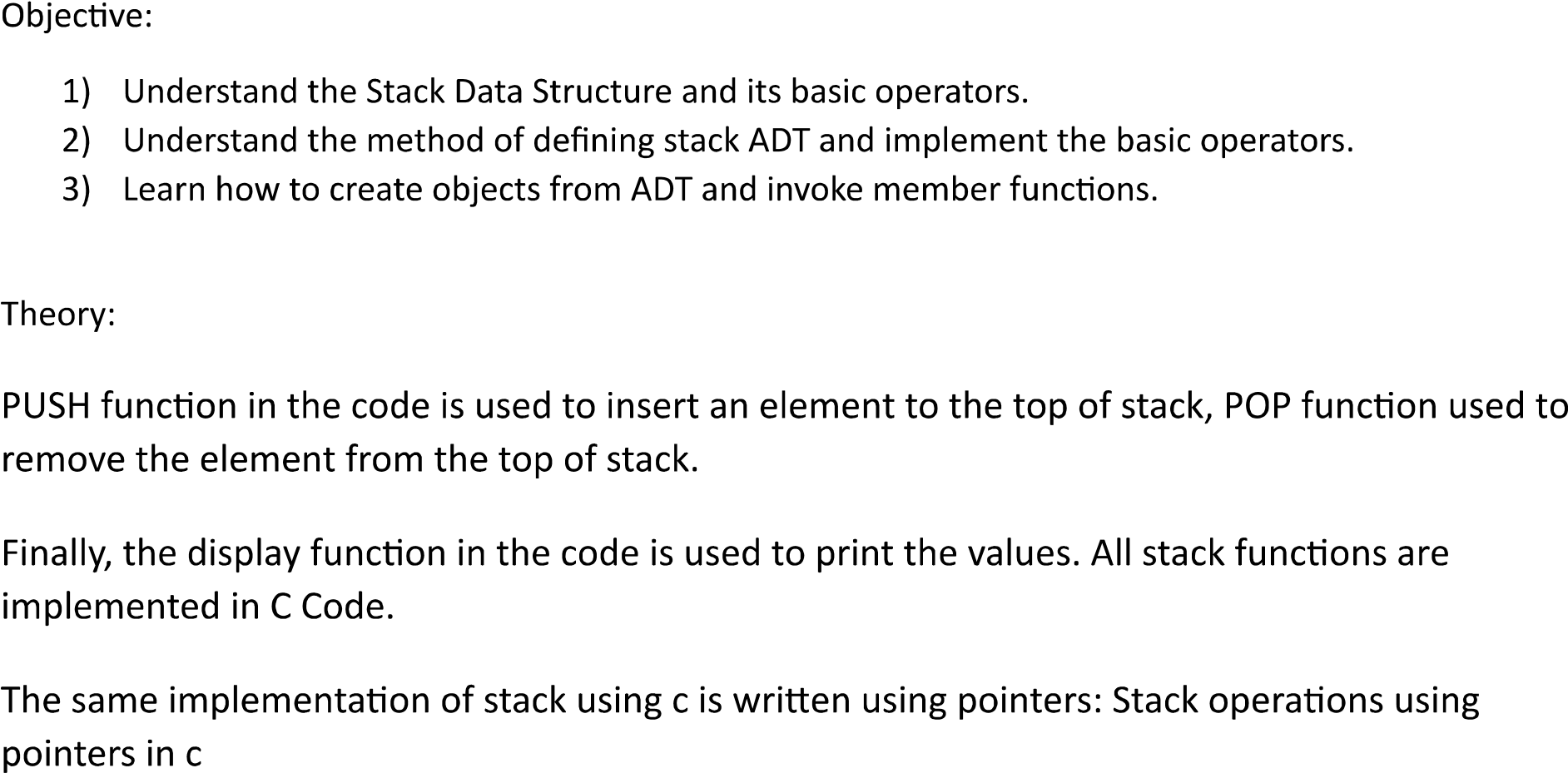
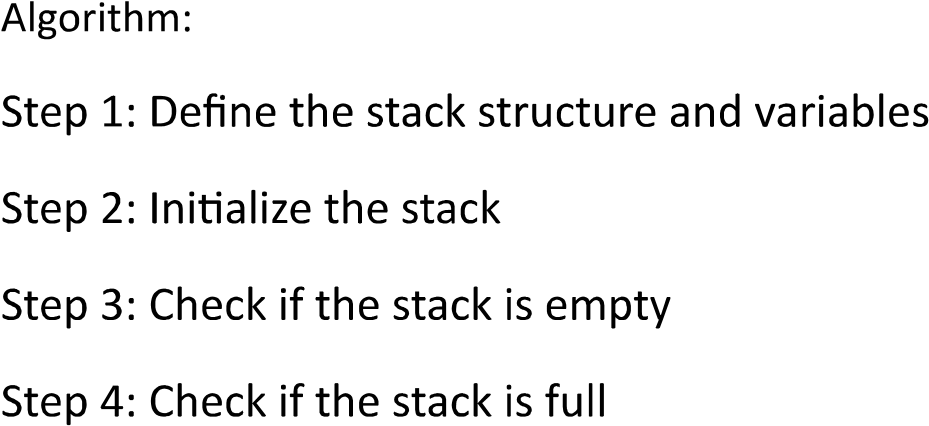
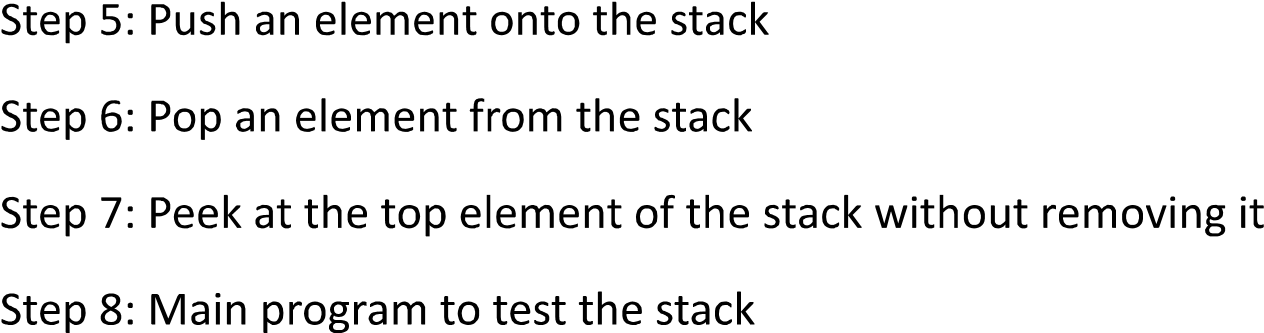
Name: Kaushal Pravin Tare , Div: C , Roll no:50

Experiment no 1: To implement stack ADT using arrays.







Code:

#include<stdio.h>

#include<conio.h>

int stack[100],top,choice,n,x,i;

void push();

void pop();

void peek();

void display();

int main()

{

top=-1;

printf("enter the size of stack:");

scanf("%d",&n);

printf("\n1.push \n2.pop \n3,peek \n4.display \n5.exit\n");

do

{

printf("enter your choice:");

scanf("%d",&choice);

switch(choice)

{

case 1:

{

push();

break;

}

case 2:

{

pop();

break;

}

case 3:

{

peek();

break;

}

case 4:

{

display();

break;

}

case 5:

{

printf("exit point\n");

break;

}

default:

{

printf("enter a valid choice\n");

}

}

}

while(choice!=5);

return 0;

}

void push(void)

{

if(top>=n-1)

{

printf("stack is overflow\n");

}

else

{

printf("enter a value to be pushed\n");

scanf("%d",&x);

top++;

stack[top]=x;

}

}

void pop(void)

{

if(top<=1)

{

printf("stack is underflow\n");

}

else

{

printf("the popped element is %d\n",stack[top]);

top--;

}

}

void peek()

{

printf("the element of the stack is %d\n",x);

}

void display()

{

if(top>=0)

{

printf("the element in the stack are \n");

for(i=top;i>=0;i--)

{

printf("%d\n",stack[i]);

}

}

else

{

printf("the stack is empty");

}

}

Output:  
