

**Stack**

|  |  |
| --- | --- |
| Name: | Wilson Vidyut Doloy |
| Reg No: | 19BCE1603 |
| Subject: | CSE2003 Data Structures and Algorithms Lab |
| Slot: | L67+L68+L69 (Prof. Sandeep Kumar) |
| Date: | 25 July 2020 |

1.

#include<stdio.h>

#include<stdbool.h>

**void** push(**int** *a*, **int** *stack***[]**, **int** \**top*, **int** *size*)

{

 if(\**top* == -1)

 {

*stack*[0] = *a*;

  \**top* = 0;

 }

 else if(\**top* == (*size*-1))

 {

  printf("The stack is already full. \n");

 }

 else

 {

  (\**top*)++;

*stack*[(\**top*)] = *a*;

 }

}

**int** pop(**int** *stack***[]**,**int** \**top*,**int** *size*)

{

 if(\**top* == -1)

 {

   printf("The stack is empty. \n");

 }

 else

 {

**int** *z*;

*z*=*stack*[(\**top*)];

    (\**top*)--;

    return *z*;

 }

}

bool isempty(**int** \**top*)

{

    if(\**top*==-1)

        return 1;

    else

        return 0;

}

**void** display(**int** *stack***[]**,**int** \**top*,**int** *size*)

 {

   if(*top*>=0)

    {

      for(**int** *i*=\**top*; *i*>=0; *i*--)

      printf("%d ",*stack*[*i*]);

   }

   else

   printf("Stack is empty");

}

**int** main()

 {

**int** *size*,*i*,*x*;

  printf("Enter the stack size:");

  scanf("%d",&*size*);

**int** *s1*[*size*];

**int** *s2*[*size*];

**int** *top1* = -1;

**int** *top2* = -1;

  printf("Enter the elements:");

  for(*i*=0;*i*<*size*;*i*++)

  {

    scanf("%d",&*x*);

    push(*x*, *s1*, &*top1*, *size*);

  }

     printf("Stack before sorting:");

    display(*s1*,&*top1*,*size*);

  while(!isempty(&*top1*))

  {

**int** *tmp* = pop(*s1*,&*top1*,*size*);

        while (!isempty(&*top2*) && *s2*[(*top2*)] < *tmp*)

        {

            push(*s2*[(*top2*)],*s1*,&*top1*,*size*);

            pop(*s2*,&*top2*,*size*);

        }

        push(*tmp*,*s2*,&*top2*,*size*);

    }

    for(*i*=0;*i*<*size*;*i*++)

    {

        push(*s2*[*i*],*s1*,&*top1*,*size*);

    }

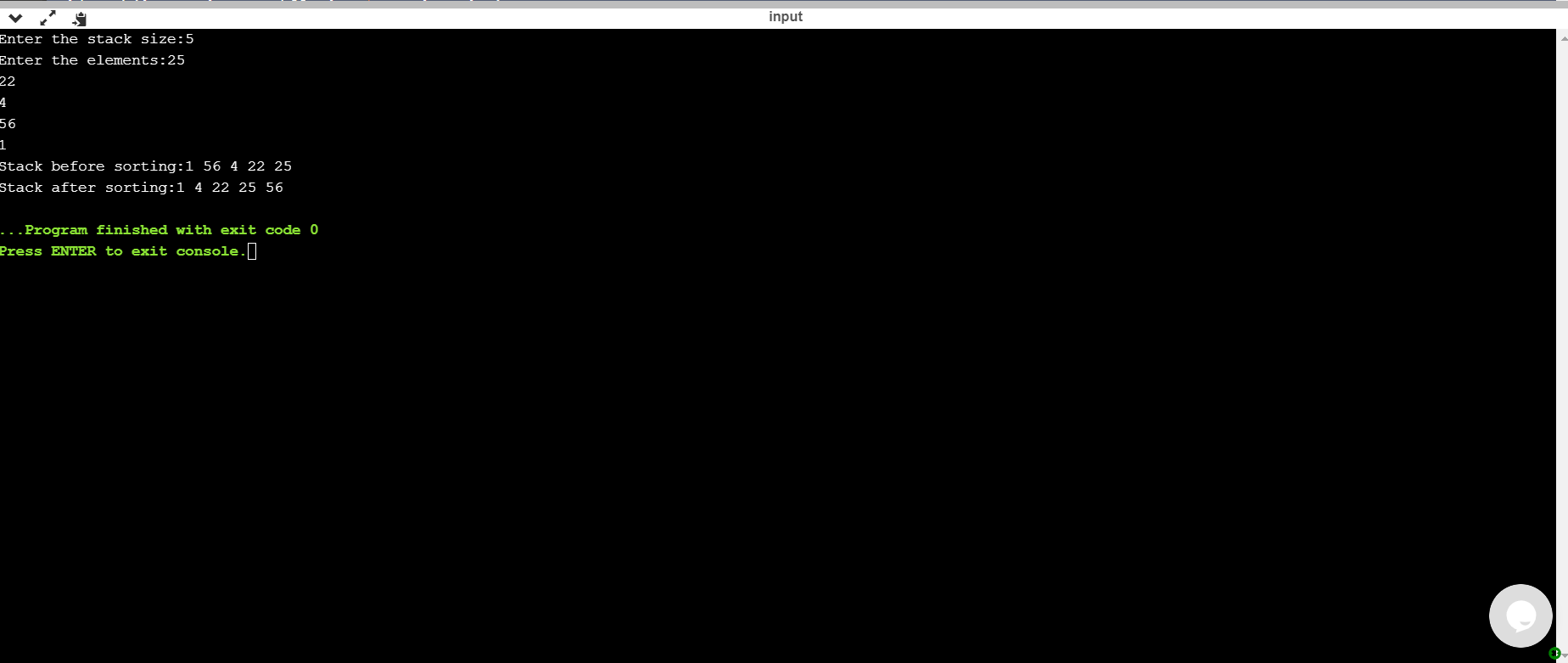
     printf("\nStack after sorting:");

     display(*s1*,&*top1*,*size*);

  return 0;

}

OUTPUT:



2.

#include<stdio.h>

#define max 5

**int** top,*a*[max];

**void** push(**void**)

{

**int** x,i;

if(top==max-1)

{

printf("stack overflow\n");

return;

}

printf("Enter Element to be inserted into Stack\n");

scanf("%d",&x);

*a*[++top]=x;

printf("Element inserted into Stack\n");

return;

}

**void** pop(**void**)

{

if(top==-1)

{

printf("stack underflow\n");

return;

}

*a*[top--]=0;

return;

}

**void** display(**void**)

{

**int** i;

if(top==-1)

{

printf("\n Stack is empty\n");

return;

}

printf("\n Elements of Stack are :\n");

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

{

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

}

return;

}

**void** popeven(**void**)

{

**int** i,t=0,*temp*[max],temptop=0;

if(top==-1)

{

printf("\n Stack is empty ! \n");

return;

}

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

{

if((i+1)%2 != 0)

{

*temp*[t]=*a*[i];

t++;

}

}

temptop=top;

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

{

pop();

}

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

{

top++;

*a*[i]=*temp*[i];

}

printf("Elements of New Stack are :\n");

top--;

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

{

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

}

return;

}

**void** main(**void**)

{

**int** c; top=-1;

do

{

printf("\n 1.Push \n 2:Pop from top(Mth element)\n 3:Pop Even positions\n 4:Display\n 5:Exit\n Choice:");

scanf("%d",&c);

switch(c)

{

case 1:push();

break;

case 2:pop();printf("Mth element deleted \n");

break;

case 3:popeven();

break;

case 4:display();

break;

case 5:printf("Program Ends\n");

break;

default :printf("Invalid Choice..\n");

break;

}

}while(c!=10);

}

OUTPUT:

