

```

#include <stdio.h>
#include <string.h>
char stack[100];
int top = -1;
void push(char c) {
    stack[++top] = c;
}
char pop() {
    if (top == -1) return '\0';
    return stack[top--];
}
int isMatching(char open, char close) {
    if (open == '(' && close == ')') return 1;
    if (open == '{' && close == '}') return 1;
    if (open == '[' && close == ']') return 1;
    return 0;
}
int isValid(char expr[]) {
    for (int i = 0; i < strlen(expr); i++) {
        char ch = expr[i];
        if (ch == '(' || ch == '{' || ch == '[') {
            push(ch);
        } else if (ch == ')' || ch == '}' || ch == ']') {
            char open = pop();
            if (!isMatching(open, ch)) return 0;
        }
    }
    return top == -1;
}
int main() {
    char expr[100];
    printf("Enter bracket expression: ");
    scanf("%s", expr);
    if (isValid(expr)) {
        printf("Valid expression\n");
    } else {
        printf("Invalid expression\n");
    }
    return 0;
}

```

C:\Users\upper\OneDrive\DATA STRUCTRES\Valid stack .exe

Enter bracket expression: {}
Valid expression

Process exited after 21.6 seconds with return value 0
Press any key to continue . . .