

Stack

Stack Program

Write a C program to implement a stack using an array.

OR

Write a C program that performs basic operations on a stack using an array.

OR

Write a C program for stack with the use of an array.

OR

Write a menu driven C program for the stack with the use of array.

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>

#define MAXSIZE 5

int stack[MAXSIZE], top = -1;
void push();
void pop();
void peep();
void update();
void display();

void main()
{
    int choice = 0;
    do
    {
        clrscr();
        printf("\n Main Menu (Basic Operations on Stack)");
        printf("\n 1. Push a New Item at the Top of the Stack");
        printf("\n 2. Pop an Item from the Top of the Stack");
        printf("\n 3. Peep a Specified Item in the Stack");
        printf("\n 4. Update (change) a Specified Item in the Stack");
        printf("\n 5. Display Items in the Stack");
        printf("\n 6. Exit");
        printf("\n Enter your choice (from 1 to 6): ");
        scanf("%d", &choice);
        switch(choice)
        {
            case 1:
                push();
                break;
```

```
        case 2:
            pop();
            break;
        case 3:
            peep();
            break;
        case 4:
            update();
            break;
        case 5:
            display();
            break;
        case 6:
            exit(0);
            break;

        default:
            printf("\n Invalid choice");
    }
    printf("\n Press any key to continue...");
    getch();
} while(choice != 6);
}

void push()
{
    int item = 0;
    if(top == MAXSIZE-1)
    {
        printf("\n Stack is full (stack overflow).");
    }
    else
    {
        printf("\n Enter the element to be pushed at TOS: ");
        scanf("%d", &item);
        top = top + 1;
        stack[top] = item;
    }
}

void pop()
{
    int item = 0;
    if(top == -1)
    {
        printf("\n Stack is empty (stack underflow).");
    }
    else
    {
        item = stack[top];
        top = top - 1;
    }
}
```

```
        printf("\n The item popped is %d.", item);
    }
}

void peep()
{
    int item = 0, pos = 0;
    if(top == -1)
    {
        printf("\n Stack is empty (stack underflow).");
    }
    else
    {
        printf("\n Enter the position to read the element: " );
        scanf("%d", &pos);
        if(pos <= 0 || pos > top+1)
        {
            printf("\n Position out of range.");
        }
        else
        {
            item = stack[top-pos+1];
            printf("\n The peeped item is %d.", item);
        }
    }
}

void update()
{
    int item = 0, pos = 0, value = 0;
    if(top == -1)
    {
        printf("\n Stack is empty (stack underflow).");
    }
    else
    {
        printf("\n Enter the position to update the element: " );
        scanf("%d", &pos);
        if(pos <= 0 || pos > top+1)
        {
            printf("\n Position out of range.");
        }
        else
        {
            item = stack[top-pos+1];
            printf("\n Enter the new value: " );
            scanf("%d", &value);
            stack[top-pos+1] = value;
            printf("\n The updated (changed) item is %d.", item);
        }
    }
}
```

```
    }  
}  
  
void display()  
{  
    int i = 0;  
    if(top == -1)  
    {  
        printf("\n Stack is empty (stack underflow).");  
    }  
    else  
    {  
        printf("\n Elements or items in the stack are: ");  
        for(i = top; i >= 0; i--)  
        {  
            printf("\n %d", stack[i]);  
        }  
    }  
}
```