Data Structures

Experiment no. 1

Develop code to Implement Stack using Array

Q. WAP to implement Stack using arrays.

Code:

```
#include<stdio.h>
#include<conio.h>
#define MAX 5

void push(int n);
void pop();
void display();
int tos=-1,stack[MAX];

void main()
{
        int choice=1,n;

        clrscr();

        do
        {
                printf("\n\n---- STACK using Arrays ----\n\n");
                printf("\nEnter your choice:\n");
                printf("\n1. Push\n2. Pop\n3. Display\n4. EXIT\n\n");
                scanf("%d",&choice);

                switch(choice)
                {
                        case 1:
                                printf("\nEnter number to push in stack : \n");
                                scanf("%d",&n);
                                push(n);
                                display();
                                break;

                        case 2:
                                pop();
                                display();
                                break;
```

```
                        case 3:
                                display();
                                break;

                        case 4:
                                exit(0);
                                break;

                        default:
                                printf("\n**** Invalid Choice! ****\nEnter valid Choice!");
                }

        } while(choice!=4);

        getch();
}


void push(int n)
{
        if(tos==(MAX-1))
        {
                printf("\nStack is full....\n\tPush is not possible!\n");
        }
        else
        {
                tos++;
                stack[tos]=n;
        }
}

void pop()
{
        if(tos==-1)
        {
                printf("\nStack is empty....\n\tCannot pop from stack!\n");
        }
        else
        {
                printf("\nPopped element is %d.\n",stack[tos]);
                tos--;
        }
}

void display()
{
        int i;
```

```
        if(tos==-1)
        {
                printf("\nStack is empty....\n\tCannot display stack!\n");
        }
        else
        {
                printf("\nCurrent Stack is :\n");
                for(i=tos;i>-1;i--)
                {
                        printf("\t|\t%d\t|\n",stack[i]);
                }
        }
}
```

Output:

```
---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

1

Enter number to push in stack :
200

Current Stack is :
      |     200   |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

1

Enter number to push in stack :
10
```

```
Current Stack is :
      |     10    |
      |     200   |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

1

Enter number to push in stack :
100

Current Stack is :
      |     100   |
      |     10    |
      |     200   |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

1

Enter number to push in stack :
50

Current Stack is :
      |     50    |
      |     100   |
      |     10    |
      |     200   |


---- STACK using Arrays ----
```

```
Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

1

Enter number to push in stack :
60

Current Stack is :
      |     60    |
      |     50    |
      |     100   |
      |     10    |
      |     200   |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

1

Enter number to push in stack :
1000

Stack is full....
      Push is not possible!

Current Stack is :
      |     60    |
      |     50    |
      |     100   |
      |     10    |
      |     200   |


---- STACK using Arrays ----


Enter your choice:
```

```
1. Push
2. Pop
3. Display
4. EXIT

2

Popped element is 60.

Current Stack is :
      |    50   |
      |    100  |
      |    10   |
      |    200  |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

2

Popped element is 50.

Current Stack is :
      |    100  |
      |    10   |
      |    200  |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

1

Enter number to push in stack :
10000
```

```
Current Stack is :
      |     10000|
      |     100  |
      |     10   |
      |     200  |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

3

Current Stack is :
      |     10000|
      |     100  |
      |     10   |
      |     200  |


---- STACK using Arrays ----


Enter your choice:

1. Push
2. Pop
3. Display
4. EXIT

4
```

Q. WAP to reverse a string using Stack.

Code:

```
#include<stdio.h>
#include<string.h>
#include<conio.h>
#define MAX 100

void push(char ch);
char pop();
int tos=-1;
char stack[MAX];

void main()
{
   int i;
   char str[100],ch,str_rev[100];

   clrscr();

   printf("\nEnter a string: \n");
   fflush(stdin);
   gets(str);

   for(i=0;i<strlen(str);i++)
   {
      push(str[i]);
   }

   for(i=0;i<strlen(str);i++)
   {
      str_rev[i]=pop();
   }

   printf("\nString after reverse is : %s",str_rev);

   getch();
}


void push(char ch)
{
   if(tos==(MAX-1))
   {
      printf("\nStack Overflow!\n");
   }
   else
```

```
   {
      tos++;
      stack[tos]=ch;
   }
}

char pop()
{
   if(tos==-1)
   {
      printf("\nStack Underflow!\n");
      return 0;
   }
   else
   {
      tos--;
      return stack[tos+1];
   }
}
```

Output:

```
Enter a string:
hello

String after reverse is : olleh
```

Q. WAP to convert Decimal to Binary using Stack.

Code:

```
#include<stdio.h>
#include<conio.h>
#define MAX 100

void push(int n);
void pop();
int tos=-1;
int stack[MAX];

void main()
{
   int num,digit=0,i;
   clrscr();

   printf("\n---- Decimal to Binary ----\n");
   printf("\nEnter a number to convert :\n");
   scanf("%d",&num);

   while(num>=1)
   {
      push(num%2);
      num=num/2;
      digit++;
   }

   printf("\nThe number in Binary form is :\t");
   for(i=0;i<digit;i++)
   {
      pop();
   }

         getch();
}


void push(int n)
{
   if(tos==(MAX-1))
   {
      printf("\nStack Overflow!\n");
   }
   else
   {
      tos++;
```

```
      stack[tos]=n;
   }
}

void pop()
{
   if(tos==-1)
   {
      printf("\nStack Underflow!\n");
   }
   else
   {
      printf("%d",stack[tos]);
      tos--;
   }
}
```

Output:

```
---- Decimal to Binary ----

Enter a number to convert :
12

The number in Binary form is :   1100
```