**דוח מעבדה 10**

**שמות מגישים:**

סער ויקטור – 312392822

אילון בן סימון – 312162951

**תרגיל 1**

קוד התכנית:  
header :

#ifndef example1

#define example1

#include <stdio.h>

#include <stdlib.h>

#include<conio.h>

//PSTACKNODE is pointer to struct (node)

typedef struct node \*PSTACKNODE;

//Functions declaration

void push(void \*, PSTACKNODE \*stack);

void pop(PSTACKNODE \* stack, void(\*free\_data)(void\*));

int isempty(PSTACKNODE stack);

void\* top(PSTACKNODE stack);

#endif // !example1

Main:

#include"example1.h"

//Define specific functions

//The function free char

void free\_char(void \*elem)

{

free(elem);

}

int main()

{

PSTACKNODE stack = NULL;

char key;

char \*ch;

puts("Enter characters, separated by space. Press enter to stop\n");

do//A do while loop that scan characters until the user insert enter

{

ch = (char\*)malloc(sizeof(char));//Creates a specific address for each variable that is scan from the user

key = getchar();

\*ch = key;

\_flushall();

push(ch, &stack);

} while (key != '\n');

puts("Here are the characters in reverse order:\n");

while (!isempty(stack))//A loop that prints data from a linked list as long as the list is not empty

{

printf(" %c ", \*(char\*)top(stack));

pop(&stack,free\_char);

}

getch();

return 0;

}

Implementation:

#include"example1.h"

//Define general functions and structures

typedef struct node

{

void\* data; /\* Store the keystroke by the user \*/

struct node \*next; /\* Pointer to the next node \*/

} STACKNODE;

//The function creates a linked list

void push(void \* key, PSTACKNODE \*stack)

{

STACKNODE \*newnode;

newnode = (STACKNODE \*)malloc(sizeof(STACKNODE));

newnode->data = key;

newnode->next = (\*stack);

(\*stack) = newnode;

}

//The function return the data (void\*) to the main

void \* top(PSTACKNODE stack)

{

return stack->data;

}

//The function free node from linked list

void pop(PSTACKNODE \* stack, void(\*free\_data)(void\*))

{

STACKNODE \*oldnode;

oldnode = (\*stack);

(\*stack) = (\*stack)->next;

free\_data(oldnode->data);

free(oldnode);

}

//The function check if the linked list is empty

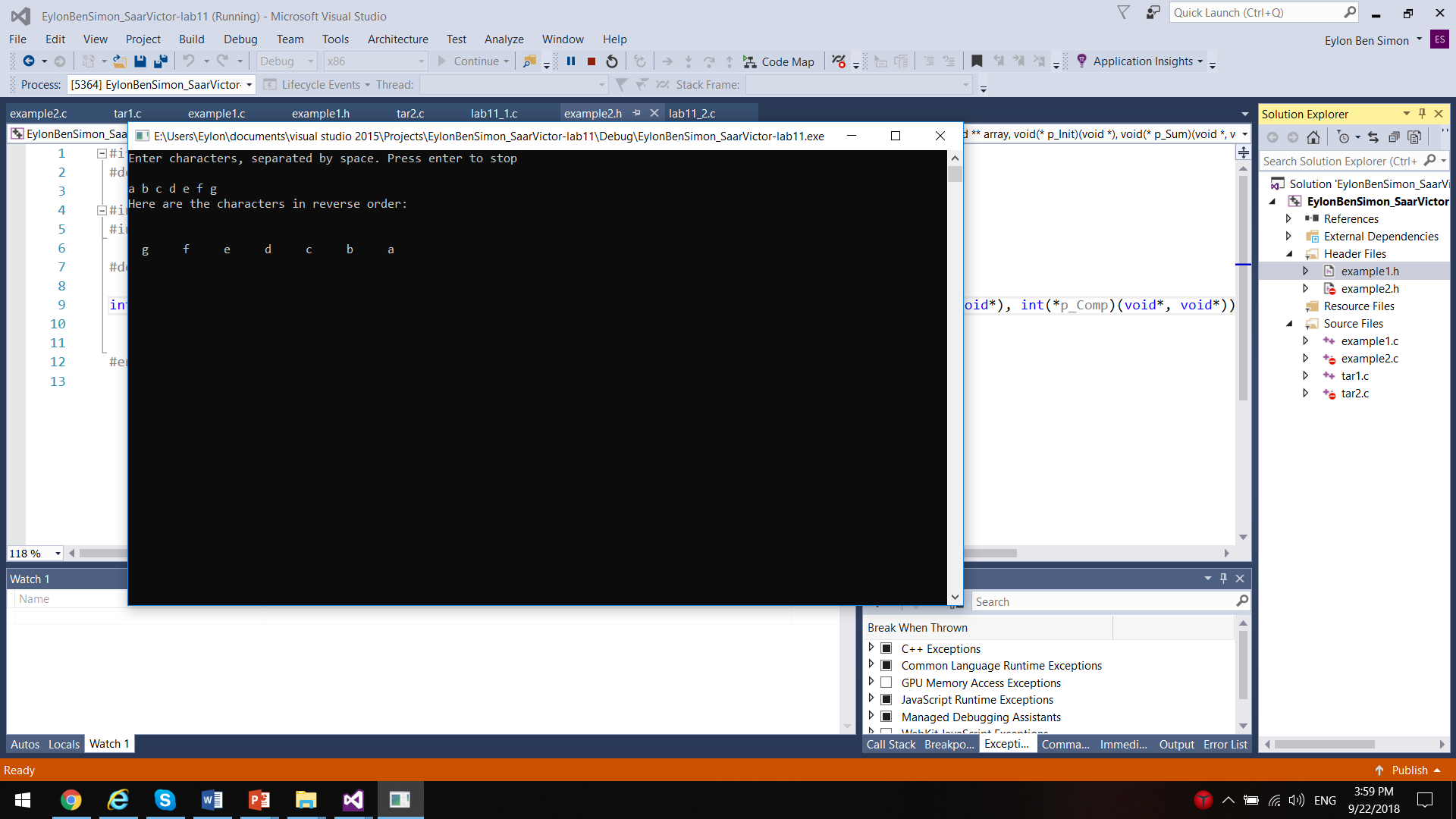
int isempty(PSTACKNODE stack)

{

return (stack == NULL);

}

פלט:



**תרגיל 2**

קוד התכנית:

header :

#ifndef example2

#define example2

#include<stdio.h>

#include<conio.h>

#define N 5

//Function declaration

int Do\_It(void \*sum, void\*\* array, void(\*p\_Init)(void \*), void(\*p\_Sum)(void\*, void\*), void(\*p\_Sub)(void\*, void\*), int(\*p\_Comp)(void\*, void\*));

#endif // !example2

Main:

#include"example2.h"

//Define specific functions

//The function compare between two numbers

int Int\_Comp(void \*a, void \*b)

{

if (\*(int\*)a == \*(int\*)b)

return 1;

return 0;

}

//The function perform sum

void Int\_Sum(void \*sum, void \*num)

{

\*(int\*)sum += \*(int\*)num;

}

//The function performs subtraction

void Int\_Sub(void \*sum, void \*num)

{

\*(int\*)sum -= \*(int\*)num;

}

//The function initializes a number

void Init(void \*p)

{

\*(int\*)p = 0;

}

int main()

{

int num[] = { 5,8,23,20,23 }, i, answer;

void \*p\_num[N];//Pointers array

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

{

p\_num[i] = &num[i];//Fill the pointers array

}

answer = Do\_It(&answer, p\_num, Init, Int\_Sum, Int\_Sub, Int\_Comp);

if (answer == 1)

printf("The answer is 'Yes'");

else

printf("The answer is 'No'");

getch();

return 0;

}

Implementation:

#include"example2.h"

//Define general functions and structures

//A function that sum numbers in the even places and subtracts numbers in the odd places and checks if the next number in the array equals the result

int Do\_It(void \*sum, void\*\* array, void(\*p\_Init)(void \*), void(\*p\_Sum)(void\*, void\*), void(\*p\_Sub)(void\*, void\*), int(\*p\_Comp)(void\*, void\*))

{

int i;

p\_Init(sum);//Initializing the variable sum

for (i = 0; i < N-1; i++)

{

if ((i+1) % 2 != 0)

p\_Sum(sum, array[i]);

else

p\_Sub(sum, array[i]);

if (p\_Comp(sum, array[i + 1]))

return 1;

}

return 0;

}

פלט:

