DATA Types

int - 2147483648 to 2147483647

Float - 1.2E-38 to 3.4E+38

Double - 1.7E-308 to 1.7E+308

char - 0 to 255

signed char - -127 to +127

Variables

container which stores a value

Eg: A = 10

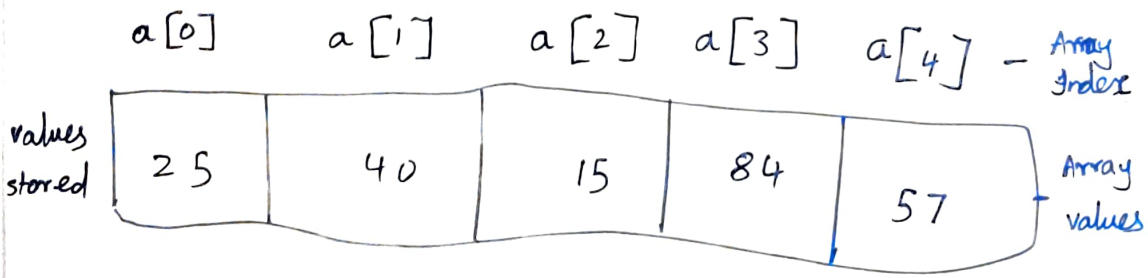
A → variable

10 → value

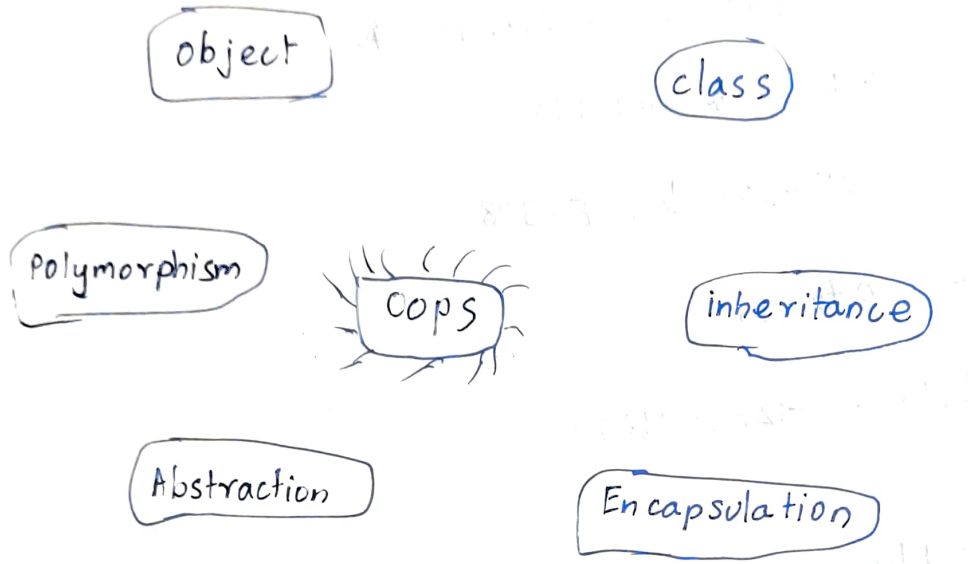
ARRAY

Array stores multiple number of values in a single variable

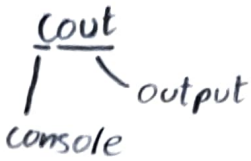
Eg: int a[5] → stores 5 value



Lets learn OOPS [object oriented programming system]



```
# include <iostream>
using namespace std;
int main()
{
   cout<<" Hello World ... ";
}
```



output :

```
Hello World
```

---

using namespace std;

If we didn't use it shows error (cout not declared)

cout and cin is stored in a package called

also can be written as ....

```
#include <iostream>
int main()
{
  std :: cout<<" Hello ... ";
}
```

output :

```
Hello...
```

declaring a value ....

```
#include <iostream>
int main()
{
  int a;
std:: cout <<" Enter the value of A : ";
std:: cin >> a;
std :: cout <<" A value :" << a;
}
```

<< → insertion
>> → ~~dissertion~~ extraction

output :

```
Enter the value of A : 25
A value : 25
```

To reduce typing use "using namespace std;" instead of std::

easy method

also can be used like ....

```
#include <iostream>
using std:: cout;
using std:: cin;
int main()
{
  int a;
```

→

```
cout <<" Enter value of a : ";
cin >> a;
cout << "value:" << a;
}
```

output :

```
Enter value of a : 22
value : 22
```

---

use of namespace ....

can't delare two variable again (re declaration error)

```
#include <iostream>
using namespace std;
int main()
{
    string name = "ram";
    string name = "ram";
}
```

can be used through the concept of namespace...

→

```
#include<iostream>
using namespace std;
namespace name1 {string name = "ram";}
namespace name2 {string name = "ram";}
int main()
{
cout << name1::name;
}
```

Output:

ram

---

error in the use of namespace ....

```
#include<iostream>
using namespace std;
namespace name1 {string name = "ra";
                 int age = 25;}
namespace name2 {string name = "ra";}
int main()
{
cout << name2::name;
{cout << name2::age;}
{cout << name1::age;}
}
```

✗ ✓

output :

ra 25

---

use '::' "scope resolution Operator" to use the variables in main for function from namespace.

---

also written as.. ..

```
#include<iostream>
using namespace std;
namespace name1 { string name="ram";
                  int age =25; }
namespace name 2 { string name ="ram"; }
using namespace name1;
int main()
{
  cout <<name;
  cout<< age;
}
```

output:

ram 25

Getting inputs in C++ programming....

```
#include <iostream>
using namespace std;
int main()
{
    int a;
    cout<<"Enter the Integer value:";
    cin>>a;
    cout<<" value of A is :"<<a;
```

→ Tells, variable 'a' stores a integer value

output:

```
Enter the Integer value:1
value of A is :1
```

```
#include <iostream>
using namespace std;
int main()
{
    int a,b;
    cout<<" Enter two values : ";
    cin>>a>>b;
    cout<<" value of A and B
    cout<<" Total .... "<<a+b;
}
```

output:

```
Enter two values : 10 90
Total .... 100
```

float → decimal values Eg : 25.25

```
# include<iostream>
using namespace std;
int main()
{
  float a,b,c;
 conte<<" Enter three values :";
 cin >>a >>b >>c;
 cout <<" Total:" << a+b+c
}
```

output :

Enter three values : 99.8 0.1 0.1

Total : 100

---

char → character letters, symbols etc...

```
#include<iostream>
using namespace std;
int main()
{
  char a;
  cout<<" \n Enter character: ";
  cin>>a;
  cout <<"char is:"<< a;
}
```

output :

Enter character : a

char is : a

getting string using array in char ✗ hard

getting string using string ✓ easy

~~#include <iostream>~~

```
#include <iostream>
using namespace std;
int main()
{
 string a;
 cout<<"\n Enter string :";
 cin>>a;
 cout<<a;
}
```

output:

Enter string : MukeshDharman

MukeshDharman

↓

If → Mukesh_Dharman

↳ space

output:

Enter string : Mukesh_Dharman

Mukesh

To avoid This →

use getline () to avoid

```
# include <iostream>
using namespace std;
int main()
{
  string a;
  cout<< "\n Enter string : ";
  getline (cin, a);
  cout << a;
}
```

input stream

variable the value to be stored.

Pass two parameter

getting input using 'cin' and storing in a

Output :

Enter string : I Love you Machii

I Love you Machii

String in C++ ....

string is in the std package [using namespace std]

* Input functions
* capacity functions
* Iterator functions
* Manipulating functions

Structure

Syntax Modal :

```
Struct employee
{
 char name[50];
 int empid;
 float salary;
};
int main ()
{
   struct employee e1,e2,e3;
   e1.name;
   e2.empid;
   e3.salary;
   return 0;
}
```

declaration, if we use string variable

```
# include <iostream>
using namespace std;
int main ()
{
string a = " I love you Machi";
cout <<"string : "<<a << endl;
}
```

output :

string : I love you Machi

---

creating an object for string and pass " I love you Machi" as a constructor value

string → In-built class

a → string variable
↳ object

```
# include <iostream>
using namespace std;
int main()
{
In-built class string a ("I love you Machi");
                        ↳ object
cout<<a << endl;
}
```

output;

I Love you Machi

String Concatenation (addition of two strings is known as string concatenation)

```
#include <iostream>
using namespace std;
int main()
{

    string firstname = "I Love you";
    string Lastname = " Machi";
also can be written as "firstname + " " + Lastname"
    cout << firstname + Lastname << endl;
}
```

operator

Output: I love you Machi

According to Concatenation there is "Append function"

→ In-built function

Append function in Concatenation

```
#include <iostream>
using namespace std;
int main()
{

    string first = "I love my ";
    string last = " Machi";
Tells that append first to last →
    string fullname = first.append(last);
    cout << fullname << endl;
}
```

output:

I love my Machi

Printing 0th of a string

```
#include <iostream>
using namespace std;
int main()
{

String name = "pradeep";
cout << name << endl;
cout << name[0] << endl;
}
```

Output:

pradeep

p

String Access

Replace V instead of A ?

```
#include <iostream>
using namespace std;
int main()
{

String name = "Apradeep";
cout << name << endl;
cout << name[0] << endl;
name[0] = 'v';
cout << name << endl;
}
```

single quotes for a single character

output:

~~Pradeep~~ Apradeep

A

vpradeep

* Input Functions (string)

```
int main()
{
    string str;
    cout << " Enter the string ";
    cin >> str;
    cout << " string : " << str;
}
```

output :

Enter the string : kamalesh

string : kamalesh

Enter the string : I love

string : I

```
int main()
{
    string str;
    cout << " Enter the string ";
    cin >> str;
    cout << " string: " << str;
    cout << " enter the string ";
    getline(cin, str);
    cout << str;
}
```

Parameters

cin → first parameter

str → variable, where we store

Output : Garbage values

Enter the string : RAM

String : RAM Enter the string : String :

To avoid this ↓ use (fflush(stdin))

```
int main()
{
 string str;
 cout<<" Enter the string ";
 cin >> str;
 cout<<"string:" << str << "\n";
 fflush (stdin)
 cout << " Enter the string:";
 getline (cin, str);
 cout << " string :" << str;
}
```

→ to move to the next line

→ used to avoid problems in getting input of char or string

→ use to avoid garbage values

output :

Enter the string : Pradeep

String : pradeep Enter the string : Hi Machi

String : Hi Machi

To move to the next line use "\n" or endl;

Push back function

→ used to add a letter at the end of a string

```
int main()
{
  string str;
  cout<<" Enter the string!";
  cin>>str;
  str.push_back('s');
  cout<<str<<endl;
}
```

output :

Enter the string : I love my brother

I love my brothers

Pop back function

→ used to remove the last letter of a string

```
int main()
{
  string str;
  cout<<" Enter the string :";
  cin>>str;
  str.push_back('s');
  cout<<Str<<endl;
  str.pop_back();
```

```
    cout << str << endl;
}
```

output :

```
Enter the string : My life my brothers
My life my brotherss
My life my brothers
```

* Capacity function (string)

getting size

```
int main()
{
string str ("Love you da Machi");
cout <<str<< "\n";
cout << "size : " << str.size << "\n";
cout << "Length : " << str.length << "\n";
}
```

output :

```
size : 17
length : 17
```

to get the size in bytes

```
cout << " Max size : " << str.max_size << "\n";
```

output : 2147483647

This function helps to find the maximum capacity in bytes

* Iterator function

In string std package

```
- - - -
- - - -
int main ()
{
string str = "pradeep";
string :: iterator D;
for (D = str.begin(); D != str.end(); D++)
cout<<*D << "\n";
}
```

The first letter of string will be given here

end of loop

The given letter from the string should not be the end Address

output :

p
r
a
d
e
e
p

If "\n" is not there [cout<<*D<<"\n";] output will be

p r a d e e p

getting Reverse iterator ⟶

```
int main()
{
  string str = "kamal";
  string::reverse_iterator D2;
  for (D2 = str.rbegin(); D2 != str.rend(); D2++)
  cout << *D2;
}
```

Function to get reversed iterator

output:

```
L a m a k
```

---

* Manipulating functions

Inbuilt swaping

```
int main()
{
  string x = "Pradeep";
  string y = "Machi";
  cout << "Before :" << x << "\n";
  cout << "Before :" << y << "\n";
  x.swap(y);
  cout << "after :" << x << "\n";
  cout << "after :" << y << "\n";
}
```

output :

```
Before : Pradeep
Before : Machi
after : Machi
after : pradeep
```

If Statement :

which num is greater a or b)?

```
- - - -
- - - -
int main ()
{
int a,b;
cout <<" Enter The value A & B :\n";
cin >> a >> b;
if (a> b)
{
  cout <<" The Greatest number.... "<<a
}
}
```

Output :

```
Enter the value A & B : 20 15
The Greatest number .... 20
The Gree
```

Multiple If statement

```
- - - -
- - - -
int main()
{
  int a,b;
  cout<<" Enter the values... "<<"\n";
  cin>>a>>b;
  if (a>b)
  {
  cout<<" The greatest....is "<< a;
  }
  if (a<b)
  {
  cout <<" The greatest.... is "<< b;
  }
  if (a==b)
  {
  cout<<a <<"and "<<b <<"are equal";
  }
}
```

After entering the values it checks the first statement if it is wrong it automatically checks the next following statements.

After checking the last statement the program closes

output:

Enter the values... 10 10

10 and 10 are equal

If else statement

finding that the letter is vowel or not?

As a character we use single quotes '' , ''
a

W.K
logical operator
"||" → or
"&&" → and

We have 52 Alphabets here (26 + 26 = 52)
small capital

```
int main()
{
  char c;
  cout<<" Enter The character :\n";
  cin >>c;
  if(c=='a' || c=='e' || c=='i' || c=='o'
        || c=='u' || c=='A' || c=='E'
     || c=='I' || c=='O' || c=='U')
  {
   cout <<" It's a vowel... " << c;
  }
  else
  {
  cout <<" It's not a vowel... " <<c;
  }
}
```

output :

Enter the character : A

It's a vowel... A

Else if ladder

```
int main()
{
  int h, t;
  float c;
  cout << " Enter the value of hardness,
        tensile strength, carbon: " << endl;
  cin >> h >> t >> c;
if (h > 50 && c < 0.7 && t > 5600)
{
  cout << " steel grade : 10" << endl;
}
else if (h > 50 && c < 0.7)
{
  cout << " steel grade : 9 " << endl;
}
else if (c < 0.7 && t > 5600)
{
  cout << " steel grade : 8 " << endl;
}
else if (h > 50 && t > 5600)
{
```

h → hardness
t → tensile strength
c → carbon

```
else if (h >50 && t > 5600)
{
 cout << " steel grade : 7" << endl;
}
else if(h >50 || c <0.7 || t >5600)
 {
  cout << "steel grade : 6" << endl;
 }
 else
 {
  cout <<" steel grade: 5 " << endl;
 }
}
```

nested if

```
- - - - -
- - - - -
int main()
{
  char gender;
  int age;
  cout << "\n Enter your gender: ";
  cin >> gender;
  cout << "\n Enter your age : ";
  cin >> age;
  if (age >=18)
  {
    cout << "\n Enter your gender : ";
    cin >> gender;
    if ( gender == 'M' || gender == 'm')
    {
      cout << "\n Go To Room -5";
    }
    else if (gender == 'F' || gender == 'f')
    {
      cout << "\n Go To Room -6";
    }
    else
    {
      cout << "\n Invalid Gender Input";
    }
}
```

```
    else
    {
      cout <<"In your age is under 18....";
    }
}
```

Switch statement

Getting month name after typing its number

```
int main ()
{
 int m;
 cout << " Enter the month in Num (1-12) \n";
 cin >> m;
 switch (m)
 {
 case 1 :
  cout << " january " << endl;
  break;
 case 2 :
  cout << " februrary " << endl;
  break;
   .
   .
   .
  case 12 :
  cout << " december " << endl;
  break;
  default
   cout << " Invalid months number " << endl;
   breaks;
  }
}
```

Write a program to get its no of days

```
int main()
{
 int m;
 cout<<"\n Enter the moth in Num";
 cin>>m;
 switch (m)
{
 case 1 :
 case 3 :
 case 5 :
 case 7 :
 case 5 :
 case 7 :
 case 8 :
 case 10 :
 case 12 :
    cout<<" Days : 31 "<<endl;
    break;
 case 2 :
    cout <<" Days : 28 || 29 "<<endl;
    break;
 case 4 :
 case 6 :
 case 9 :
 case 11 :
    cout <<" Days : 30 "<< endl;
    break;
 }
}
```

Group Switch

If else

Qn Eg: basic salary Rs. 1500
HRA = 10% of basic salary
DA = 90% of basic salary
if salary is either equal or above 1500
then HRA = Rs. 500 & DA = 98% of basic salary
if the employee's salary is input through the keyboard
Find gross salary

Ans ↓

```
- - - -
- - - -
int main()
{
float bs, gs, da, hra;
cout << "\n Enter your basic salary:";
cin >> bs;
if (bs <1500)
{
  hra = bs * 10/100;
  da = bs* 90/100;
}
else
{
  hra =500;
  da= bs* 98/100;
}
```

```
    gs=bs+hra+da;
    cout<<"\n Basic Salary :"<<bs
    cout<<"\n Hra         :"<<hra
    cout<<"\n DA          :"<<da;
    cout<<"\n-------------------\n";
    cout<<"\n Cross salary :"<<bs;
}
```

output:

```
Enter ur bs: 1200

Basic salary : 1200
Hra          : 120
DA           : 1080
-------------------
Cross salary : 2400
```

(or)

```
Enter your basic salary : 5000

Basic salary :5000
Hra          :500
DA           : 49000
-------------------
Cross Salary :10400
```

sample program — If else

Qn: Insurance Eligibility status

A company insures its drivers in the following cases:

a. If the driver is married
b. If the driver is unmarried, male & above 30 yrs of age
c. If the driver is unmarried, female & above 25 yrs of age

↓ ans

```
- - - -
- - - -
int main()
{
char marital, gender;
int age;
cout << "\n Enter marital status : M as married | U as unmarried";
cin >> marital;
if (marital == 'M' || marital == 'm')
{
cout << "\n you are eligible for insurance";
}
else if (marital == 'U' || marital == 'u')
{
cout << "\n Enter Gender : M as male | F as Female :";
cin >> gender;
cout << "\n Enter Age :";
```

```
cin >> age;
IF((gender == 'M' || gender == 'm')
                    && age >= 30)
 {
  cout << "\n you are Eligible for
                        insurance";
 }
else if ((gender == 'F' || gender
                    == 'f')
                && age >= 25)
 {
  cout << "\n you are Eligible
                    for Insurance";
 }
else
 {
  cout << "\n you are not Eligible for
                    Insurance...";
  cout << "\n nor";
  cout << "\n Invalid Gender Input....";
 }
```

Sample prog

qn A library charges a fine for every book returned late for first 5 days the fine is 50 paise, for 6-10 days the fine is one rupee & above 10 days fine is 5 rupees.

if you return after 30 days your membership will be cancelled.

write a program to accept the no of days the member is late to return the book & display the fine or the approx message.

>0 <=5 /0-50	>=6 <=10 /1	>10 <=30 /5
		>30

```
int main()
{
  int days;
  cout << "Enter the number of days :";
  cin >> days;
  if( days > 0 && days <= 5)
  {
    cout << "In Per Day fine Amount is : 0.50";
    cout << "In Total fine Amount is :" <<
                                  days * 0.50;"
  }
```

```
else if (days>=6 && days<=10)
{
 cout<<"\n per day fine Amount is :1";
 cout<<"\n Total fine Amount is :"<<days*1;
}
else if (days>10 && days<=30)
{
 cout<<"\n per day fine Amount is :5";
 cout<<"\n Total Fine Amount is :"<<days*5;
}
else
{
 cout<<"\n membership cancelled....";
}
}
```

Looping Statement

1. while — entry check
2. do while — exit check
3. for — entry check
4. for each — entry check

---

while loop:

```
- - - -
- - - -
int main ()
{
int i=1 , n;
cout <<" \n Enter the value of n :";
cin >> n;
while (i <=n)
{
  cout <<" \n " <<i ;
  i++ ;
}

}
```

do while :

```
- - - -
- - - -
int main()
{
int n, i = 1;
cout << "\n Enter the value of n :";
cin >> n;
do
{
if (i % 2 == 0)
{
cout << i << endl;
}
i++;
} while (i <= n);

}
```

For loop: (counter control)

```
- - - -
- - - -
int main ()
{
 int i , n;
 cout <<" \n Enter the limit : ";
 cin >> n;
 for(i = 1 ; i <= n ; i++)
 {
  cout << i << endl;
 }
}
```

Sample prob | for loop:

table)

```
- - - -
- - - -
int i, n, t;
cout <<" \n Enter the limit :";
cin >> n
cout <<" \n Enter the table : ";
cin >> t;
for (i = 1 ; i <= n ; i++)
```

```
    {
        cout <<t <<" * " << i<< t* i << endl;
    }
}
```

2:1:00 For each new loop in C++

while

```
int main ()
{
int a[ ] = {65, 66, 67, 68, 69, 70}, i=0;
while (i <= 5)
{
 cout <<a[i] << endl;
  i++;
}
}
```

output:

```
65
66
67
68
69
70
```

do while

```
int main()
{
 int a[] = {65,66,67,68,69,70}, i=0;
  do
 {
   cout << a[i] << endl;
 i++;
 } while (i<=5);
}
```

for

```
int main()
{
 int a[] = {1, 2, 3, 4, 5, 6, 7}, i = 0
 for (i=0; i<=5; i++)
 {
  cout << a[i] << endl;
 }
}
```

For each

array

```
int main()
{
int a[] = {1, 2, 3, 4, 5, 6, 7};
for (int x : a)
{
  cout << x << endl;
}
}
```

output :

1
2
3
4
5
6
7

```
- - - -
- - - -
int main()
{
  char a[] = {65,66,67,68,69,70};
  for(char x : a)
  {
    cout << x << endl;
  }
}
```

output:

```
A
B
C
D
E
F
```

---

new trick in foreach (auto → automatic)

```
- - - -
- - - -
int main()
{
 int a[] = {65,6667,68,69,70};
 for (auto x : a)
 {
  cout << x << endl;
 }
}
```

sample program

find sum of n numbers

```
- - - -
- - - -
int main()
{
  int n, i, total = 0
  cout << "\n Enter the value of n:";
  cin >> n;
  for (i=0 ; i <=n ; i++)
  {
   total = total + 1 ;
  }
}  cout << "\n sum of N num is :" << total
```

Simple prog

Finding the factors of given number

```
int main()
{
  int n;
  cout<<"\n Enter the Number : ";
  cin >>n;
  for (int i=1; i<=n ; i++)
  {
    if (n%i==0)
        cout<<i<< endl;
  }
}
```

Sample program

Program for ~~an~~ armstrong number between 100 - 999

what is armstrong number ?

153

1 5 3

$1^3 = 1$ $5^3 = 125$ $3^3 = 27$ ⇒ 153

If both are equal it is Armstrong, otherwise not Armstrong

eg

```
- - - -
- - - -
int main()
{
int sum=0, n, t, r;
cout << "\n Enter the 3 digit number:";
cin >> n;
t = n;
while (n>0)
{
  r = n % 10;
  Sum = sum + (r*r*r);
  n = n/10;
}
```

storing n in a dummy variable 't'

```
if (sum==t)
 {
  cout<<" Armstrong Num";
 }
else
 {
  cout<<"Not an Armstrong Num";
 }
}
```

AM

```
int main()
{
int sum=0,n, t,r;
for( int i=100; i<=999 ;i++)
{
 n=i;
 while (n>0)
  {
    r=n %10;
    sum = sum + (r*r*r);
    n=n/10;
  }
 if(sum==i)
 {
  cout<<i<< endl;
 }
} sum=0;
}
```

output :

153
370
371
407

Sample program :

~~Program to find the factors~~

Program for prime number between 1-100

eg

```
- - - -
- - - -
int main()
{
 int n, c = 0;
 cout << "\n Enter the number :";
 cin >> n;
 for (int i = 1; i <= n; i++)
 {
  if (n % i == 0)
   c++;
 }
 if (c == 2)
  {
   cout << n << " is a prime Number...";
  }
 else
  {
   cout << n << " is not a prime number...";
} }
```

```
- - - -
- - - -
int main()
{
  int n, c=0;
  for (n=1; n<=100; n++)
  {
    for (int i=1; i<=n; i++)
    {
      if (n%i==0)
        c++;
    }
    if (c==2)
    {
      cout << n << " is a prime num..." << endl;
    }
    c=0;
  }
}
```

Array

declaration method 1

```
int main()
{
    int a[5] = {10,20};
    int b[] = {1,2,3,4,5};
    for(int x : a)
    cout << x << endl;
    cout <<"----------" << endl;
    for (int x : b)
     cout << x << endl;
}
```

Finding size of a from the above program...

```
int main()
{
    int a[5] = {10,20};
    cout <<"count :" << sizeof(a)/sizeof(int);
}
```

SP...

Program to find a number in an array.

```
- - - -
- - - -
int main()
{
int a[100], n, i, x;
cout << "\n Enter the limit : ";
cin >> n;
for (i=0; i<n; i++)
{
  cout <<"\n Enter the value : ";
  cin >> a[i];
}
cout <<"\n Enter the value to search:";
cin >> x;
for (i = 0; i<n; i++)
{
   if (a[i] == x)
   {
     cout << "value found @" <<i;
     return 0;
   }
}
cout <<" value not found ";
}
```

It is called Linear search ....

SP....

Find sum of element in an array.

```
- - - -
- - - -
int main()
{
    int a[100], n, i, sum=0;
    cout << "\n Enter the limit :";
    cin >> n;
    for (i=0; i<n ; i++)
    {
     cout << "\n Enter the value :";
     cin >> a[i];
     sum+ = a[i];
    }

    cout << "sum : " << sum ;
}
```

sp... find greatest element in an array.

```
- - - -
- - - -
int main()
{
 int a[100], n, i, t;
 cout<<"\n Enter the value: ";
 cin>>n
 for (i=0; i<n; i++)
{
  cout<<"\n Enter the value: ";
  cin>>a[i];
 }
 t=a[0];
 for (i=0; i<=n; i++)
 {
   if(t<a[i])
   {
     t=a[i];
   }
 }
}  cout<<"\n greatest num is..."<<t;
```

To find smallest change as >

Sp....

Sort an array in ascending order

```
int main()
{
 int a[100], n, i, j, temp;
 cout<<"\n Enter the limit.";
 cin >>n;
 cout <<"\n Enter "<<n<<"Value:"
                    <<endl;
 for(i=0 ; i<n ; i++)
 {
   cin >>a[i];
 }
 for(i=0 ; i<n ; i++)
  {
    for (j=i+1; j<n ; j++)
     {
       if(a[i] <= a[j])   to get ascending change as >
        {
          temp = a[i];
          a[i] = a[j];
          a[j] = temp;
        }
     }
  }
}
```

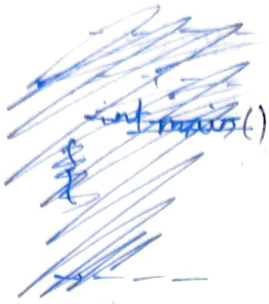
```
cout <<"\n sorted Array : " << endl;
for(i = 0; i < n ; i++)
{
  cout << a [i] << endl;
}
}
```

Function

```
void display ()
{
  int a, b, c;
  cout << "\n Enter two nums: ";
  cin >> a >> b;
  c = a + b;
  cout << "Total : " << c;
}

int main ()
{
  display ();
  display ();
}
```

Function overloading



normal function

```
int sum (int a, int b)
{
  return a+b;
}
int main ()
{
  cout << "Total : " << sum(10,20);
  return 0;
}
```

overloading →

```
int sum (int a, int b)
{
  return a+b;
}
int sum (int a, int b, int c)
{
  return a+b+c;
}
int main()
{
  cout << " Total : " << sum(10,20) << endl;
  cout << " Total : " << sum(10,20,30) << endl;
  return 0;
}
```

another eg

```
void sum(int a, int b)
{
  return a+b;
}
void sum( float a, float b)
{
  return a+b;
}
int main()
{
cout << "Total: " << sum(10,20)<< endl;
cout<< "Total: " << sum(10.10f,20.20f)<<
                                    endl;
```

output:

error

use f in float parameters

Default argument

```
void biodata (string name, int age,
                        string city)
{
  cout << name << "is from" << city
  = "salem" << "and age is " << age << endl;
}
int main ()
{
biodata ("Ram", 25);
biodata ("Sam", 22, "Namakkal");
return 0;
}
```

(new)

Templates

reference type, so address is given...

```
- - - -
- - - -
void swaping(int &a, int &b)
{
int t=a;        // temporary value
a=b;
b=t;
}
int main()
{
int a=10, b=20;
cout<<" before swap A: "<<a<<" B: "<<b<<
~~cout<<" before swap A:~~                endl;
~~cout<<" After~~
swaping(a,b);
cout<<" After swap A: "<<a<<" B: "<<b<<
}                                          endl;
```

JFF a understanding

↓ real ↓

```
- - - -
- - - -
template <class T>      // can give any other name & can give any variable
void swapping(T &a, T &b)
{                       // gives the output of both the ones
   T t=a;
   a=b;
   b=t;
}
```

```
int main()
{
    char a = 'A', b = 'B';
    int x=10, y=20;
    cout<<"Before swap A :"<<a<<" | B :"<<b<<endl;
    swaping (a, b);
    cout<<"After swap A :"<<a<<" | B :"<<b<<endl;
    cout<<"Before swap X :"<<x<<" | Y:"<<y<<endl;
    Swaping (x, y);
    cout<<"After swap X :"<<x<<" | Y :"<<y<<endl;
}
```

char swaping

int swaping

saved the programss from 3.05 to 3.45

→

~~Index.1~~

Inheritance

Full concept in 11th computer book