**C++ Programming Lab (18MCA16)**

**PART -A**

**==================================================================================**

1. **Write a C++ program to find the sum for the given variables using function with default arguments.**

**//Program:** Demonstrate Default Argument

#include <iostream>

using namespace std;

void sum (int a =2, int b=3, int c=4, int d=5)

{

int res;

res = a + b + c + d;

cout << "Sum =" << res;

}

int main ( )

{

int a, b, c, d;

cout << "\n Enter 4 Nos : ";

cin >> a >>b >>c >>d;

cout<<"-- Output with 1 argument-- "<<endl;

sum (a) ; // 3 values are default

cout<<"-- Output with 2 arguments-- "<<endl;

sum (a, b) ; // 2 values are default

cout<<"-- Output with 3 arguments-- "<<endl;

sum (a, b, c) ; // 1 value is default

return 0;

}

1. **Write a C++ program to swap the values of two variables and demonstrates a function using call by value.**

***//Program: Call-by-Value***

#include <iostream>

using namespace std;

**void swap (int , int );** *// function prototype*

**int main ()**

**{**

int a, b;

*cout<<"\*\*\*\*\*\*\*\* SWAPPING USING CALL BY VALUE \*\*\*\*\*\*\n"<<endl;*

cout << "Input a = ";

cin >> a;

cout << "Input b = ";

cin >> b;

*cout << "\*\*\*\* Output Before swap \*\*\*\*"<< endl;*

cout << " a = " << a << endl;

cout << " b = " << b << endl;

**swap (a, b);**  *// calling a function as: Call-by-value*

*cout << "\*\*\*\* Output After swap \*\*\*\*"<< endl;*

cout << " a = " << a << endl;

cout << " b = " << b << endl;

return 0;

**}**

**void swap (int a, int b) // function definition of swap for Call-by-value**

**{**

int temp;

temp = a;

a = b;

b = temp;

**}**

1. **Write a C++ program the swap the values of two variables and demonstrates a function using Call by reference using reference variable (&).**

**// Program: Call by reference using ‘&’ operator.**

#include <iostream>

using namespace std;

void swap (int &x, int &y)

{

int temp;

temp = x;

x = y;

y = temp;

}

int main ()

{

cout<<"Swapping Using Call By Reference Using(&)\n"<<endl;

int a = 100;

int b = 200;

cout << "\*\*\*\* Output Before swap \*\*\*\*"<< endl;

cout << " a = " << a << endl;

cout << " b = " << b << endl;

swap (a, b);

cout << "\*\*\*\* Output After swap \*\*\*\*"<< endl;

cout << " a = " << a << endl;

cout << " b = " << b << endl;

return 0;

}

1. **Write a C++ program the swap the values of two variables and demonstrates a function using Call by reference using pointer.**

**// Program: Call by reference using ‘&’ operator.**

#include <iostream>

using namespace std;

void swap (int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

int main ()

{

cout<<"Swapping Using Call By ReferenceUsingPointer(\*)\n"<<endl;

int a = 100;

int b = 200;

cout << "\*\*\*\* Output Before swap \*\*\*\*"<< endl;

cout << " a = " << a << endl;

cout << " b = " << b << endl;

swap (&a, &b);

cout << "\*\*\*\* Output After swap \*\*\*\*"<< endl;

cout << " a = " << a << endl;

cout << " b = " << b << endl;

return 0;

}

1. **Write a C++ program to swap the values of two dynamically allocated variables and release the memory after swapping. (use new & delete operators)**

#include <iostream>

using namespace std;

void swap (int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

int main ()

{

cout<<"Swapping Using Dynamic Variable\n"<<endl;

int \*a;

int \*b;

a = new int (50);

b = new int (100);

cout << "\*\*\*\* Output Before swap \*\*\*\*"<< endl;

cout << " a = " << a << endl;

cout << " b = " << b << endl;

swap (a, b);

cout << "\*\*\*\* Output After swap \*\*\*\*"<< endl;

cout << " a = " << a << endl;

cout << " b = " << b << endl;

return 0;

}

**6. Write a program to find the largest, smallest & second largest of three numbers. (Use inline function MAX and MIN to find largest & smallest of 2 numbers)**

#include<iostream>

using namespace std;

inline int MAX (int a, int b)

{

return (a>b) ? a : b;

}

inline int MIN (int a, int b)

{

return (a<b) ? a : b;

}

int main()

{

cout<<" \*\*\*\*\*\*\*\* Demonstrate INLINE Function \*\*\*\*\*\*\*";

int a, b, c, large, small, secLargest;

cout<<"\n Enter values for : a, b and c :";

cin>>a>>b>>c;

large = MAX (a, MAX (b, c) );

small = MIN (a, MIN (b, c) );

cout<<"\n Largest number :"<< large <<endl;

cout<<"\n Smallest number :"<< small << endl;

secLargest = (a+b+c) -large - small;

cout<<"\n Second largest :"<<secLargest;

return 0;

}

|  |
| --- |
| **7.Write a program to calculate the volume of different geometric shapes like cube, cylinder and sphere and hence implement the concept of Function Overloading.** |

#include<iostream>

using namespace std;

float volume ( float, int);

float volume (float);

int volume (int);

int main()

{

float cRadius, sRadius, height;

int side;

cout <<"Enter cylinder Details:" << endl ;

cout <<"Cylinder Radius and height= ";

cin >> cRadius >>height;

cout<<endl<< "Enter Cube Details:";

cout <<"Cube Side = " ;

cin>>side;

cout<<endl<< "Enter Sphere Details:";

cout <<"Sphere Radius = " ;

cin>>sRadius;

cout << "Cube Volume = "<< volume(side);

cout << "Cylinder Volume = "<< volume(cRadius, height);

cout << "Sphere Volume = "<< volume(sRadius);

return 0;

}

float volume (float rad, int height)

{

return ( 3.14 \* rad \* rad \* height );

}

float volume ( float rad )

{

return ( (4/3.0) \* 3.14 \* rad \* rad \* rad);

}

int volume ( int side )

{

return (side \* side \* side);

}

8. **Write a C++ program to create a template function for Bubble Sort and demonstrate sorting of integers and doubles.**

#include<iostream>

using namespace std;

template <class T > void bubble(T a[], int n)

{

int i,j;

T temp;

for(i=1;i<n;i++)

{

for(j=0;j<n-i;j++)

{

if(a[j]>=a[j+1])

{

temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

}

}

}

int main()

{

int intarr[10],i,n,m;

double dbarr[10];

cout<<”Enter size of an integer array”;

cin>>n;

cout<<”enter elements for integer array”;

for(i=0;i<n;i++)

cin>>intarr[i];

cout<<”Enter size of an double array”;

cin>>m;

cout<<”enter elements for double array”;

for(i=0;i<m;i++)

cin>>dbarr[i];

bubble(intarr,n);

bubble(dbarr,m);

cout<<”Sorted integer array elements”<<endl;

for(i=0;i<n;i++)

cout<<intarr[i]<<endl;

cout<<”Sorted double array elements”<<endl;

for(i=0;i<n;i++)

cout<<dbarr[i]<<endl;

return 0;

}