוגמא:



**תרגיל 3**

קוד התכנית:

#include<stdio.h>

#include<math.h>

#include<stdlib.h>

#include<conio.h>

#define NUM 3//Quantity of complex numbers

typedef struct Complex//structure definition, complex numbers type (x+yi)

{

float real, img;

}Complex;

/\*functions declaration\*/

float CRadius(Complex a);

float\* Cmax(Complex ar[]);

int main()

{

int i;

Complex \*p;

printf("Enter 3 complex numbers:\n");

Complex \*ar;

ar = (Complex\*)malloc(sizeof(Complex)\*NUM);//dynamic memory allocation

if (ar == NULL)//check if the memory allocation was successful

{

printf("Error! memory not allocated.");

exit(0);

}

for (i = 0; i < NUM; i++)

{

scanf("%f %f", &ar[i].real, &ar[i].img);

}

p = Cmax(ar);

printf("the max complex number is %.2f+%.2fi\n", p->real, p->img);

printf("the radius of the max number is %.2f\n", CRadius(\*p));

free(ar);//free the allocation of space

getch();

return 0;

}

float CRadius( Complex a)//a function that calculates the size of the radius

{

float radius;

radius = sqrt(pow(a.real, 2) + pow(a.img, 2));

return radius;

}

float\* Cmax(Complex ar[])//a function that checks the largest complex number according to the radius

{

int i,max\_index;

float max\_radius = 0;

for (i = 0; i < NUM; i++)

{

if (CRadius(ar[i]) > max\_radius)

{

max\_radius = CRadius(ar[i]);

max\_index = i;

}

}

return &ar[max\_index];

}

פלט לדוגמא:

