**Assignment - 35 A Job Ready Bootcamp in C++, DSA and IOT**

**Templates**

1. Write a C++ program to demonstrate the addition of multiple types of data using

generic functions or templates.

Sol – 1.

#include<iostream>

using namespace std;

template<typename T>

T add(T x,T y)

{

return x+y;

}

int main()

{

cout<<"Addition : "<<add<float>(5.2,4);

return 0;

}

2. Write a C++ Program to find Largest among two numbers using function template.

Sol – 2.

#include<iostream>

using namespace std;

template<class T>

T greaterno(T x,T y)

{

return x>y?x:y;

}

int main()

{

cout<<"Greater : "<<greaterno<char>('Z','A');

return 0;

}

3. Write a C++ program to find the largest of three elements using a template.

Sol – 3.

#include<iostream>

using namespace std;

template<class T>

T greaterno(T x,T y,T z)

{

if(x>=y)

{

if(x>=z)

return x;

else

return z;

}

else

{

if(y>=z)

return y;

else

return z;

}

}

int main()

{

cout<<"Greater : "<<greaterno<char>('b','A','a');

return 0;

}

4. Write a C++ Program to Swap data using function template.

Sol – 4.

#include<iostream>

using namespace std;

template<class T>

void swapno(T &x,T &y)

{

T n;

n=x;

x=y;

y=n;

}

int main()

{

int x,y;

cout<<"Enter value of x and y : ";

cin>>x>>y;

cout<<"Before Swapping"<<endl;

cout<<"X is "<<x<<" and Y is "<<y<<endl;

swapno<int>(x,y);

cout<<"After Swapping"<<endl;

cout<<"X is "<<x<<" and Y is "<<y<<endl;

return 0;

}

5. Write a C++ Program to Add two numbers using function template.

Sol – 5.

Same as 1

6. Write a C++ Program to find Sum of Array using function template.

Sol – 6.

#include<iostream>

using namespace std;

template<class T>

T sumarr(T a[],int sizeofar)

{

T sum=0;

for(int i=0;i<sizeofar;i++)

{

sum=sum+a[i];

}

return sum;

}

int main()

{

float a[5];

cout<<"Enter 5 no : ";

for(int i=0;i<5;i++)

{

cin>>a[i];

}

cout<<"Sum of all elements : "<<sumarr<float>(a,5);

return 0;

}

7. Write a C++ Program of Templated class derived from Non-templated class.

Sol – 7.

#include<iostream>

#include<string>

using namespace std;

class Shape

{

private :

string s;

public :

Shape(string a)

{

s=a;

}

};

template<class T>

class Rectangle : public Shape

{

private :

T x,y;

public :

Rectangle(T a,T b,string s):Shape(s)

{

x=a;

y=b;

}

T area()

{

return x\*y;

}

};

int main()

{

Rectangle<double>R(5,6.7,"Rect");

cout<<"Area of Rectangle : "<<R.area();

return 0;

}

8. Write a C++ Program to implement push and pop methods from stack using

template.

Sol – 8.

#include<iostream>

using namespace std;

#include<string>

template<class T>

class stack

{

private :

T a[50];

int size=0;

public :

void push(T t)

{

if(size==50)

{

cout<<"OVERFLOW"<<endl;

}

else

a[size++]=t;

}

T pop()

{

if(size==0)

{

cout<<"UNDERFLOW"<<endl;

return 0;

}

return a[--size];

}

void display()

{

for(int i=0;i<size;i++)

{

cout<<a[i]<<" ";

}

cout<<endl;

}

};

int main()

{

stack<float>s;

s.pop();

s.push(5.2);

s.push(60);

s.push(7.3);

s.display();

cout<<"Deleted item : "<<s.pop()<<endl;

s.display();

stack<string>s2;

s2.push("hey");

s2.push("hello");

s2.push("hii");

s2.display();

cout<<"Deleted item : "<<s2.pop()<<" and "<<s2.pop()<<endl;

s2.display();

return 0;

}

9. Write a C++ Program to Perform Simple Addition Function Using Templates.

Sol – 9.

Same as 1

10. Write a C++ program to implement Hash Table using Template Class.

Sol – 10.

#include<iostream>

using namespace std;

int hashfun(int x)

{

return x%10;

}

int main()

{

int a[10],n;

a[hashfun(50)]=50;

a[hashfun(43)]=43;

a[hashfun(21)]=21;

a[hashfun(62)]=62;

a[hashfun(89)]=89;

a[hashfun(57)]=57;

a[hashfun(36)]=36;

a[hashfun(18)]=18;

a[hashfun(15)]=15;

a[hashfun(4)]=4;

cout<<"Enter a num you want to search : ";

cin>>n;

if(a[hashfun(n)]==n)

cout<<"Value found";

else

cout<<"Value not found";

return 0;

}