```c
#include <stdio.h>
#include <ctype.h>  // for isdigit
#include <stdlib.h> // for atoi

#define SIZE 100

int stack[SIZE];
int top = -1;

// Push an element to the stack
void push(int value) {
    if (top >= SIZE - 1) {
        printf("Stack Overflow\n");
        return;
    }
    stack[++top] = value;
}

// Pop an element from the stack
int pop() {
    if (top == -1) {
        printf("Stack Underflow\n");
        return -1;
    }
    return stack[top--];
}

// Evaluate the postfix expression
int evaluatePostfix(char* expr) {
    int i;
    for (i = 0; expr[i] != '\0'; i++) {
        char ch = expr[i];

        // If character is a space, skip it
        if (ch == ' ')
            continue;

        // If the character is a digit
        else if (isdigit(ch)) {
            // Convert char to int and push
            int num = ch - '0';
            push(num);
        }

        // If the character is an operator
        else {
            int val2 = pop();
            int val1 = pop();
```

```
            switch (ch) {
                case '+': push(val1 + val2); break;
                case '-': push(val1 - val2); break;
                case '*': push(val1 * val2); break;
                case '/': push(val1 / val2); break;
                default:
                    printf("Unsupported operator: %c\n", ch);
                    return -1;
            }
        }
    }
    return pop();
}

int main() {
    char expr[SIZE];

    printf("Enter a postfix expression (e.g. 23*54*+9-): ");
    scanf("%s", expr);

    int result = evaluatePostfix(expr);
    printf("Result = %d\n", result);

    return 0;
}
```