ASSIGNMENT 3 DATA STRUCTURE

Q1. Develop a menu driven program demonstrating the following operations on a Stack using array:

(i) push(), (ii) pop(), (iii) isEmpty(), (iv) isFull(), (v) display(), and (vi) peek().

Code :

Q1.cpp > main()

```cpp
// Develop a menu driven program demonstrating the following operations on a Stack using array:
//(i) push(), (ii) pop(), (iii) isEmpty(), (iv) isFull(), (v) display(), and (vi) peek().
#include <iostream>
using namespace std;
#define MAX 10
int stack[MAX];
int top=-1;

void push(){
    if(top==MAX-1)
        cout<<"Stack is Full\n";
    else{
        int x;
        cout<<"Enter element: ";
        cin>>x;
        stack[++top]=x;
    }
}

void pop(){
    if(top==-1)
        cout<<"Stack is Empty\n";
    else{
        cout<<"Popped element: "<<stack[top]<<endl;
        top--;
    }
}

void isEmpty(){
    if(top==-1) cout<<"Stack is Empty\n";
    else cout<<"Stack is not Empty\n";
}

void isFull(){
    if(top==MAX-1) cout<<"Stack is Full\n";
    else cout<<"Stack is not Full\n";
}
```

```
38
39  void display(){
40      if(top==-1) cout<<"Stack is Empty\n";
41      else{
42          cout<<"Stack elements: ";
43          for(int i=top;i>=0;i--)
44              cout<<stack[i]<<" ";
45          cout<<endl;
46      }
47  }
48
49  void peek(){
50      if(top==-1) cout<<"Stack is Empty\n";
51      else cout<<"Top element: "<<stack[top]<<endl;
52  }
53
54  int main(){
55      int ch;
56      while(1){
57          cout<<"\n1.Push\n2.Pop\n3.isEmpty\n4.isFull\n5.Display\n6.Peek\n7.Exit\n";
58          cout<<"Enter choice: ";
59          cin>>ch;
60          switch(ch){
61              case 1: push(); break;
62              case 2: pop(); break;
63              case 3: isEmpty(); break;
64              case 4: isFull(); break;
65              case 5: display(); break;
66              case 6: peek(); break;
67              case 7: return 0;
68              default: cout<<"Invalid choice\n";
69          }
70      }
71  }
72
```

Output:

```
Enter element: 445

1.Push
2.Pop
3.isEmpty
4.isFull
5.Display
6.Peek
7.Exit
Enter choice: 1
Enter element: 55

1.Push
2.Pop
3.isEmpty
4.isFull
5.Display
6.Peek
7.Exit
Enter choice: 2
Popped element: 55

1.Push
2.Pop
3.isEmpty
4.isFull
5.Display
6.Peek
7.Exit
Enter choice: 5
Stack elements: 445

1.Push
2.Pop
3.isEmpty
4.isFull
5.Display
6.Peek
7.Exit
Enter choice: 7
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
```

Q2. Given a string, reverse it using STACK. For example “DataStructure” should be output as “erutcurtSataD.”

Code with output :

Q1.cpp | Q2.cpp | Q2.exe

Q2.cpp > main()

```cpp
//2. Given a string, reverse it using STACK. For example “DataStructure” should be output as “erutcurtSataD.”
#include <iostream>
#include <string.h>
using namespace std;
int main(){
    char str[100];
    char stack[100];
    int top=-1;
    cout<<"Enter string: ";
    cin>>str;
    int n=strlen(str);
    for(int i=0;i<n;i++) stack[++top]=str[i];
    cout<<"Reversed string: ";
    while(top!=-1){
        cout<<stack[top--];
    }
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>  & 'c:\Users\hxrle\.vscode\extensions\ms-vscode.cpptools-1.28.3-win32-x64\debugAdapters\bin\WindowsDebugLaun
cher.exe' '--stdin=Microsoft-MIEngine-In-jkobqa4i.4it' '--stdout=Microsoft-MIEngine-Out-0g0pijxx.p14' '--stderr=Microsoft-MIEngine-Error-ao40uvpq.t4v' '--pid=Mic
rosoft-MIEngine-Pid-0ojy5dbw.eke' '--dbgExe=C:\MinGW\bin\gdb.exe' '--interpreter=mi'
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> jdf^C
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>  & 'c:\Users\hxrle\.vscode\extensions\ms-vscode.cpptools-1.28.3-win32-x64\debugAdapters\bin\WindowsDebugLaun
cher.exe' '--stdin=Microsoft-MIEngine-In-l0h54wib.meu' '--stdout=Microsoft-MIEngine-Out-oom21h2o.jv1' '--stderr=Microsoft-MIEngine-Error-l0xzyehc.ei4' '--pid=Mic
rosoft-MIEngine-Pid-etyqjdck.gm2' '--dbgExe=C:\MinGW\bin\gdb.exe' '--interpreter=mi'
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> g++ Q1.cpp -o Q1
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> ./Q2
Enter string: DataStructures
Reversed string: serutcurtSataD
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
```

Q3. Write a program that checks if an expression has balanced parentheses.

Code with Output

Q3.cpp > main()

```cpp
#include <iostream>
#include <string.h>
using namespace std;
int main(){

    char exp[100],stack[100];
    int top=-1,f=1;
    cout<<"Enter expression: ";
    cin>>exp;

    for(int i=0;i<strlen(exp);i++){
        if(exp[i]=='(' || exp[i]=='{' || exp[i]=='[')
            stack[++top]=exp[i];
        else if(exp[i]==')' || exp[i]=='}' || exp[i]==']'){
            if(top==-1){f=0;break;}
            char c=stack[top--];
            if((exp[i]==')' && c!='(') || (exp[i]=='}' && c!='{') || (exp[i]==']' && c!='[')){
                f=0; break;
            }
        }
    }
    if(f==1 && top==-1)
        cout<<"Balanced";
    else
        cout<<"Not Balanced";
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> g++ Q3.cpp -o Q3
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> ./Q3
Enter expression: I am a student (in thapar)
Balanced
```

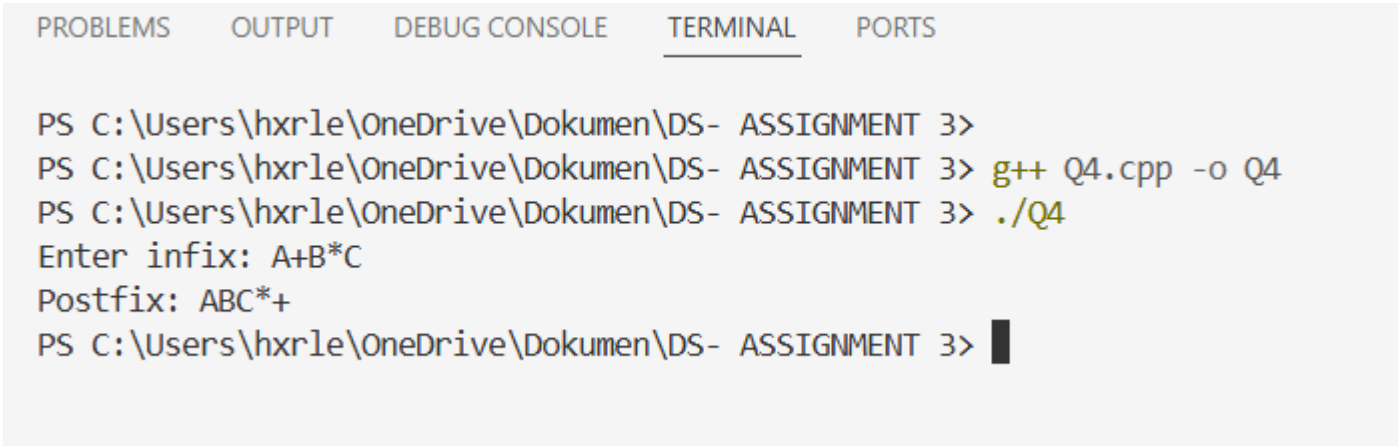
Q4. Write a program to convert an Infix expression into a Postfix expression.

Code :

Q4.cpp > prec(char)

```cpp
#include <iostream>
#include <string.h>
using namespace std;
char stack[100];
int top=-1;
int prec(char c){
    if(c=='^') return 3;
    if(c=='*' || c=='/') return 2;
    if(c=='+' || c=='-') return 1;
    return -1; }
int main(){
    char infix[100],postfix[100];
    cout<<"Enter infix: ";
    cin>>infix;
    int k=0;
    for(int i=0;infix[i]!='\0';i++){
        char c=infix[i];
        if((c>='a'&&c<='z')||(c>='A'&&c<='Z')||(c>='0'&&c<='9'))
            postfix[k++]=c;
        else if(c=='(')
            stack[++top]=c;
        else if(c==')'){
            while(top!=-1 && stack[top]!='(')
                postfix[k++]=stack[top--];
            if(top!=-1 && stack[top]=='(')
                top--;
        }
        else{
            while(top!=-1 && prec(stack[top])>=prec(c))
                postfix[k++]=stack[top--];
            stack[++top]=c;
        } }
    while(top!=-1)
        postfix[k++]=stack[top--];
    postfix[k]='\0';
    cout<<"Postfix: "<<postfix;
}
```

OUTPUT :


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> g++ Q4.cpp -o Q4
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> ./Q4
Enter infix: A+B*C
Postfix: ABC*+
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>

Q5. Write a program for the evaluation of a Postfix expression.

Code with Output :

Q5.CPP > ...

```cpp
#include <iostream>
#include <ctype.h>
using namespace std;
int main(){
    char exp[100];
    int stack[100],top=-1;
    cout<<"Enter postfix: ";
    cin>>exp;
    for(int i=0;exp[i]!='\0';i++){
        if(isdigit(exp[i]))
            stack[++top]=exp[i]-'0';
        else{
            int b=stack[top--];
            int a=stack[top--];
            switch(exp[i]){
                case '+': stack[++top]=a+b; break;
                case '-': stack[++top]=a-b; break;
                case '*': stack[++top]=a*b; break;
                case '/': stack[++top]=a/b; break;
                case '^': stack[++top]=pow(a,b); break;
            }
        }
    }
    cout<<"Result: "<<stack[top];
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
rosoft-MIEngine-Pid-4hwrdyx5.pm4' '--dbgExe=C:\MinGW\bin\gdb.exe' '--interpreter=mi'
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> g++ Q5.cpp -o Q5
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3> ./Q5
Enter postfix: ABC=*+
Result: -490261152
PS C:\Users\hxrle\OneDrive\Dokumen\DS- ASSIGNMENT 3>
```