

DSA LAB – 6

Name: Etcherla Sai Manoj

Mis. No: 112015044

Branch: CSE

Question 1:

Code:

```
#include<iostream>
#include<string.h>
#define SIZE 30
using namespace std;

char stack[SIZE];
int top = -1;
//function prototype declaration
void push(char);
void pop();

//main function
int main(){
    char expr[SIZE];
    int flag = 0;
    //input expression
    cout << "\nEnter the expression : ";
    cin >> expr;
    cout << "\n-----ERRORS-----\n";
    //check open braces and push to stack
    for(int i = 0; i < strlen(expr); i++){
        if(expr[i] == '(' || expr[i] == '{' || expr[i] == '['){
            push(expr[i]);
        }
        //check for matching open braces based on close braces
        if(expr[i] == '){
            if(stack[top] == '('){
                pop();
            }
            else{
                cout << "opening brace '(' is not matched\n";
                flag = 1;
            }
        }
        else if(expr[i] == '){
            if(stack[top] == '{'){
                pop();
            }
            else{
                cout << "opening brace '{' is not matched\n";
                flag = 1;
            }
        }
    }
}
```

```

else if(expr[i] == ' '){
    if(stack[top] == '['){
        pop();
    }
    else{
        cout << "opening brace '[' is not matched\n";
        flag = 1;
    }
}
}
//stack is empty after pop and expression balanced
if(top == -1 && flag != 1){
    cout << "-----\n\n";
    cout << "Status : Balanced\n";
    cout << "This is a valid expression\n\n";
}
//check for remaining braces in stack(if any)
else{
    while(top != -1){
        if(stack[top] == '('){
            pop();
            cout<<"closing brace ')' is not matched\n";
        }
        else if(stack[top] == '{'){
            pop();
            cout<<"closing brace '}' is not matched\n";
        }
        else if(stack[top] == '['){
            pop();
            cout<<"closing brace ']' is not matched\n";
        }
    }
    cout << "-----\n\n";
    cout << "Status : NOT BALANCED\n";
    cout << "This is an invalid expression\n\n";
}
return 0;
}

//push function for stack
void push(char c){
    if(top == SIZE){
        cout << "Stack Overflown\n";
    }
    else{
        top = top + 1;
        stack[top] = c;
    }
}
//end of push

```

```
//pop function for stack
void pop(){
    if(top == -1){
        cout << "Stack Underflown\n";
    }
    else{
        top--;
    }
}
//end of pop
```

Input & Output:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6\" ; if ($?) { g++ parentheses.cpp -o parentheses } ; if ($?) {
.\parenthesiss }

Enter the expression : (a+b)

-----ERRORS-----

Status : Balanced
This is a valid expression

PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6> █
```

```
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6\" ; if ($?) { g++ parentheses.cpp -o parentheses
} ; if ($?) { .\parenthesiss }

Enter the expression : {a+b)

-----ERRORS-----
opening brace '(' is not matched
closing brace ')' is not matched
-----

Status : NOT BALANCED
This is an invalid expression

PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6> █
```

```
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6\" ; if ($?) { g++ parentheses.cpp -o parentheses
} ; if ($?) { .\parenthesiss }

Enter the expression : {(a+b)[c*d]}

-----ERRORS-----

Status : Balanced
This is a valid expression

PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6> █
```

```
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6\" ; if ($?) { g++ parentheses.cpp -o parentheses
} ; if ($?) { .\parenthesiss }

Enter the expression : {(a+b)}[a*b][

-----ERRORS-----
opening brace '{' is not matched
opening brace '{' is not matched
closing brace ']' is not matched
closing brace ']' is not matched
closing brace '}' is not matched
closing brace '}' is not matched
-----

Status : NOT BALANCED
This is an invalid expression

PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 6> █
```