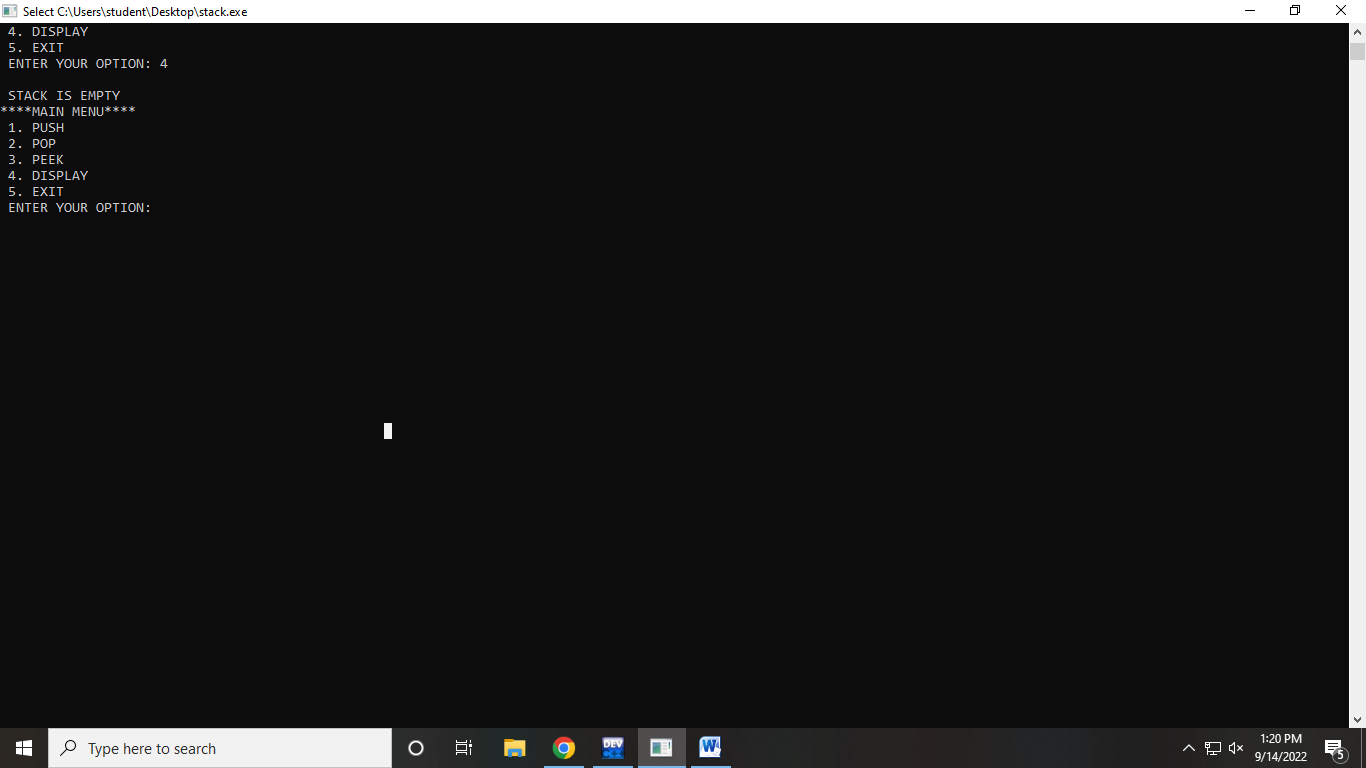
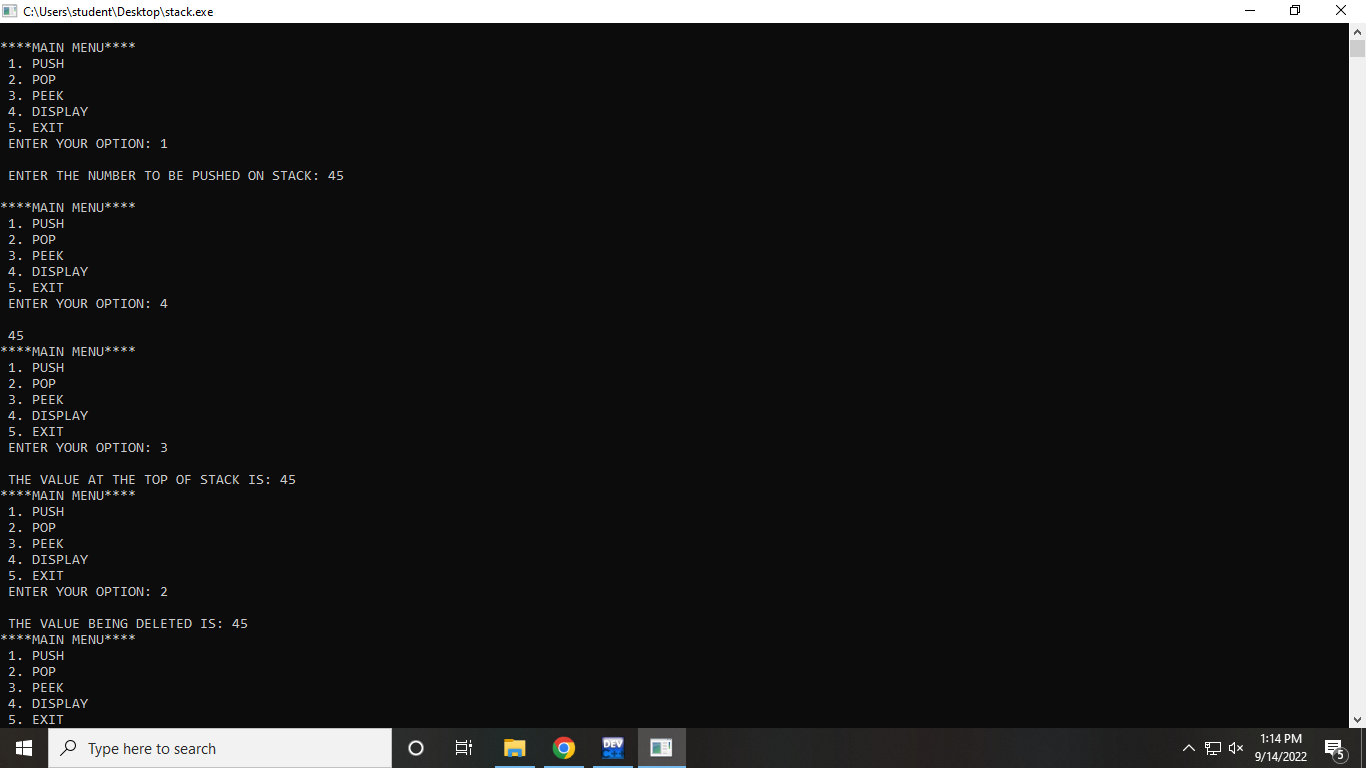
**OUTPUT**:-

****

**PROGRAM CODE:**

#include<stdio.h>

#include<stdlib.h>

#include<malloc.h>

struct stack

{

int data;

struct stack\*next;

};

struct stack\*top =NULL;

struct stack\*push(struct stack \*,int);

struct stack\*display(struct stack \*);

struct stack\*pop(struct stack \*);

int peek(struct stack\*);

int main(int argc,char\*arv[]){

int val,option;

do

{

printf("\n\*\*\*\*MAIN MENU\*\*\*\*");

printf("\n 1. PUSH");

printf("\n 2. POP");

printf("\n 3. PEEK");

printf("\n 4. DISPLAY");

printf("\n 5. EXIT");

printf("\n ENTER YOUR OPTION: ");

scanf("%d",&option);

switch(option)

{

case 1:

printf("\n ENTER THE NUMBER TO BE PUSHED ON STACK: ");

scanf("%d",&val);

top=push(top, val);

break;

case 2:

top=pop(top);

break;

case 3:

val= peek(top);

if(val !=-1)

printf("\n THE VALUE AT THE TOP OF STACK IS: %d",val);

else

printf("\n STACK IS EMPTY");

break;

case 4:

top=display(top);

break;

}

}while(option!=5);

return 0;

}

struct stack\*push(struct stack\*top,int val)

{

struct stack\*ptr;

ptr=(struct stack\*)malloc(sizeof(struct stack));

ptr ->data=val;

if(top==NULL)

{

ptr ->next =NULL;

top = ptr;

}

else

{

ptr ->next =top;

top= ptr;

}

return top;

}

struct stack \*display(struct stack\*top)

{

struct stack\*ptr;

ptr=top;

if(top==NULL)

printf("\n STACK IS EMPTY");

else

{

while(ptr!=NULL)

{

printf("\n %d",ptr->data);

ptr = ptr -> next;

}

}

return top;

}

struct stack\*pop(struct stack\*top)

{

struct stack\*ptr;

ptr= top;

if(top==NULL)

printf("\n STACK UNDERFLOW");

else

{

top=top-> next;

printf("\n THE VALUE BEING DELETED IS: %d",ptr->data);

free(ptr);

}

return top;

}

int peek(struct stack\*top)

{

if(top==NULL)

return -1;

else

return top ->data;

}