**דוח מעבדה 4**

**שמות מגישים:**

סער ויקטור – 312392822

אילון בן סימון – 312162951

**תרגיל 1**

קוד התכנית:

#include<stdlib.h>

#include<stdio.h>

#include<string.h>

#include<stdio.h>

//define structures

#define MAX 256

typedef struct Book

{

char code[10];

char \*name;

}Book;

typedef struct Library

{

char name[MAX];

int num\_books;

Book \*books;

}Library;

//declertion function

void Get\_Lost(char\* str);

void input\_book(Book\* B, FILE \*in);

void input\_library(Library \*L, FILE \*in);

void output\_book(Book\* B, FILE \*out);

void output\_library(Library\* L, FILE \*out);

int main()

{

FILE \*in, \*out;

Library Libr;

int i;

in = fopen("input.txt", "rt");//open file for read "input.txt"

if (in == NULL)//check if the open was successful

{

Get\_Lost("Error faiile didn't open");//Handle case where couldn't open file.

}

input\_library(&Libr, in);

fclose(in);

out = fopen("output.txt", "wt");//open file for write "output.txt"

if (out == NULL)//check if the open was successful

{

Get\_Lost("Error faiile didn't open");//Handle case where couldn't open file.

}

output\_library(&Libr, out);

fclose(out);

for (i = 0; i < Libr.num\_books; i++)

free(Libr.books[i].name);

free(Libr.books);

getch();

return 0;

}

void Get\_Lost(char\* str)

{

printf("\n%s", str);

exit(1);

}

void input\_library(Library \*L, FILE \*in)//scan data from file"input.txt" to srtucture Library Libr

{

int i;

fscanf(in, "%s %d", L->name, &L->num\_books);

L->books = (Book\*)malloc(sizeof(Book)\*(L->num\_books));

if (L->books == NULL)

{

printf("Error! memory not Allocated");

exit(1);

}

for (i = 0; i < L->num\_books; i++)

{

input\_book(&L->books[i], in);

}

}

void input\_book(Book\* B, FILE \*in)////scan data from file"input.txt" to srtucture Book code and name of book

{

char temp[MAX];

fscanf(in, "%s %s", B->code, temp);

B->name = (char\*)malloc(strlen(temp) + 1);

if (B->name == NULL)

{

printf("Error! memory not Allocated");

exit(1);

}

strcpy(B->name, temp);

}

void output\_library(Library\* L, FILE \*out)//print from Library Libr to "output.txt" file

{

int i;

fprintf(out, "%s\n", L->name);

for (i = 0; i < L->num\_books; i++)

output\_book(&L->books[i], out);

}

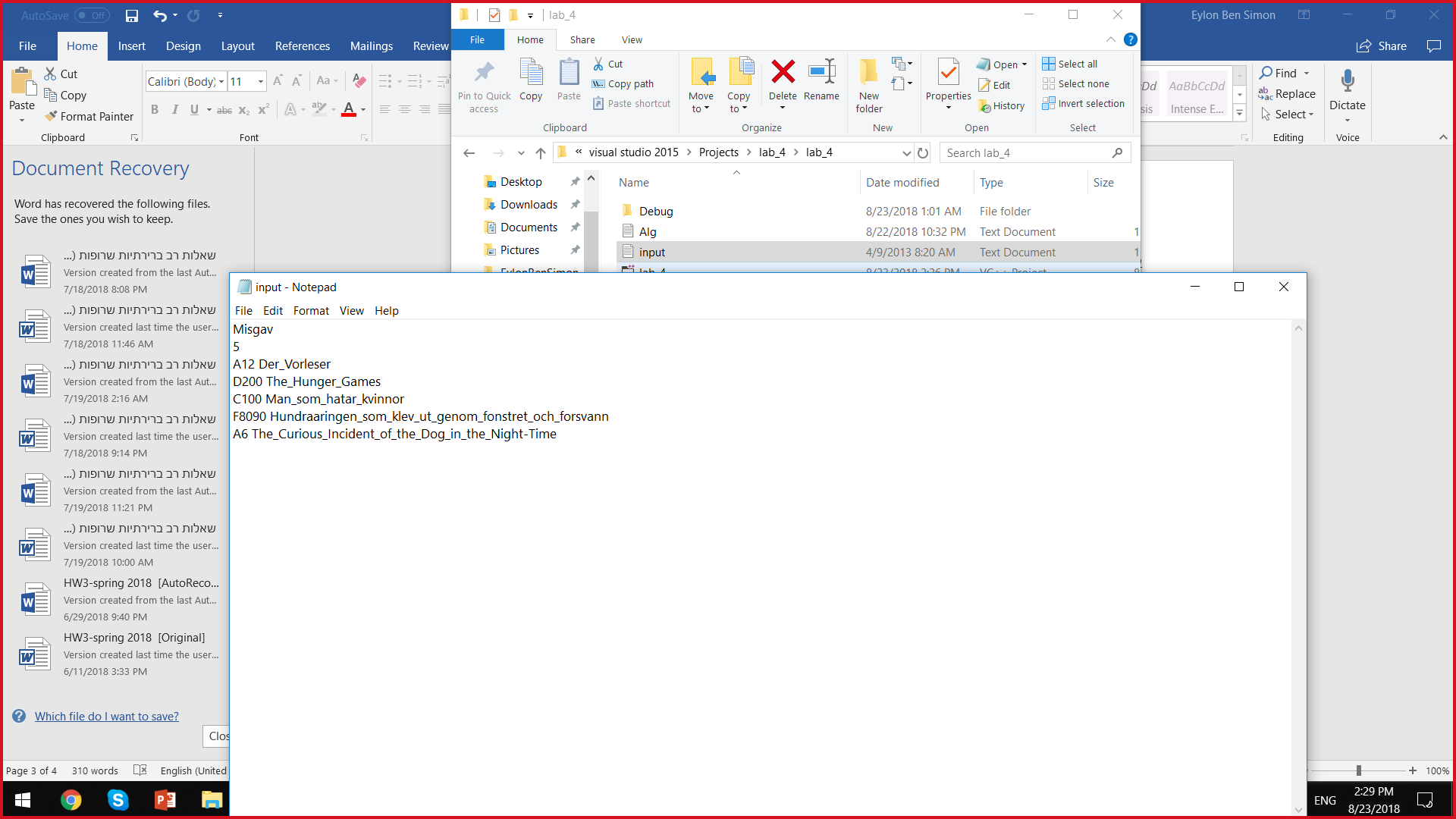
void output\_book(Book\* B, FILE \*out)

{

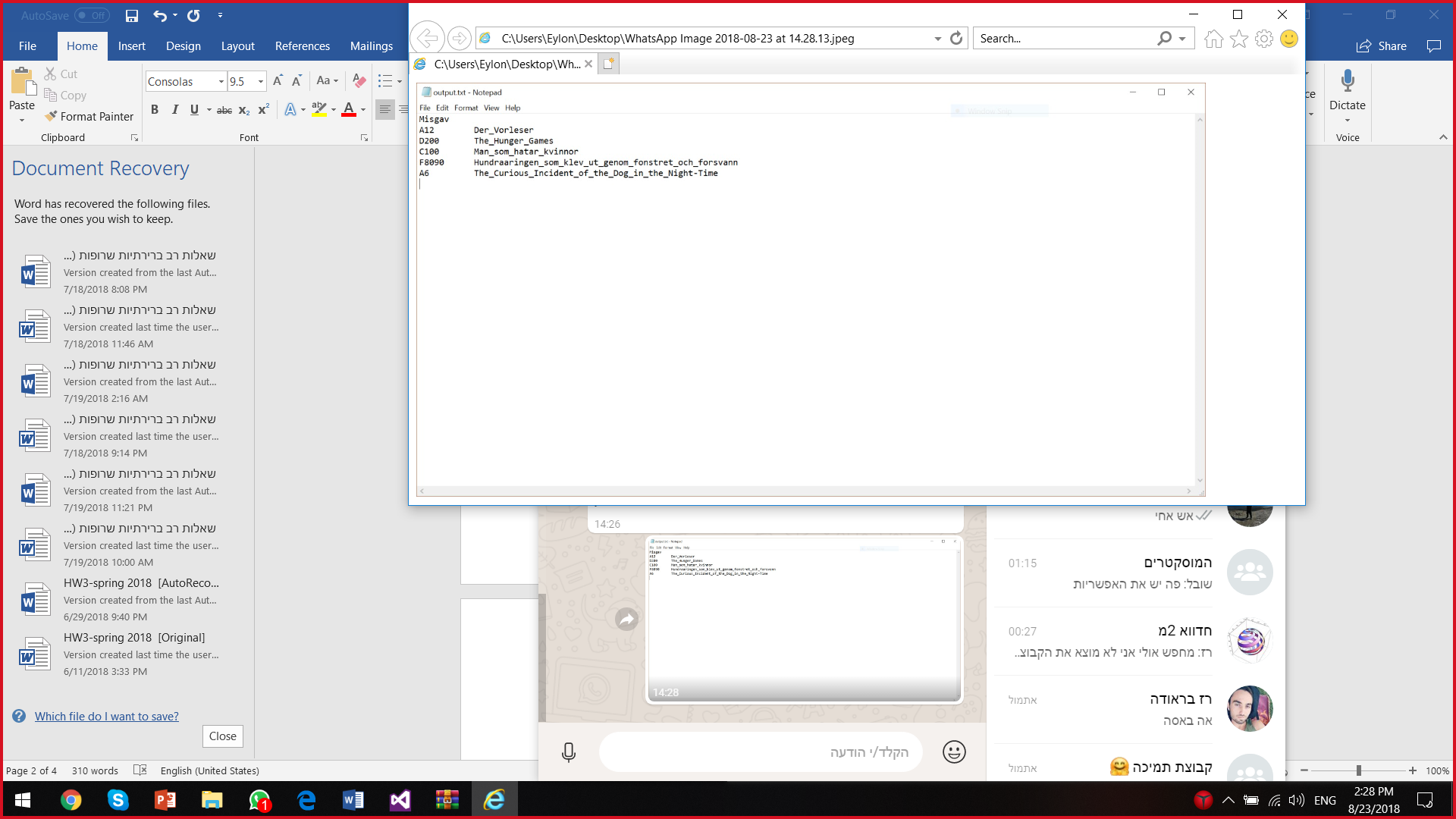
fprintf(out, "%-10s %s\n", B->code, B->name);

}

פלט (לפני שינוי):



פלט (לאחר שינוי):



**תרגיל 2**

קוד התכנית:

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <conio.h>

//define structure

typedef struct Student

{

char courseName[4];

char ID[5];

char Fullname[17];

}Student;

//declertion function

void Get\_Lost(char \*);

void Find(FILE \*file, char \*course);

int main()

{

char course[4];

FILE \*file;

file = fopen("Stud.txt", "rt");//open file for read "Stud.txt"

if (file == NULL)//check if the open was successful

Get\_Lost("Cannot open file");//Handle case where couldn't open file.

printf("\nEnter the name of course,up to 3 letters");

scanf("%s", course);

Find(file, course);

fclose(file);

getch();

return 0;

}

void Find(FILE \*file, char \*course)

{

char txtname[8],fullname[17];

Student temp;

int binaryID;

FILE \*fp;

strcpy(txtname, course);//copy the name of course to textname

strcat(txtname, ".txt");//add ".txt" to the end string textname

fp = fopen(txtname,"wt");//open a new file called textname

if (fp == NULL)//check if the open was successful

Get\_Lost("Cannot open file");//Handle case where couldn't open file.

while (!feof(file))//loop for scanning data and compering the name of course in the file to what the user printed

{

fgets(temp.courseName,4,file);

fgets(temp.ID,5,file);

fgets(temp.Fullname,17,file);

strcpy(fullname, temp.Fullname);

if (!(strcmp(temp.courseName,course)))

{

binaryID = atoi(temp.ID);

itoa(binaryID, temp.ID, 2);//converting the id to binary number

fprintf(fp, "%s, %s, %s\n", temp.ID, fullname,temp.courseName );//print the data to the file

}

}

fclose(fp);

}

void Get\_Lost(char\* str)

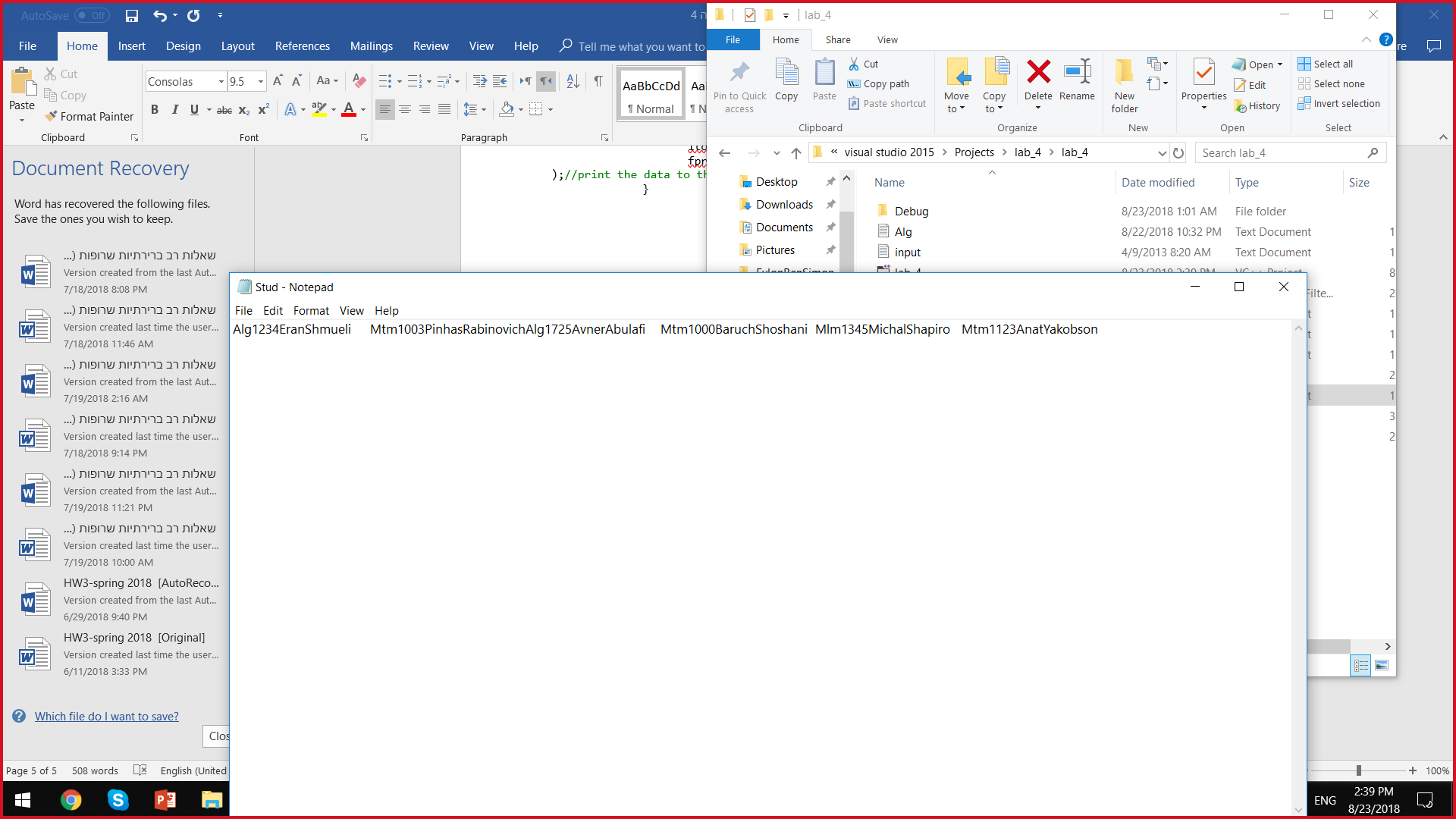
{

printf("\n%s", str);

exit(1);

}

פלט (לפני שינוי):



פלט (אחרי שינוי):

