```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define MAX_ELE 4

void status(int *s, int top) {
    int used;
    if (top == -1)
        used = 0;
    else
        used = top + 1;

    printf("%d locations of the stack are used up\n", used);
    printf("%d locations of the stack are free\n", MAX_ELE - used);
}

void display(int s[], int *top) {
    int i;
    if ((*top) == -1)
        printf("stack empty\n");
    else {
        printf("stack elements are\n");
        printf("TOS is: ");
        for(i = (*top); i >= 0; i--)
            printf("%d\n", s[i]);
    }
}

void push(int *s, int *top, int ele) {
    if ((*top) == MAX_ELE - 1) {
        printf("stack overflow\n");
        return;
    }
    (*top)++;
    printf("enter the element: ");
    scanf("%d", &ele);
    s[*top] = ele;
}

void pop(int s[], int *top) {
    if ((*top) == -1)
        printf("stack underflow\n");
    else {
        printf("element popped is: %d\n", s[*top]);
        (*top)--;
    }
}

void palindrome(char *s) {
    int length = strlen(s);
    int stack[MAX_ELE];
    int top = -1;
```

```
54
55      // Push characters into the stack
56      for (int i = 0; i < length; i++)
57          push(stack, &top, s[i]);
58
59      // Check if the string is a palindrome
60      for (int i = 0; i < length; i++) {
61          if (stack[top--] != s[i]) {
62              printf("String is not a palindrome\n");
63              exit(0);
64          }
65      }
66      printf("String is a palindrome\n");
67  }
68
69  int main() {
70      int ch, top = -1;
71      int s[MAX_ELE];
72      char str[10] = "mom";
73      for (;;) {
74          printf("1: push\n2: pop\n3: palindrome\n4: display\n5: Status\n6:
    Exit\n");
75          printf("Enter choice: ");
76          scanf("%d", &ch);
77          switch(ch) {
78              case 1: push(s, &top, 0); break;  // '0' will just act as a
    placeholder for now
79              case 2: pop(s, &top); break;
80              case 3: palindrome(str); break;
81              case 4: display(s, &top); break;
82              case 5: status(s, top); break;
83              default: exit(0);
84          }
85      }
86  }
```