**1. WAP to read data in hex and octal format then print result in**

**hexadecimal, octal and decimal format.**

#include<iostream>

using namespace std;

int main()

{

int a,b;

cout<<"enter number in hex"<<endl;

cin>>hex>>a;

cout<<"enter number in oct"<<endl;

cin>>oct>>b;

cout<<"a in hex="<<hex<<a<<endl;

cout<<"b in octal="<<oct<<b<<endl;

cout<<"a in dec ="<<dec<<a<<endl;

cout<<"b in dec="<<dec<<b<<endl;

}

**2). Find 2 more differences between printf and scanf over cin**

**and cout?**

1. printf & scanf needs format specifiers

cout & cin doesn’t required.

2. printf & scanf returns value

cout & cin doesn’t return any value.

**3. WAP to print first 5 perfect numbers starting from 100 ?**

#include <iostream>

using namespace std;

int main()

{

int a , b,i,x,sum=0,count=0;

for(i=100;i;i++)

{

sum=0;

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

{

if(i%a==0)

sum=sum+a;

}

if(sum==i)

{

count++;

cout << sum <<endl ;

}

if(count==5)

break;

}

return 0;

}

**4. WAP to find given number is strong number or not ?**

#include <iostream>

using namespace std;

int fact(int i)

{

if (i==1 || i==0)

return 1;

else

return (i\*fact(i-1));

}

int main()

{

int a , b,i,x,sum=0;

cout << "ENTER your NUMBER" << endl;

cin>>a ;

b=a;

while(b)

{

i=b%10;

x=fact(i);

sum=sum+x;

b=b/10;

}

if(sum==a)

cout << " number is a strong number =" << sum<< endl;

else

cout << " number is NOT a strong number !" << endl;

return 0 }

**5. Try to find 3 more input output formated flags?**

[**boolalpha:**](https://www.geeksforgeeks.org/ios-manipulators-boolalpha-function-in-c/) The boolalpha manipulator of stream manipulators in C++ is used to turn on bool alpha flag

**ends:** The ends manipulator of stream manipulators in C++ is used to Output a null

**fixed:** The fixed manipulator of stream manipulators in C++ is used to Turns on the fixed flag

**6. Design a function to find sum of 2 numbers with the help of**

**call by reference?**

#include <iostream>

using namespace std;

// Function prototype

int sum(int& n1, int& n2)

{

return (n1+n2);

}

int main()

{

int a , b,i;

cout << "ENTER TWO NUMBER" << endl;

cin>>a ;

cin >>b;

i= sum(a, b); // function call

cout << "sum of two number =" << i<< endl;

return 0;

}

**7. Design a function to sort the data in ascending order with the**

**help of call by reference?**

#include<iostream>

using namespace std;

int sum(int (&a)[5])

{

int b;

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

{

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

{

if(a[j]<a[i])

{

b=a[j];

a[j]=a[i];

a[i]=b;

}

}

}

}

int main()

{

int c;

cout<<"enter size of array";

cin>>c;

int a[c]={2,3,1,4,5};

sum(a);

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

{

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

}

}

**8. What is the difference between pointer and reference with basic examples?**

1). A pointer is variable that points to the address of another variable whereas reference is an alias name for a variable.

2).The reference has to be always initialised at the time of declaration while it is not necessary to do the same with pointer variable.

3).A pointer can change its value several time in a program whereas a reference cannot change its value.

**9. What are the features of c++ and advantages?**

Features of c++

1. Encapsulation 2. Objects and class 3. Inheritance 4. Data Abstraction 5. Polymorphism

Advantage :

1. It is a platform independence programming language.
2. Memory management is in hand of programmer.
3. C++ is pretty much compatible with C
4. Scalability is reliable. Means program is capable of running on a small scale as well as a large scale of data.

**10. Explain types of access specifiers?**

There are three type of access specifiers (purpose is Data hiding, nothing else)

1. Public 2. Protected 3. Private
2. Public: user can give the permission to access data or member function to everyone at everywhere.
3. Private: The class members declared as private can be accessed only by the functions inside the class. They are not allowed to be accessed directly by any object or function outside the class.

Note: Only the member functions or the friend functions are allowed to access the private data members of a class.

1. Protected: Protected access modifier is similar to that of private access modifiers, the difference is that the class member declared as Protected are inaccessible outside the class but they can be accessed by any subclass (derived class) of that class.