

# Support Session



Hi !!



Any question !!



# Agenda

- Variables
- Operators
- If condition
- Break
- Loops
- Nested loops
- Break
- Array

# First Program

**Say "Hello World"**



# Output

To do this and to **print some sentences in the screen** using ..

```
cout <<
```

Output

```
cout << Hello World ;
```

# Start program



- how to **Start** any program !!?



# Start program

```
#include <iostream>
using namespace std;
int main ()
{
    return 0;
}
```

# Start program

```
#include <iostream>
using namespace std;
int main ()
{
    return 0;
}
```



# Start program

```
#include <iostream>
using namespace std;
int main ()
{
    return 0;
}
```



# Start program

```
#include <iostream>
using namespace std;
int main ()
{
    return 0;
}
```



# Start program

```
#include <iostream>
using namespace std;
int main ()
{
    return 0;
}
```



# Start program

```
#include <iostream>
using namespace std;
int main ()
{
    return 0;
}
```



# Start program



## First program

```
#include <iostream>
using namespace std ;
int main ()
{
    cout << Hello World ;
    return 0;
}
```

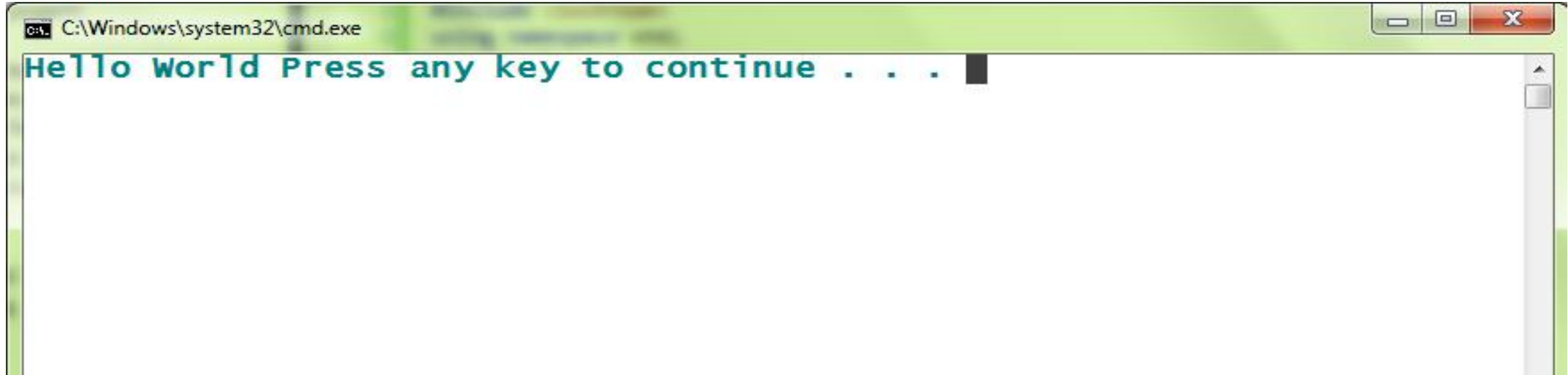


# First program

```
#include <iostream>
using namespace std ;
int main ()
{
    cout << "Hello World" ;
    return 0;
}
```

Just print "Hello world"  
in the screen

# First program



A screenshot of a Windows command prompt window. The title bar at the top shows the file path 'C:\Windows\system32\cmd.exe'. The main area of the window displays the text 'Hello world Press any key to continue . . .' in a green monospaced font. A black cursor is positioned at the end of the text. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
C:\Windows\system32\cmd.exe
Hello world Press any key to continue . . .
```

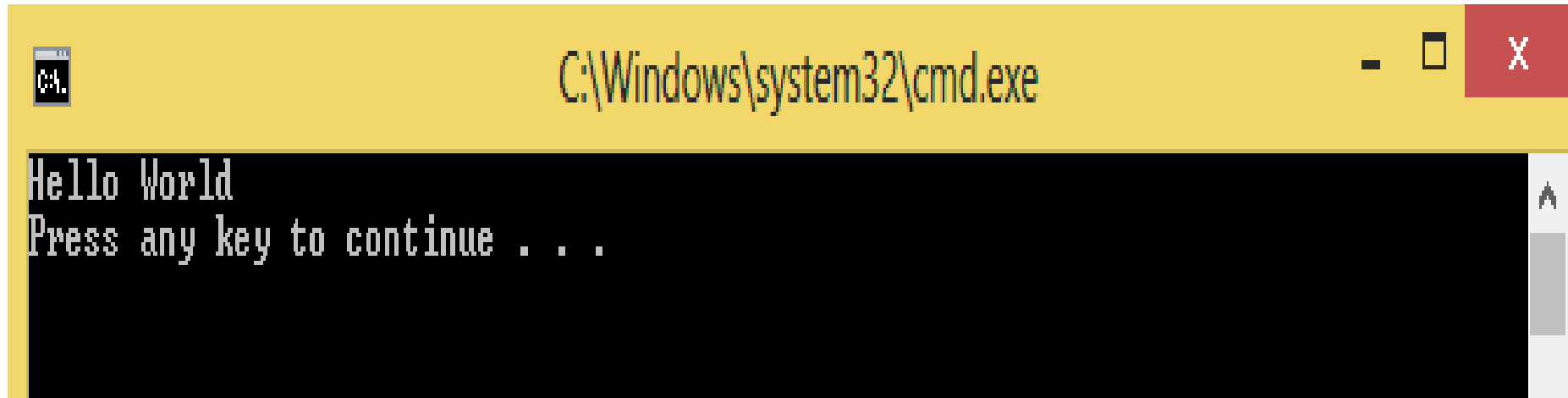
# First program

---

```
#include <iostream>
using namespace std ;
int main ()
{
    cout << "Hello World" ;
    cout << endl ;
    return 0;
}
```

Just print new line

# First program



A screenshot of a Windows command prompt window. The title bar is yellow and contains the text 'C:\Windows\system32\cmd.exe' on the right side. The main area is black with white text that reads 'Hello World' followed by 'Press any key to continue . . .' on the next line. A small icon is visible in the top-left corner of the window.

# First program

## Yes I did it :D



# First program



Not Useful -\_-

# Calculator

Now we need to create  
A simple Calculator that can  
just add two Integers



# Calculator

**Calculate**

$$7 + 5 = ?$$





# Calculator

## **What you need now ?!!**

# Input

To help you .. And to made the program Useful more and more ..

what about **get input** from user

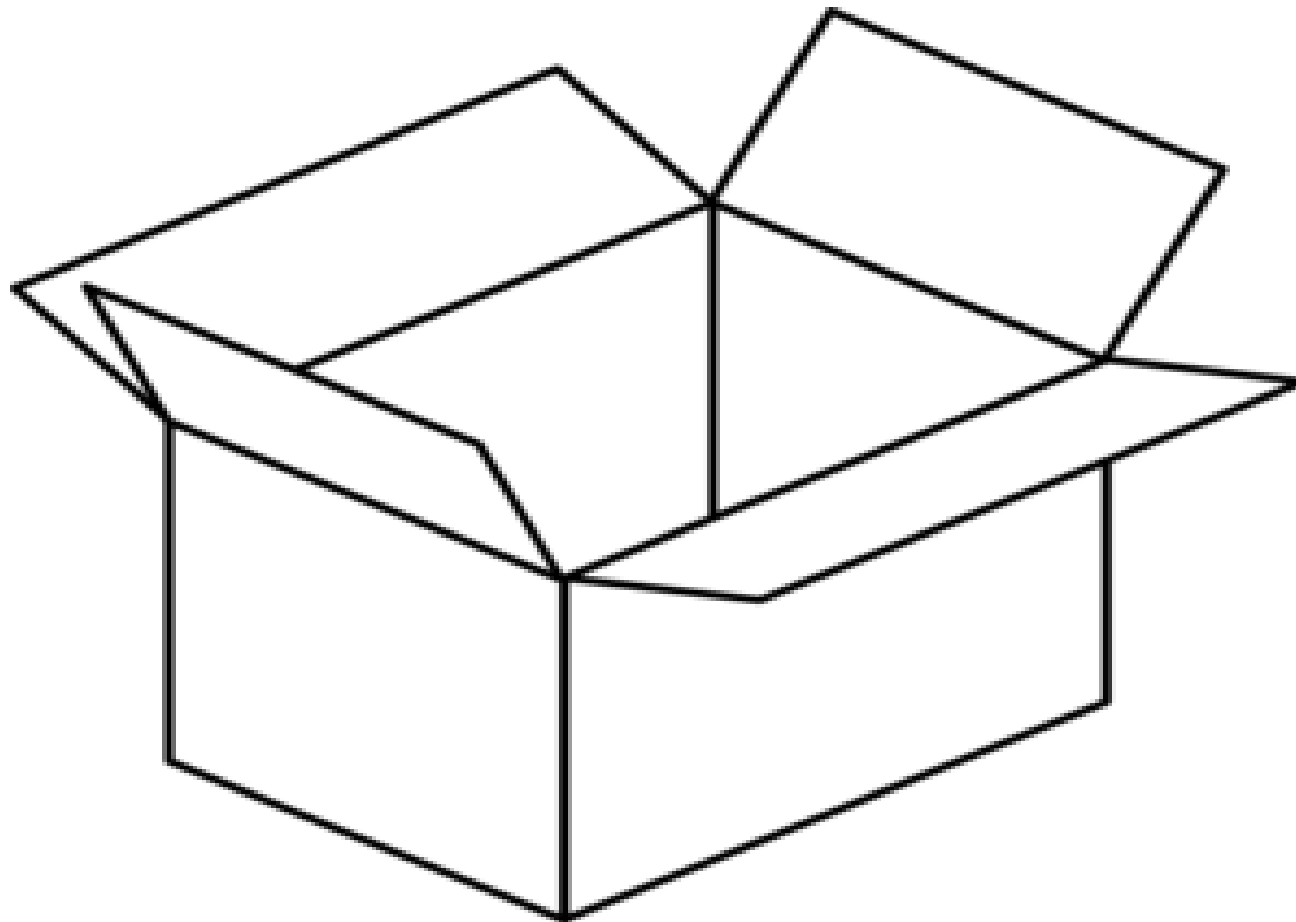
by using

```
cin >>
```

# *Variables*



# Variables



# Variables

Kinds Of data

**Integer number**

**10**

**Real Number**

**7.3**

**Character**

**'A' '?'**

# Variables

bool

- Two values true or false
- True, false

int

- integer number
- 100000

char

- Character
- 'A' '?'

float

- Real number
- 123.4

long long

- long integer
- 9,123,456,789,000,000,000

Short

- Short integer
- 1000

double

- double precision
- 123.4



# Variables

- How to create a VARIABLE !!

***OR***

**datatype** name ;

**datatype** name = initial value ;

# Variables

- **Variable in Memory**





# Variables

- Variable in memory with initial value

```
int x = 10;
```



Any question !!



# Calculator



```
#include <iostream>
using namespace std;
int main()
{
    int number1 ;
    int number2 ;

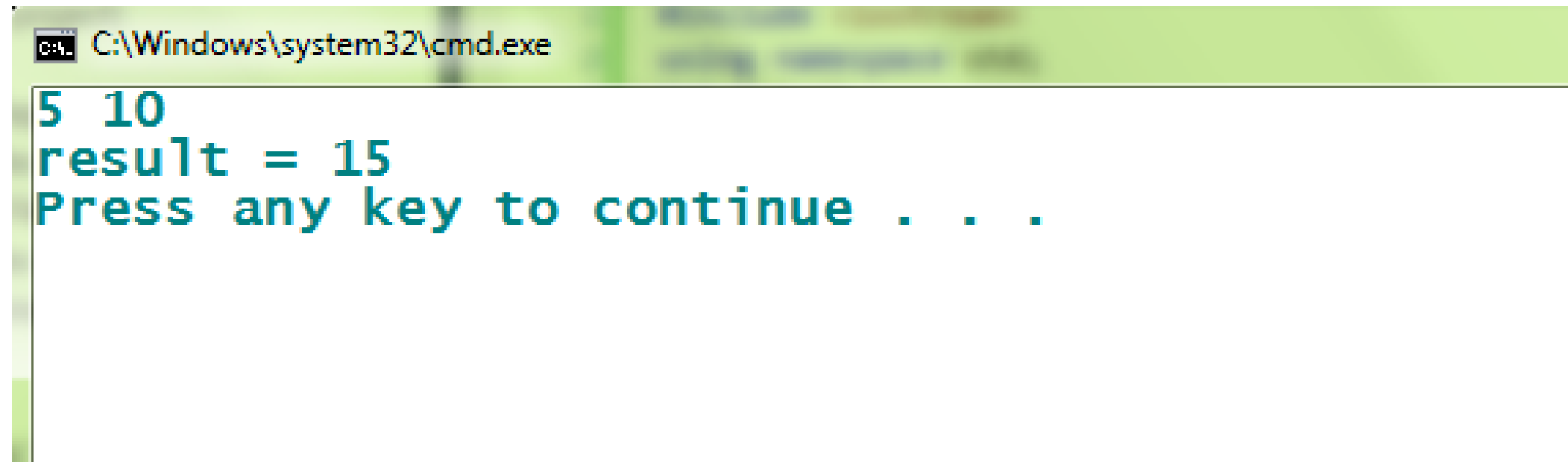
    cin >> number1 ;
    cin >> number2 ;

    int result = number1 + number2 ;

    cout << "result = " ;
    cout << result ;
    cout << endl ;

    return 0;
}
```

# Calculator



```
C:\Windows\system32\cmd.exe  
5 10  
result = 15  
Press any key to continue . . .
```

A screenshot of a Windows command prompt window. The title bar at the top is green and contains the text "C:\Windows\system32\cmd.exe". The command prompt itself has a black background with green text. It displays the numbers "5 10" on the first line, "result = 15" on the second line, and "Press any key to continue . . ." on the third line. The cursor is positioned at the end of the third line.

# Calculator



```
#include <iostream>
using namespace std;
int main()
{
    int number1 , number2 , result ;

    cin >> number1 >> number2 ;

    result = number1 + number2 ;

    cout << "result = " << result << endl ;

    return 0;
}
```

# Second Program

Yes I did it :D





# Operators

# Operators

- **Kinds of Operators**

- Assignment
- Increment and decrement
- Arithmetic
- Logical
- Relational



# Operators

- **Kinds of Operators**

- **Assignment**

=

- Arithmetic
- Increment and decrement
- Logical
- Relational

# Operators

- **Kinds of Operators**

- Assignment

- **Arithmetic**

- Increment and decrement
- Logical
- Relational

=  
**+** , **-** , **\*** , **/** , **%**

# Operators

- **Kinds of Operators**

- Assignment

=

- Arithmetic

+, -, \*, / , %

- **Increment and decrement** ++ , --

- Logical

- Relational

# Conditions

# Conditions

**if** (Some Condition)

{

// a sequence of statement

}

**a Boolean expression**

# Conditions

**If** (Some condition)

```
{  
    // Sequence of statements  
}
```

**else if** (another Condition )

```
{  
    //Sequence of Statement  
}
```

**else**

```
{  
    // Sequence of Statement  
}
```

# Conditions

- What if I want to add more than Condition :/

# Operators

- **Kinds of Operators**

- Assignment
- Arithmetic
- Increment and decrement

- **Logical**

- Relational

=

+, -, \*, / , %

++, --

**&& , || , !**



# Operators

- **Kinds of Operators**

- Assignment =
- Arithmetic + , - , \* , / , %
- Increment and decrement ++ , --
- Logical && , || , !
- **Relational** < , > , <= , >= , ==

# Operators

**=** **=** **VS** **=**



# Conditions

## •Sections !!



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

    if(You_Section_Number >= 10)
    {
        cout << "Reise Your hand" ;
    }
    else if (You_Section_Number <= 10)
    {
        cout << "Stand Up";
    }
    return 0;
}
```

# Conditions

- What about Section **10**



# Conditions

## {Scoping}

Any variable defined in { } can't be used **Outside** them

Any question !!



# Conditions



**Let's Code :D**

# Conditions

- **Odd or Even**

Nagate is a teacher has N number of Todos , their is two students in his class he wants to know whether he can divide the Todos between them equally or not

input : the input will contain one integer ( $1 \leq N \leq 10000$ ).

output : if it is odd print a line containing "NO" else print "Yes".



# Conditions

- **Odd or Even**

Nagate is a teacher has **N number** of Todos , their is two students in his class he wants to know whether he can divide the Todos between them equally or not

input : the input will contain one integer ( $-10000 \leq N \leq 10000$ ).

output : if it is odd print a line containing "NO" else print "Yes".

# Conditions

- **Odd or Even**

Nagate is a teacher has **N number** of Todos , their is two students in his class he wants to know whether he can divide the Todos between them equally or not

input : the input will contain one integer  **$(-10000 \leq N \leq 10000)$** .

output : if it is odd print a line containing "NO" else print "Yes".

# Conditions

- **Odd or Even**

**Sample input #1 :**

4

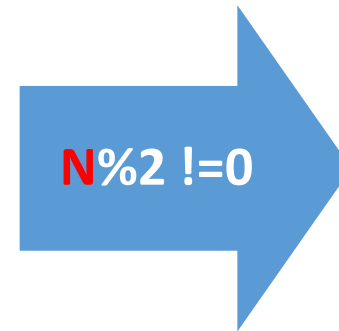
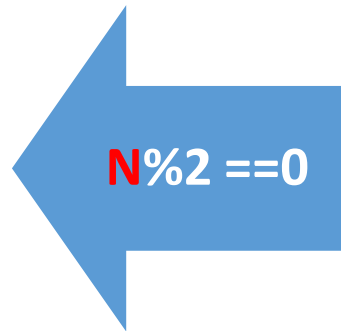
**Sample output #1 :**

Yes

# Conditions

- **Odd or Even**

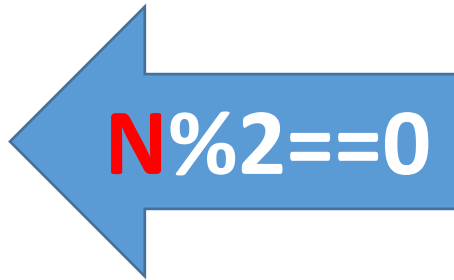
Get **N** from user



# Conditions

- Odd or Even

$$N = 4$$


$$N \% 2 == 0$$



# Conditions

- Odd or Even

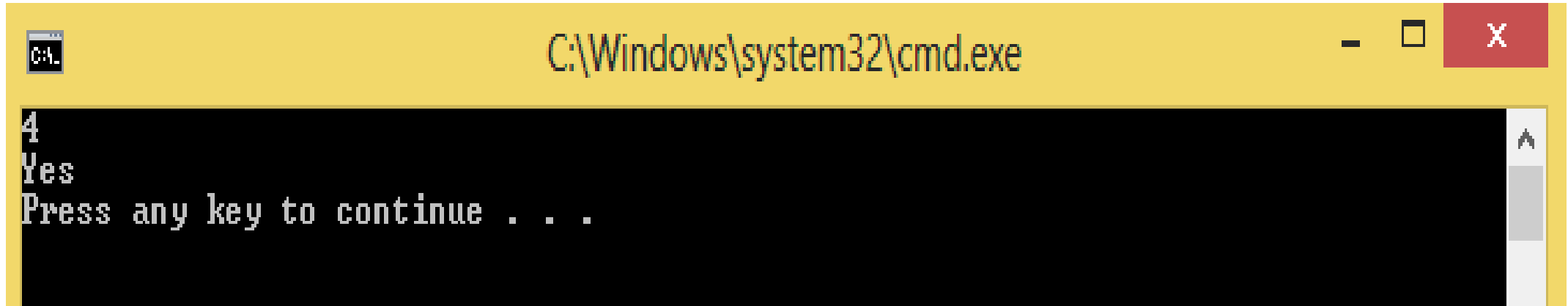
```
#include <iostream>
using namespace std;
int main()
{
    int number ;
    cin >> number ;

    if(number % 2 == 0)
    {
        cout << "Yes" << endl ;
    }
    else
    {
        cout << "No" << endl ;
    }

    return 0;
}
```

# Conditions

- **Odd or Even**



```
C:\Windows\system32\cmd.exe  
4  
Yes  
Press any key to continue . . .
```

Any question !!







# It's Your Turn!

# Conditions

- **Max of 3 Integers**

Given three integers you are to find the maximum element among the three.

input:

the input will contain three integers

$a, b, c$  ( $-10000 \leq a, b, c \leq 10000$ ).

output :

print "the maximum is : " followed by the maximum element on a line per itself.

# Conditions

- **Max of 3 Integers**

Given three integers you are to find the maximum element among the three.

input:

the input will contain three integers

$a, b, c$  ( $-10000 \leq a, b, c \leq 10000$ ).

output :

print "the maximum is : " followed by the maximum element on a line per itself.



# Conditions

- **Max of 3 Integers**

**Sample input :**

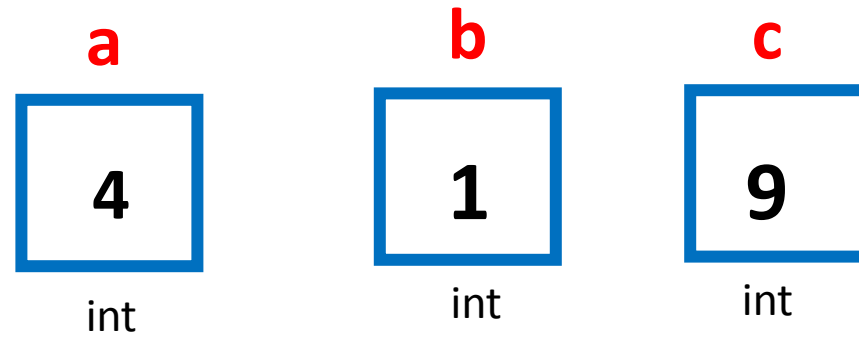
4 1 9

**Sample output :**

the maximum is : 9

# Conditions

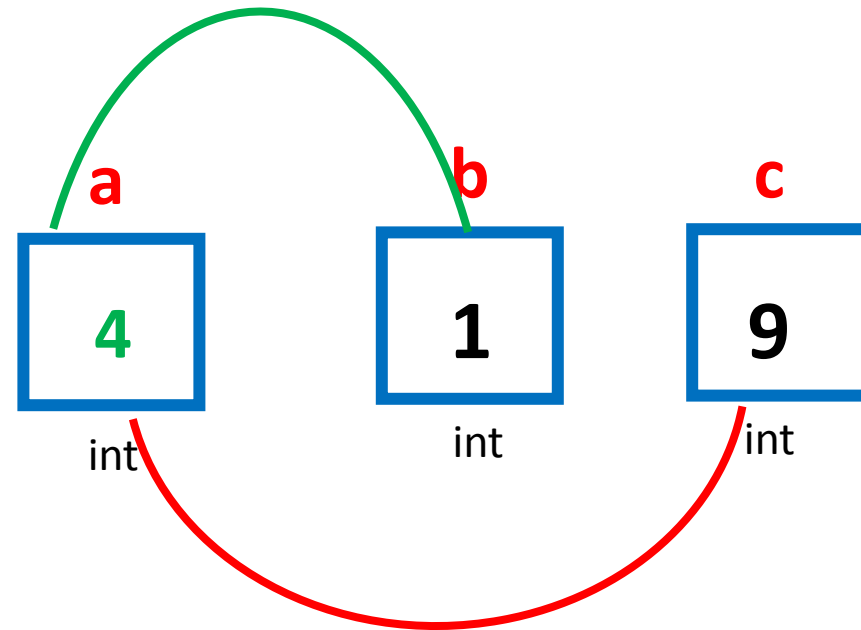
- **Max of 3 Integers**



# Conditions

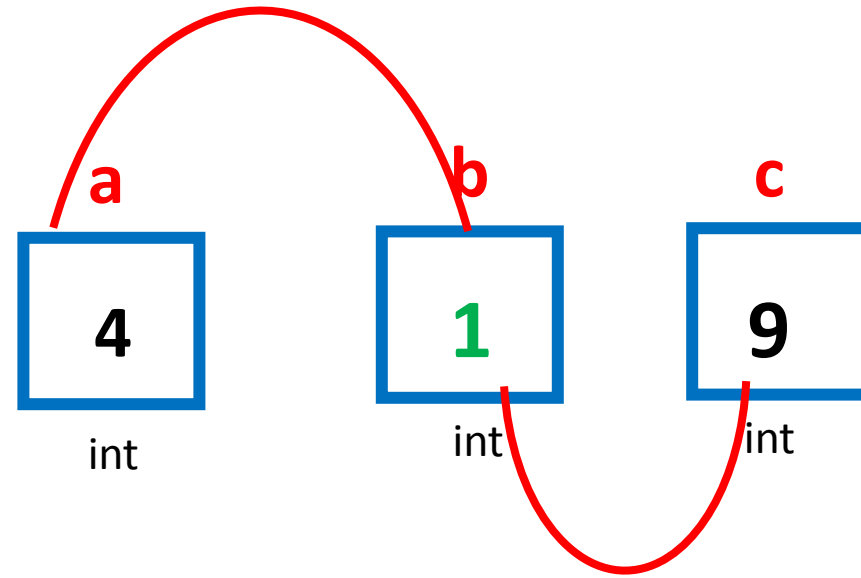
- **Max of 3 Integers**

**False**



# Conditions

- **Max of 3 Integers**

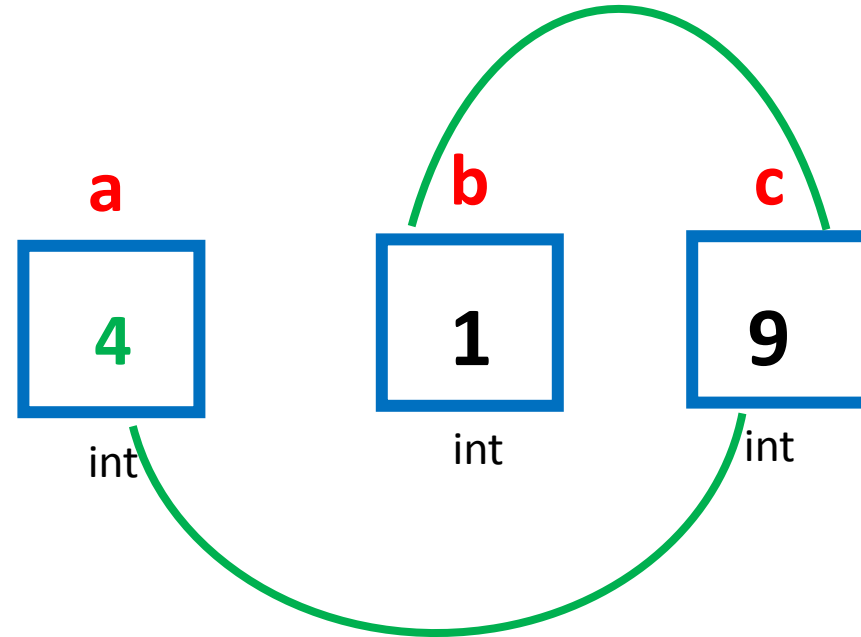


**False**

# Conditions

- **Max of 3 Integers**

True





# Conditions

- **Max of 3 Integers**

```
#include<iostream>
using namespace std;

int main()
{
    int x,y,z;
    cin>>x>>y>>z;
    if(x>=y && x>=z)
        cout<<"the maximum is : "<<x<<endl;
    else if(y>=x&&y>=z)
        cout<<"the maximum is : "<<y<<endl;
    else
        cout<<"the maximum is : "<<z<<endl;
    return 0;
}
```

# Common mistake !!

```
int main()  
{  
    cin >> x;  
    return 0;  
}
```

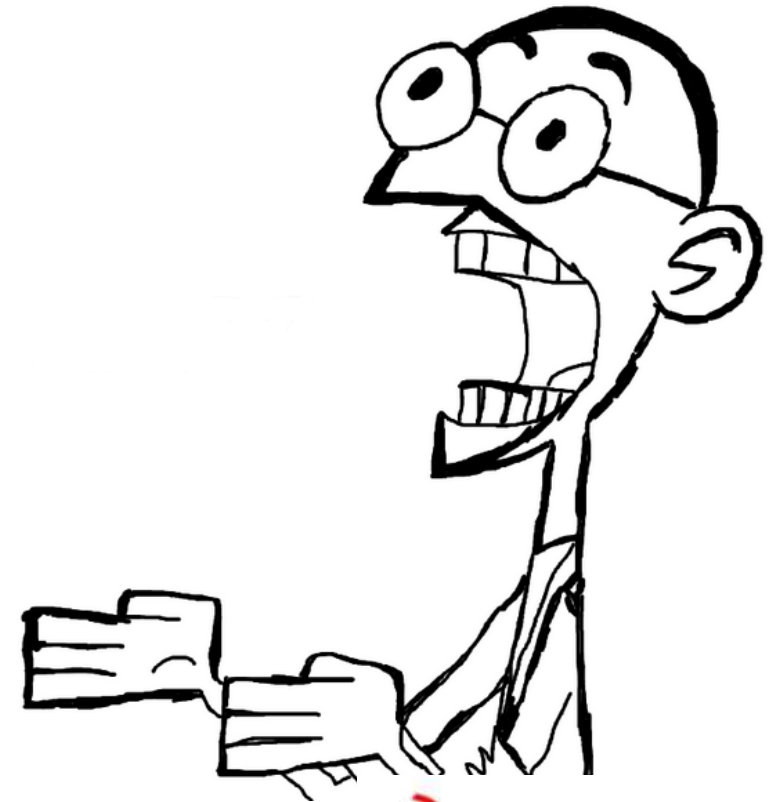


# Common Mistake !!

```
int main()  
{  
    cin >> x;  
    return 0;  
}
```



What's x ?!!



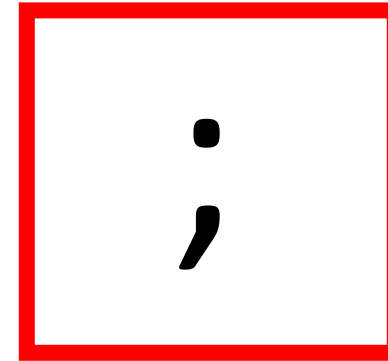
# Common Mistake !!

```
☐ int main()  
{  
    int x;  
  
    cin >> x;  
  
    return 0;  
}
```



Common Mistake !!

Missed



Any question !!



# Break





# Loops



MY HOMEWORK WAS NOT STOLEN  
BY A ONE-ARMED MAN  
MY HOMEWORK WAS NOT STOLEN  
BY A ONE-ARMED MAN  
MY HOMEWORK WAS NOT STOLEN  
BY A ONE-ARMED MAN  
MY HOMEWORK WAS NOT  
BY A ONE-ARMED MAN  
MY HOMEWORK WAS NOT  
BY A ONE-ARMED MAN  
MY HOMEWORK WAS  
BY A ONE-ARMED MAN

A yellow cartoon character with a large, spiky head, wearing an orange shirt, is shown from the chest up, looking towards the left. He has a small, mischievous smile and is holding a small object in his right hand. The background is black.

I need some help



# What we did till now ?

- Variables
- Operators
- Conditions
- Break

## • Loops



- Break
- Arrays



# loops



# What is the mean of Loop ?

- A block of code (set of Operations) **repeats** itself for a **number of times** or as long **as a condition is valid**

# Types of loops

- While Loop
- do While Loop
- For Loop



# while loop

**While** (my Cup is filled)

{

// i will Drink some coffee  
// i will eat peace of cake

}





# Loops

- **While**

**While** (Condition)

{

do Operation 1 ;

do Operation 2 ;

..

do Operation N ;

}



Boolean expression

Any question !!



# Do while

```
do  
{
```

```
    // i will drink some of coffee  
    // i will eat peace of cake
```

```
} while (my cup is filled) ;
```

```
do {  
    Operation 1;  
    Operation 2 ;  
    ..  
    Operation N ;  
} while (Some condition) ;
```



Boolean expression

**do** {

Operation 1;

Operation 2 ;

..

Operation N ;

} **while** (Some condition) ;

Don't miss it

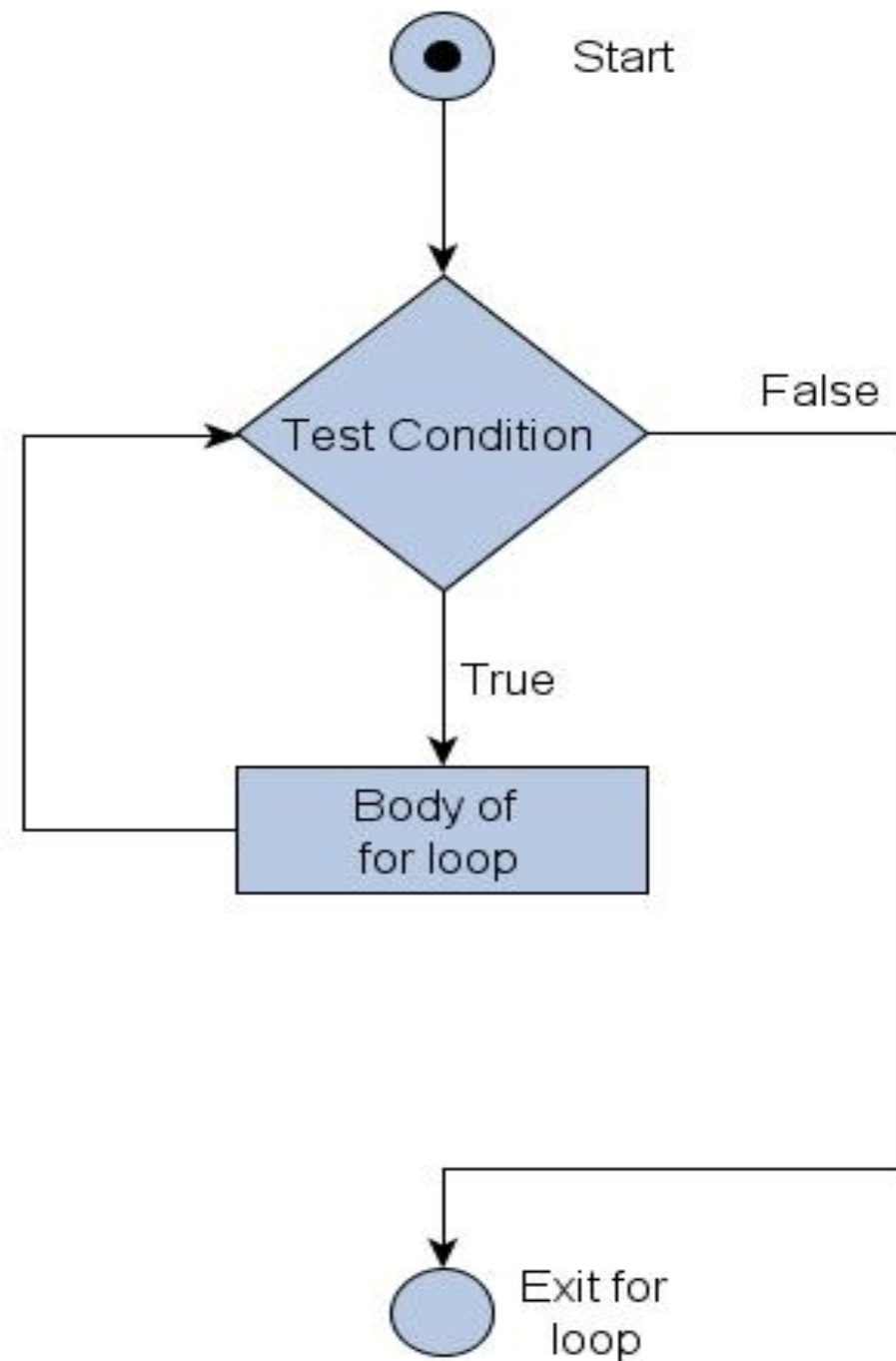
Any question !!



# For



- For



# Loops

- **For**

**for**(Initialization ; Condition ; Iteration step )

{

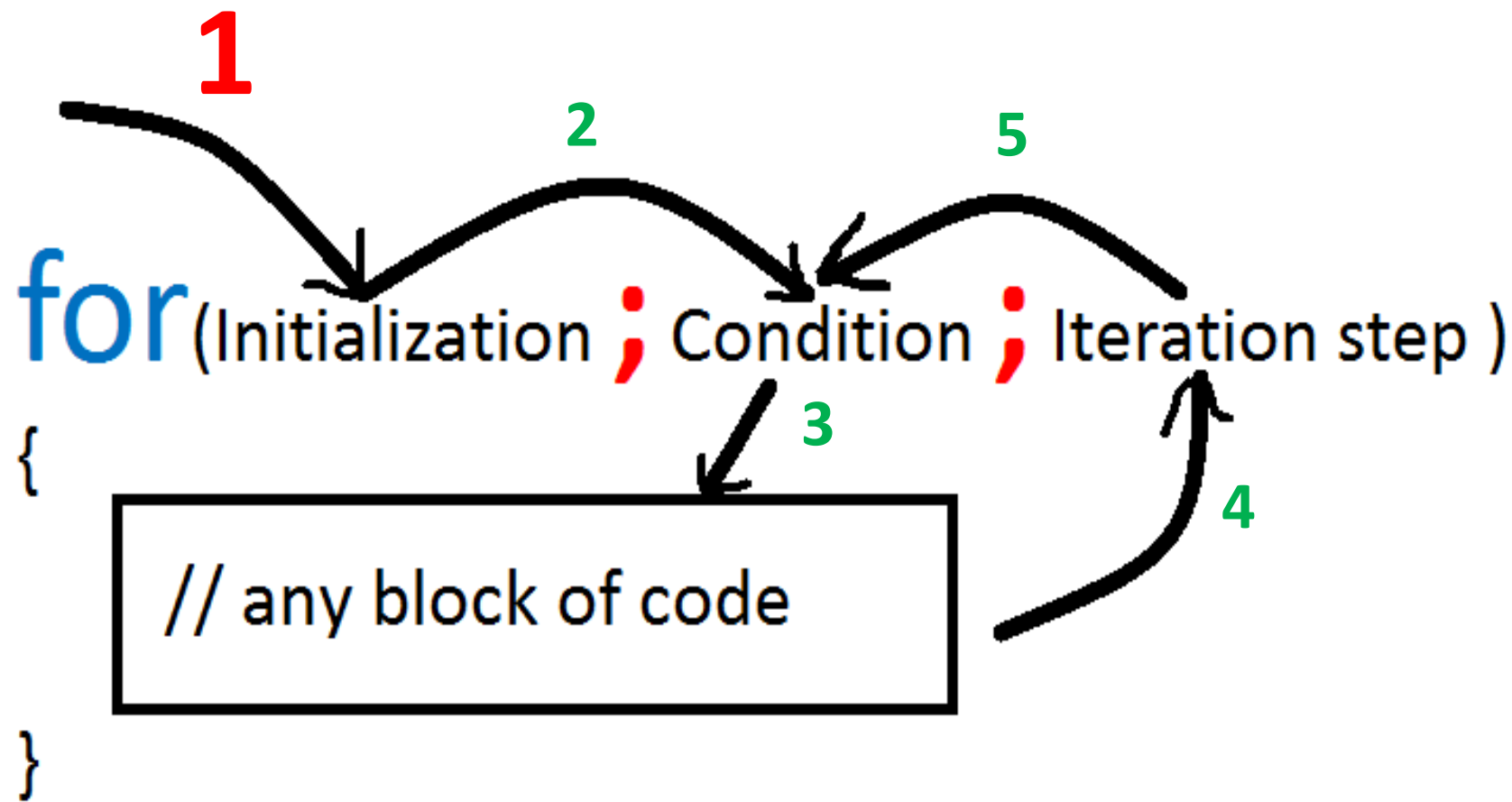
do Operation 1 ;

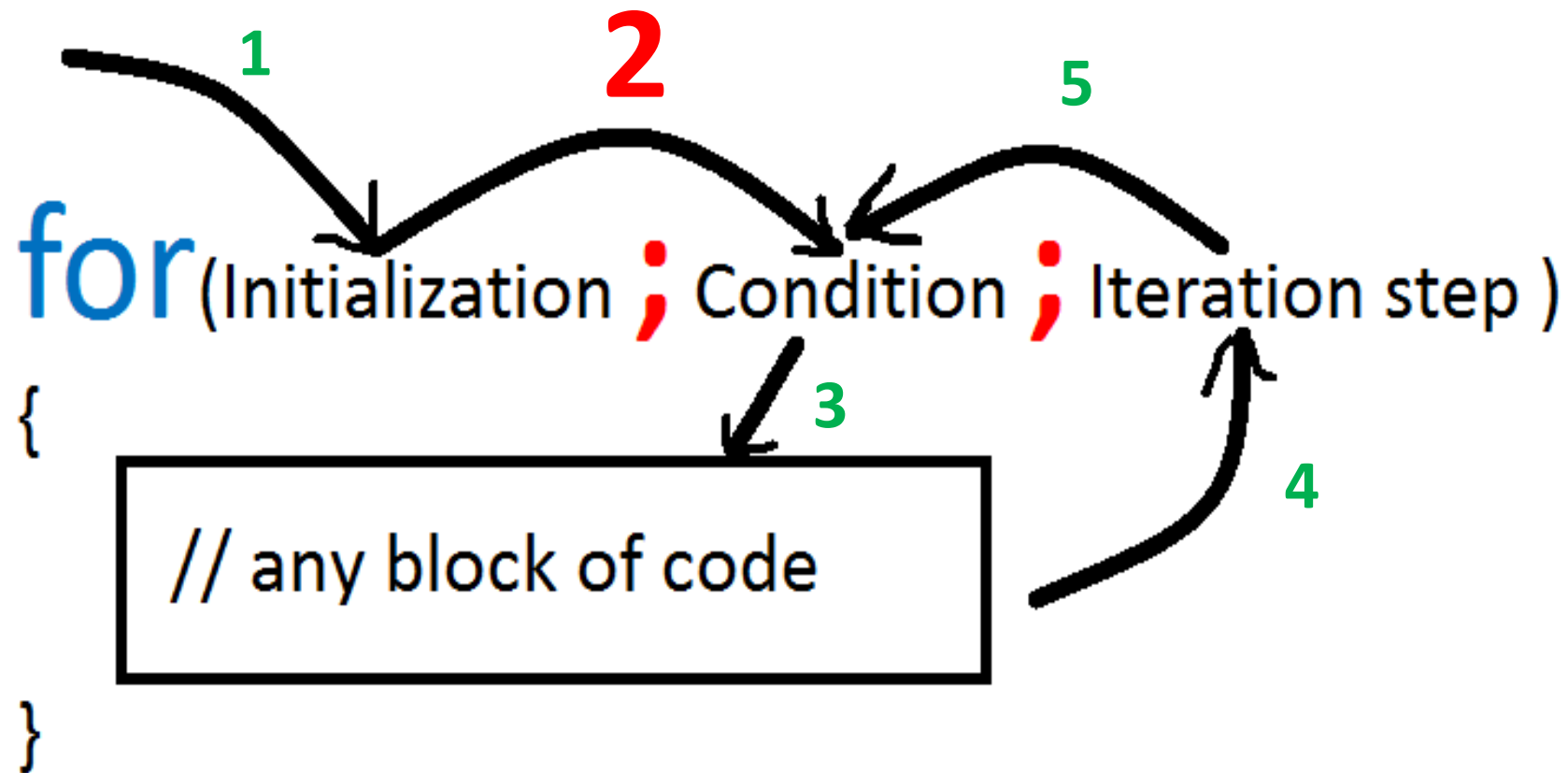
do Operation 2 ;

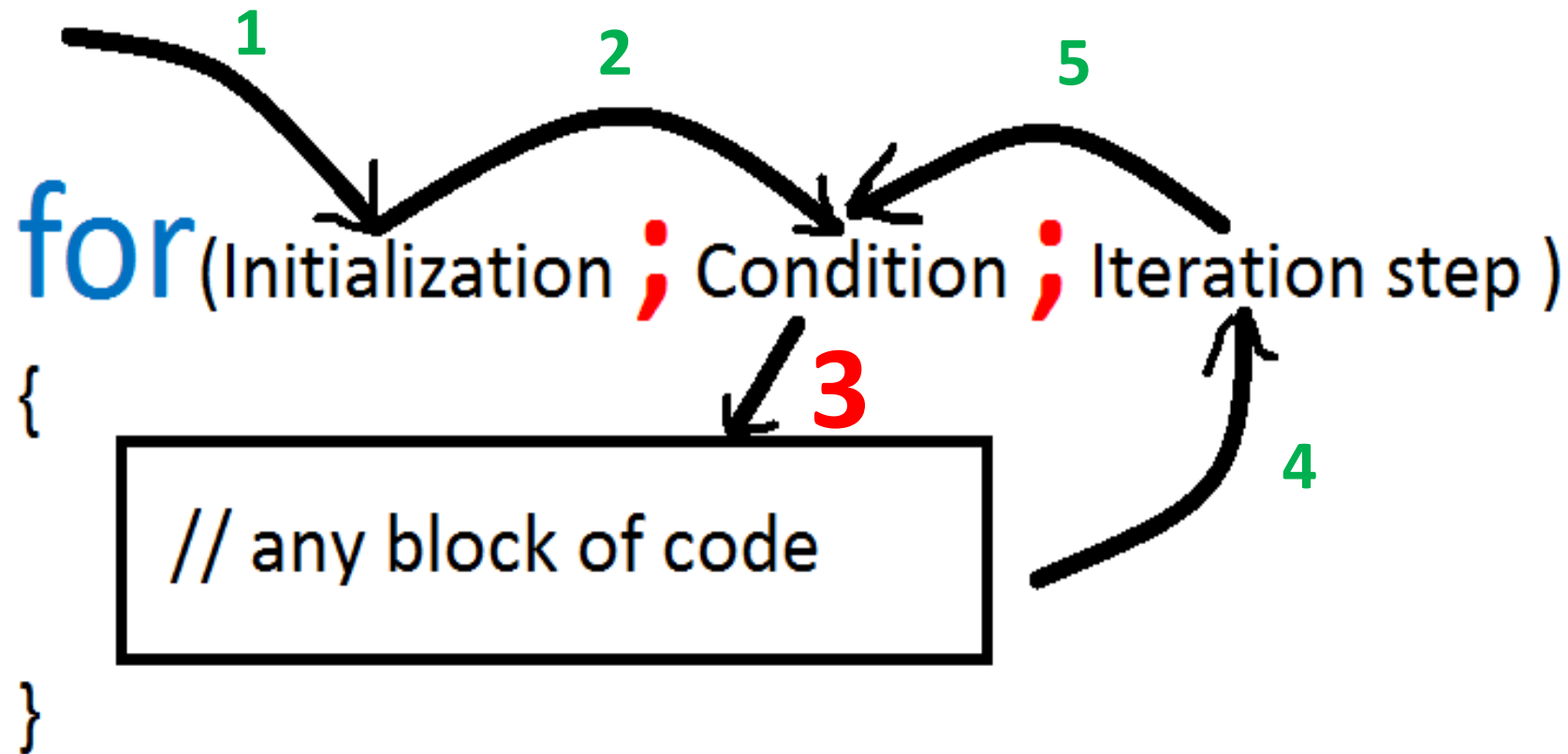
..

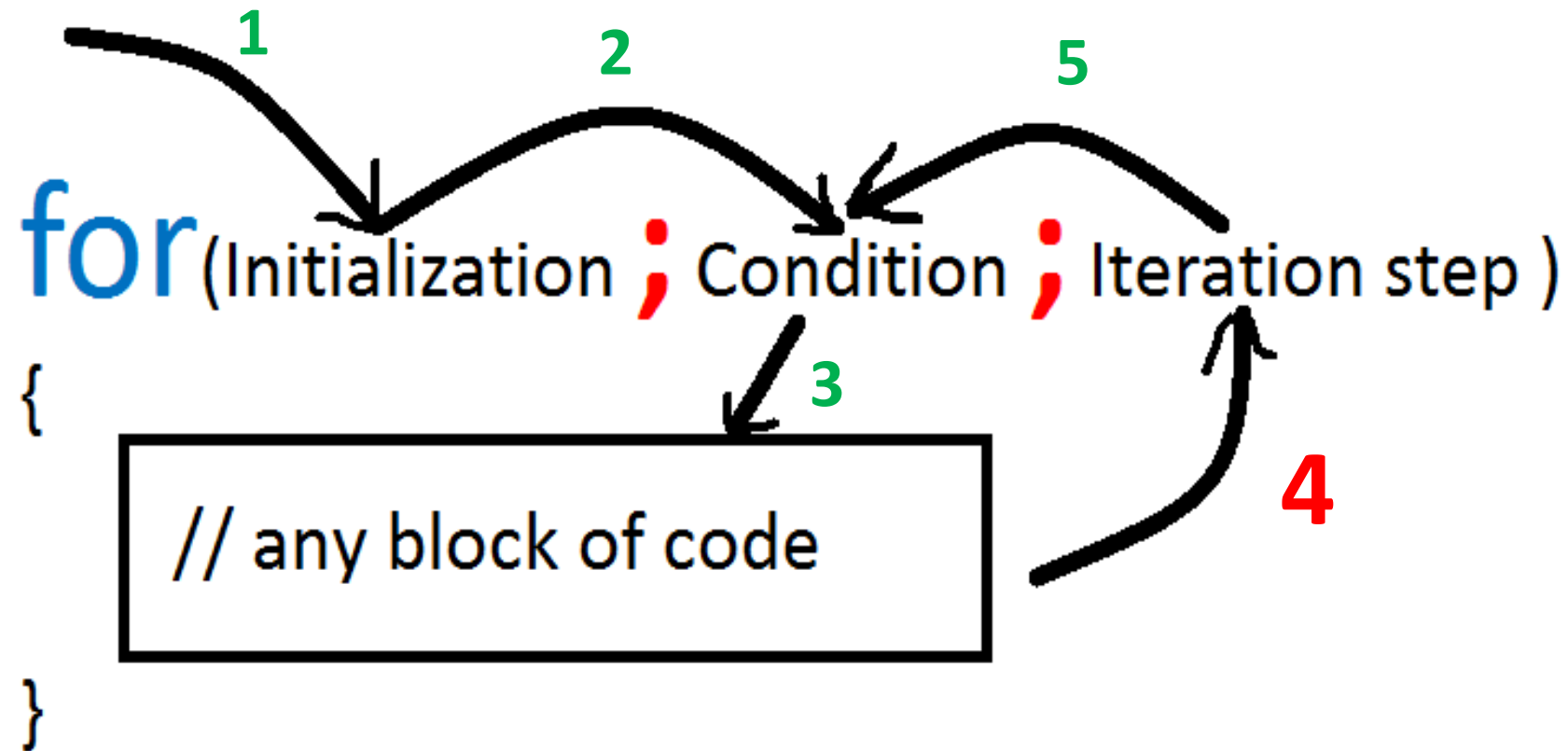
do Operation N;

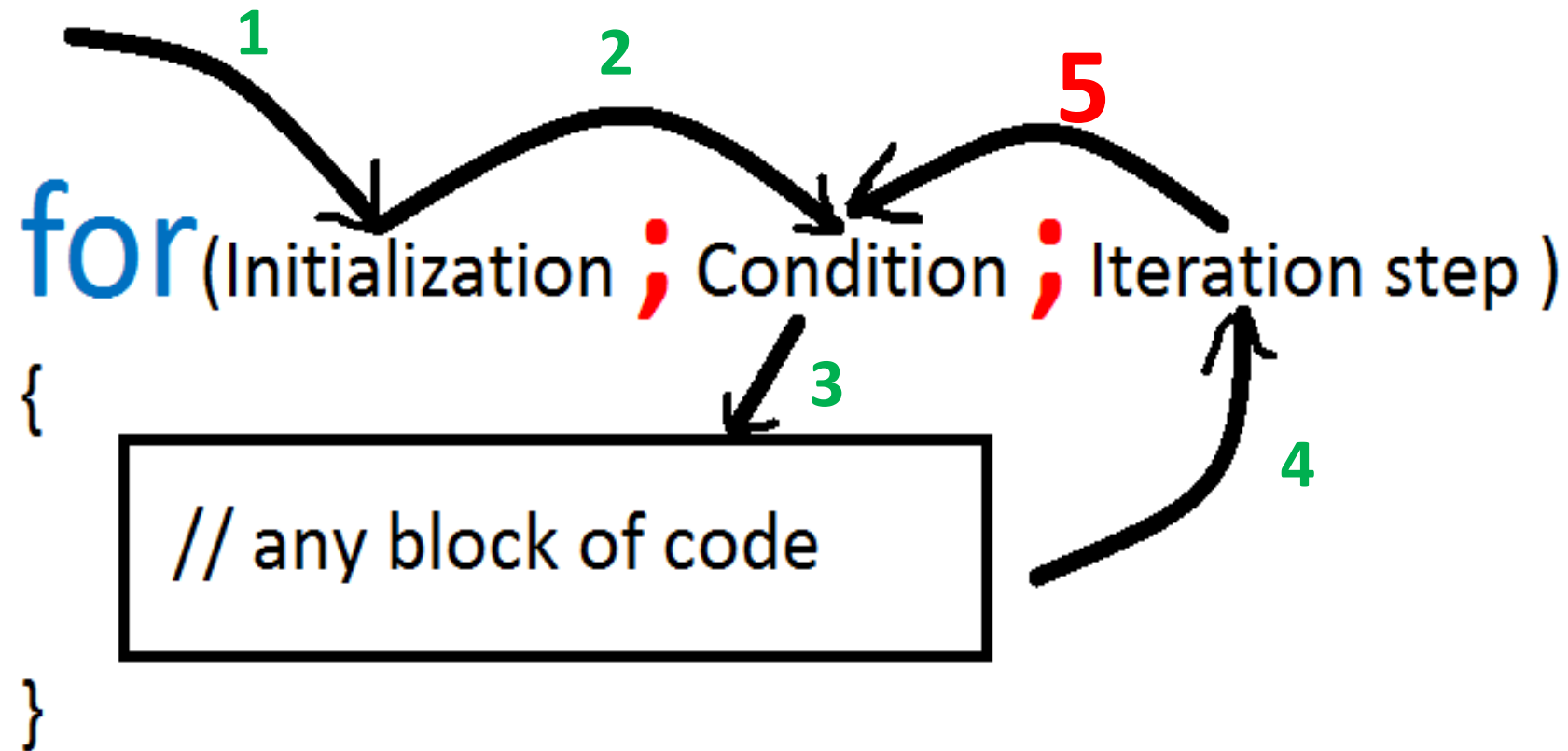
}

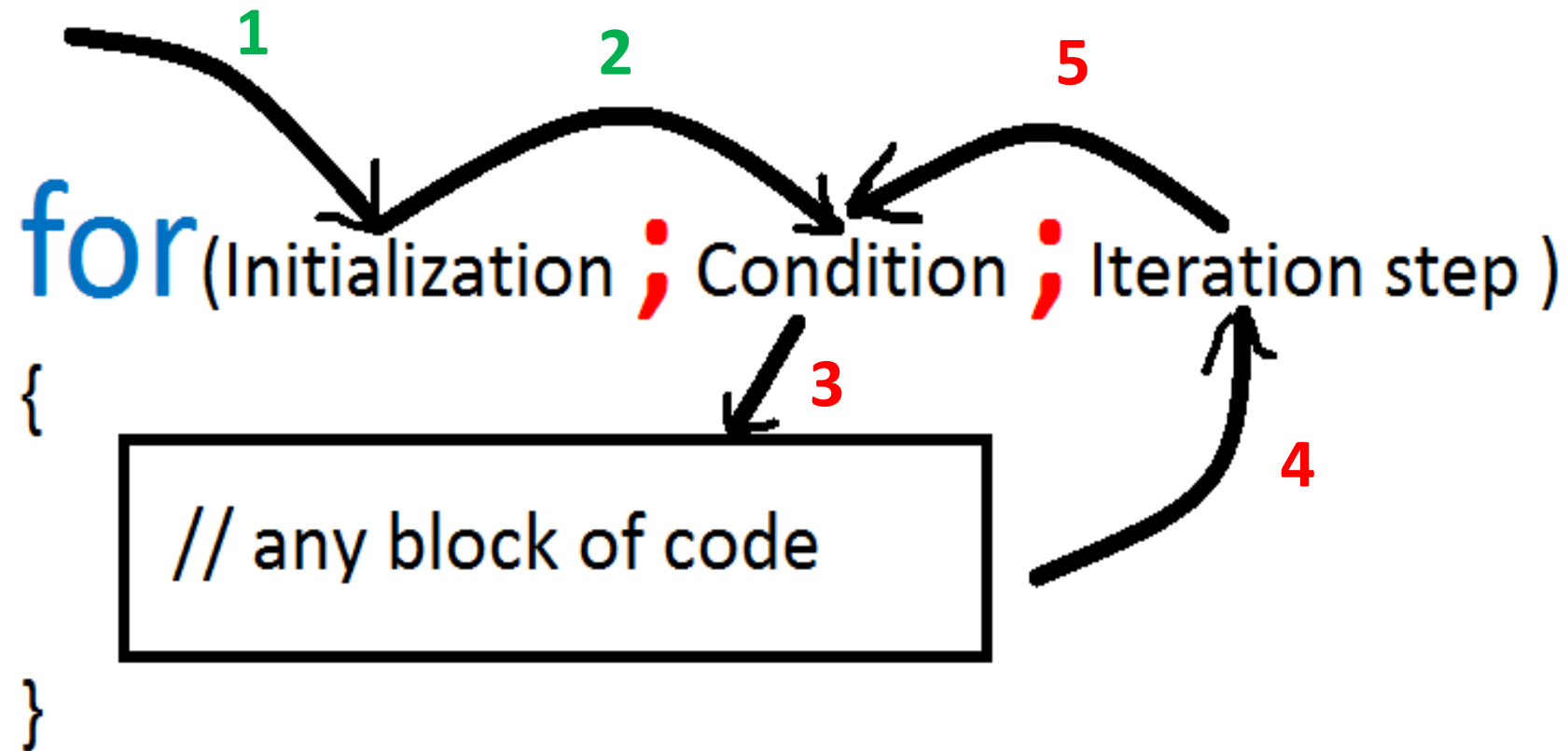














# •For

for loop :

- depends on **fixed number of iterations**

```
for (int f = 1 ; f < 15; f ++ )  
{  
    cout << f << endl;  
}
```

guess !



```
for ( ; ; )  
{  
    cout << "Hello world " << endl;  
}
```

**guess !**



Any question !!



# • Factorial

- Given a number  $n$  you have to calculate the factorial of  $n$

## input :

input starts with a  $t$  ( $t < 100$ ) denotes the number of test cases  
then each test case start with the number  $n$  ( $n \leq 15$ )

## output :

For each test case, output one line in the format “**Case x a**”  
where  $x$  is the case number and  $a$  is the factorial.



# Loops

- **Factorial**

- Given a number  $n$  you have to calculate the factorial of  $n$

input :

input starts with a  $t$  ( $t < 100$ ) denotes the number of test cases  
then each test case start with the number  $n$  ( $n \leq 15$ )

output :

For each test case, output one line in the format “**Case  $x$  a**”  
where  $x$  is the case number and  $a$  is the factorial.



# Loops

- **Factorial**

**Sample input :**

2

5

10

**Sample output :**

Case 1 : 120

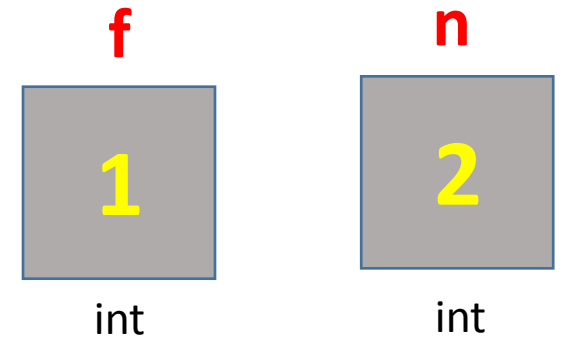
Case 2 : 3628800

# Loops

- **Factorial**

Before for

```
for (int i=n ; i>= 1 ; i--)  
{  
    f = f*i ;  
}
```





# Loops

- **Factorial**



```
for (int i=n ; i>= 1 ; i--)  
{  
    f = f*i ;  
}
```

**1** initialization

**f**



int

**i**



int

**n**



int

# Loops

- **Factorial**

**2** Check

`for (int i=n ; i>= 1 ; i--)`  
{  
    `f = f*i ;`  
}

**f**  
1  
int

**i**  
2  
int

**n**  
2  
int

# Loops

- **Factorial**

**3**      **do**

```
for (int i=n ; i>= 1 ; i--)  
{  
    f = f*i ;  
}
```



# Loops

- **Factorial**

**4** Iteration

```
for (int i=n ; i>= 1 ; i--)  
{  
    f = f*i ;  
}
```

**f**  
2  
int

**i**  
1  
int

**n**  
2  
int

# Loops

- **Factorial**

Check , do , iteration ..

```
for (int i=n ; i>=1 ; i--)  
{  
    f= f*i;  
}
```

**f**  
2  
int

**i**  
1  
int

**n**  
2  
int

# Loops

- **Factorial**

Condition == false

↓

```
for ( int i = n ; i >= 1 ; i-- )  
{  
    f=f*i ;  
}
```

**f**

2

int

↓

**i**

0

int

**n**

2

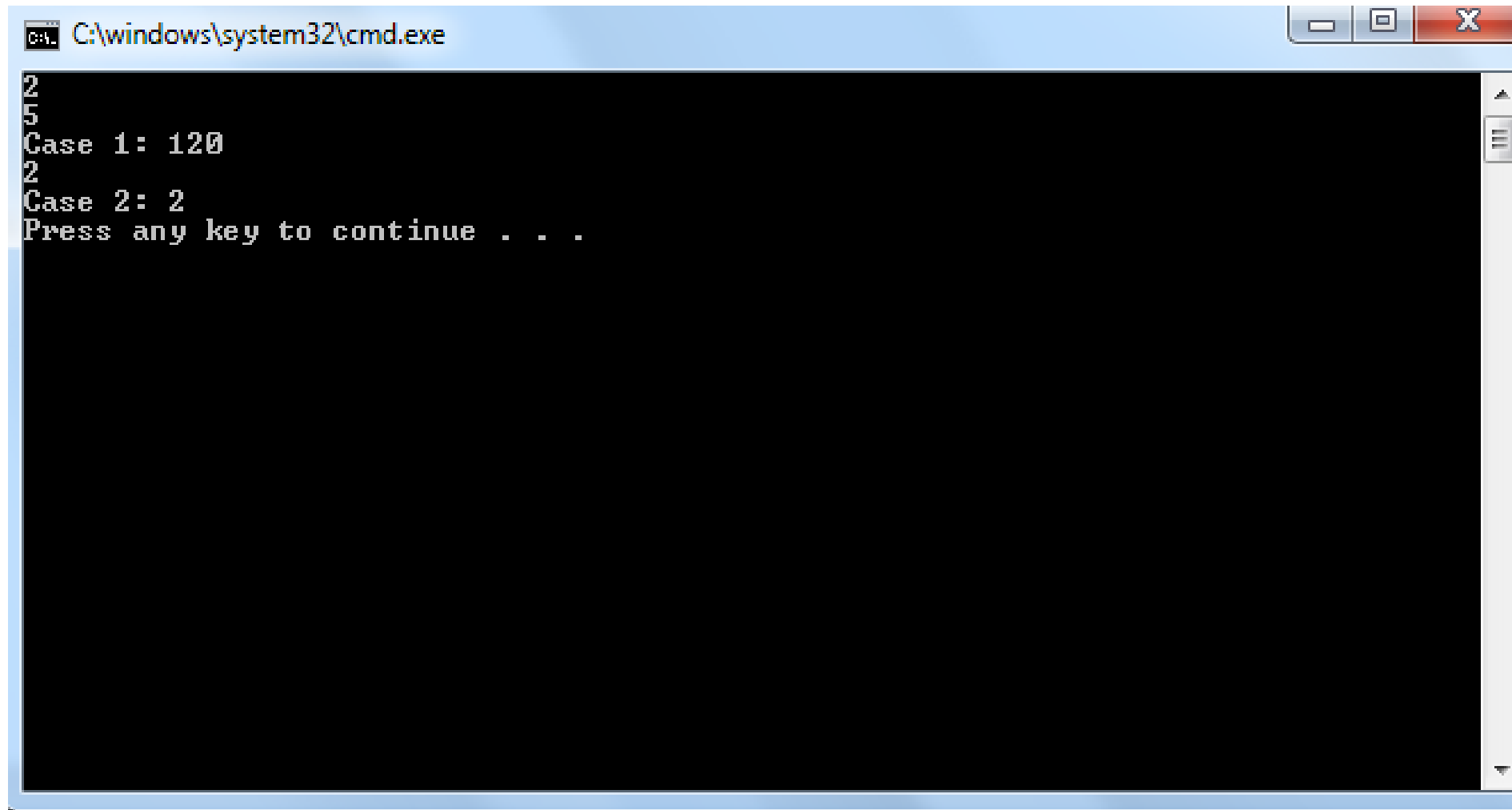
int

Now .. Out :D

- **Factorial**

**Let's code**

- Factorial



A screenshot of a Windows command prompt window titled "C:\windows\system32\cmd.exe". The window has a black background with white text. The text displayed is as follows:

```
2
5
Case 1: 120
2
Case 2: 2
Press any key to continue . . .
```





# It's Your Turn!

# Loops

- **Easy  $3n+1$**

- 1 - given a number  $n$
- 2 - if  $n = 1$  then STOP
- 3 - if  $n$  is odd then  $n=3n+1$
- 4 - else  $n=n/2$ ;
- 5 - go to step 2

Given the input 22, the following sequence of numbers will be computed 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1 , you have to calculate the sum of these numbers .

## **Input :**

The input will contain several test cases each test case starts with one integer  $n$  ( $0 < n \leq 100000$ ).input terminates with end of file .

## **Output :**

For each line of input, print "the sum is : " followed by the sum of the elements of the sequence on a line per itself.



- **Easy  $3n+1$**

- Sample input :

- 22

- 10

- Sample output :

- the sum is : 281

- the sum is : 46

- **Easy  $3n+1$**

let's code

```
C:\windows\system32\cmd.exe
22
The sum is 281
10
The sum is 46
```

Any question !!



# Small game with stars

- Star
- Line
- Square
- Rectangle
- Traingle



# Loops

- One star





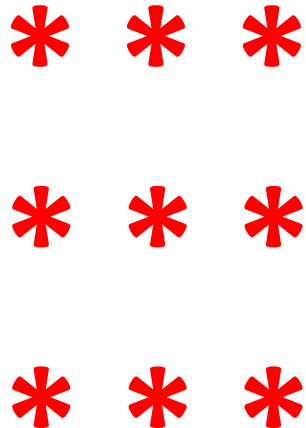
# Loops

- Line ..

\*\*\*\*

# Loops

- Square



# Loops

- Rectangle

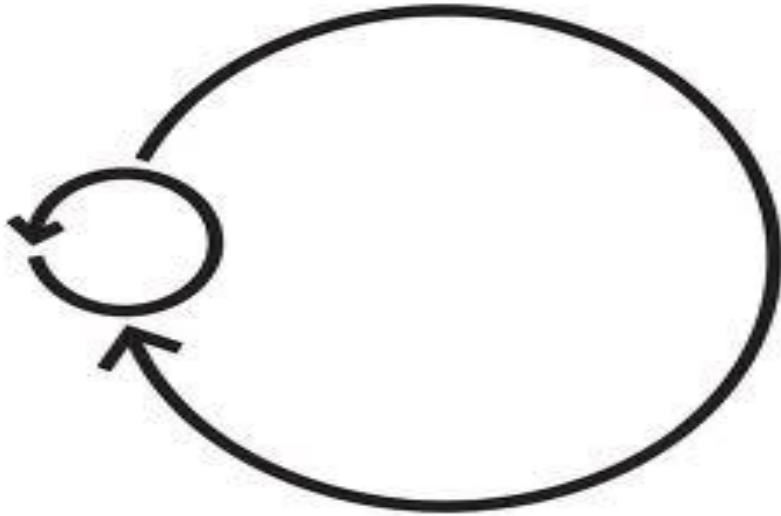
\* \* \* \* \*

\* \* \* \* \*

\* \* \* \* \*

# Loops

- **Nested loops**



# Loops

- Nested loops

```
for (initialization ; condition ; iteration)
```

```
{
```

```
    for ( initialization ; condition ; iteration)
```

```
    {
```

```
        .....
```

```
    }
```

```
}
```





# It's Your Turn!

# Loops

- Nested loops

- Triangle ..

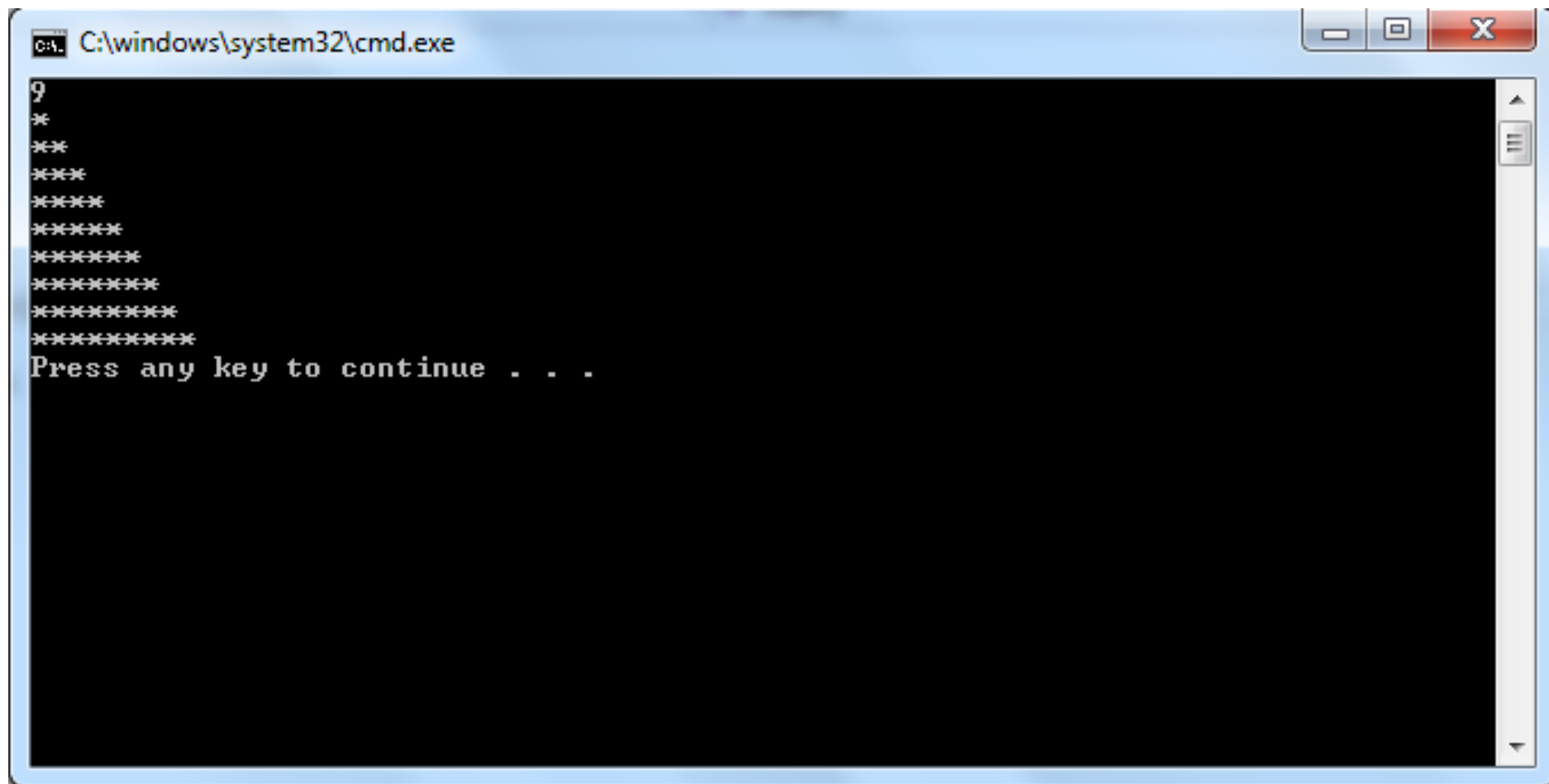
\*

\* \*

\* \* \*

\* \* \* \*

# • Triangle



```
C:\windows\system32\cmd.exe
9
*
**
***
****
*****
*****
*****
*****
*****
*****
Press any key to continue . . .
```



```
C:\windows\system32\cmd.exe

9
*      i = 1      S = 1
**     i = 2      S = 2
***    i = 3      S = 3
****   i = 4      S = 4
***** i = 5      S = 5
***** i = 6      S = 6
***** i = 7      S = 7
***** i = 8      S = 8
***** i = 9      S = 9
Press any key to continue . . .
```

- **Triangle**

let's code

Any question !!



# Jump statements :

- The **break** statement
- The **continue** statement

- **The break statement**

```
for( int i = 0 ; i < 8 ; i++ )  
{  
    if( i == 5 )  
        break;  
  
    cout << i << endl;  
}
```

# The continue statement

```
for( int i = 0 ; i < 8 ; i++ )  
{  
    if( i == 5 )  
        continue;  
  
    cout << i << endl;  
  
}
```

Any question !!



# Break





Array [ ]



# Array [ ]

## 1 Variable in Memory

```
int x ;
```



# Array [ ]

10 variables

```
int x0 ;
```

```
int x1 ;
```

```
int x2 ;
```

```
int x3 ;
```

```
..
```

```
int x9;
```



Array [ ]

**100 Variable ..**



int Mohammed;	int Abdullah;	int Amien;	int Mohamed;
int mina;	int Hatem;	int Bora;	int Norhan;
int menna;	int Awad;	int Mido;	int abdElwahab;
int chrisitna;	int Janna;	int nour;	int Halim;
int ahmed;	int Nahed;	int mansour;	int Hani;
int Hassan;	int Joe;	int morsy;	int Fakhry;
int joseph;	int Matt;	int sakka;	int fawzy;
int Anas;	int jasmine;	int ramzy;	int safi;
int mark;	int Sara;	int gimmy;	int Nora;
int Ismail;	int Koko;	int foo;	int Nawal;
int Boda;	int Saad;	int toot;	int Omar;
int Romy;	int Matt;	int Mounira;	int hussin;
int Mahdi;	int Hamada;	int Mounir;	int essam;
int Nagaty;	int Nada;	int Waheda;	int mahmoud;
int Ashraf;	int Jasmine;	int Wahed;	int afify;
int Abdo;	int harankash;	int Samira;	int AboMuslim;
int Ibrahim;	int sylvester;	int Samir;	int Sabry;
int Halla;	int erd;	int Hesham;	int sebrs;
int Toka;	int Soma;	int Ali;	int tenno;
int manar;	int Basma;	int mosherf;	int AboZaid;

Array [ ]

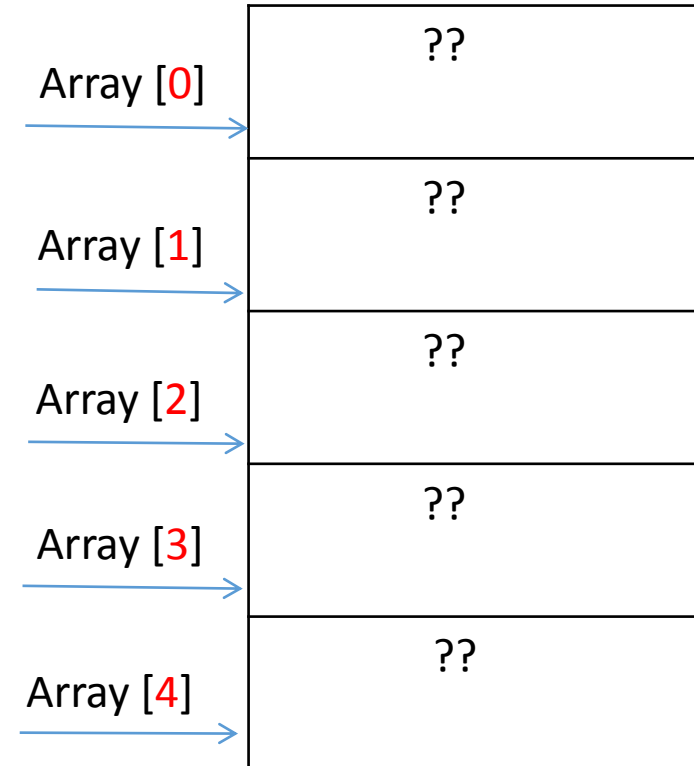
**1000000 Variable ...**



# Array [ ]

Firstly , you should know  
that **Array** is a  
**block of contiguous values of  
the same type**

```
int array [5] ;
```

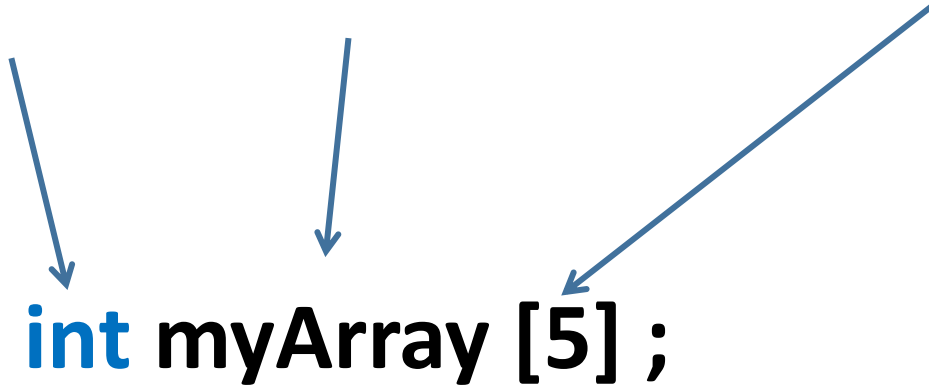


# Array [ ]

How to declare Array ?

**DataType** **ArrayName** [**Size**] ;

**int** **myArray** [**5**] ;





# Array [ ]

## How to declare Array ?

- **Size must be constant**

```
int arr[4];
```

```
float FloatNums[4];
```

```
char Name[4];
```

Memory
??
??
??
??

# Array [ ]

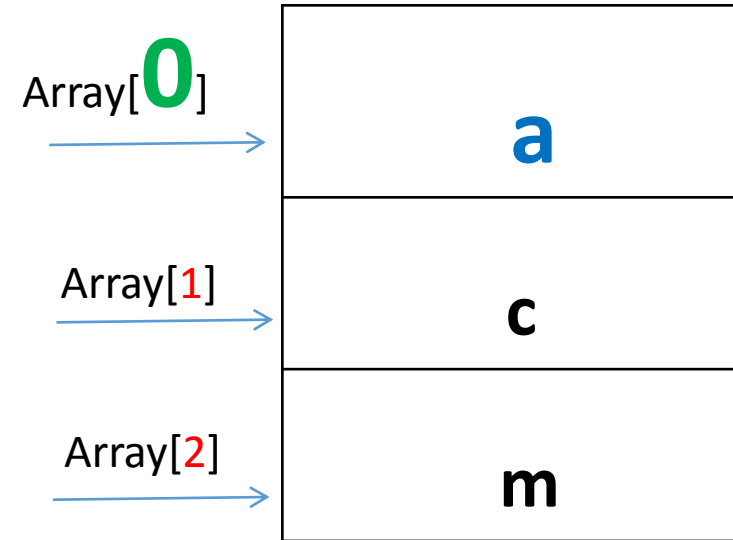
- index

- An **integer** starts from **zero**
- It is used to reach a specific value in the array.

# Array [ ]

- **Initializing array**

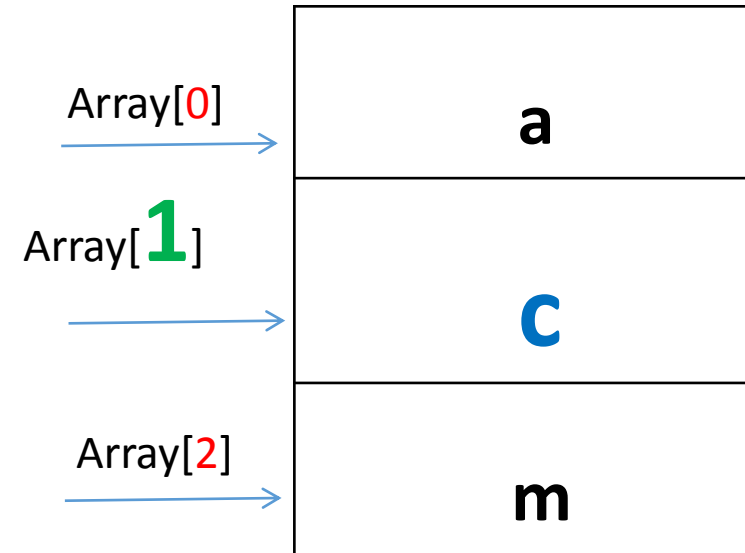
```
char Array [3] = { 'a' , 'c' , 'm' };
```



# Array [ ]

- Initializing array

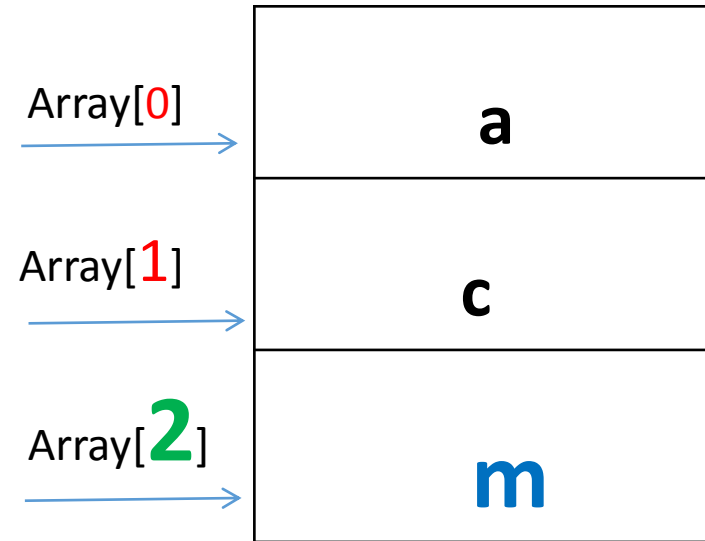
```
char Array [3] = { 'a' , 'c' , 'm' };
```



# Array [ ]

- **Initializing array**

```
char Array [3] = { 'a' , 'c' , 'm' };
```



Any question !!



# Array [ ]

- Common mistake

```
int Array [x] ;
```



Array [ ]





# Array [ ]

- Common mistake

```
int x ;
```

```
int Array [x] ;
```



Array [ ]



# Array [ ]

- Common mistake

```
int Array [100] ;
```



Array [ ]



# Array [ ]

- Common mistake

```
int main ()  
{  
  
int Array [4] = { 1 , 3 , 5 , 10 }  
  
cout << Array [4];  
  
return 0;  
}
```



Array [ ]



# Array [ ]

```
cin >> arr[0];  
cin >> arr[1];  
cin >> arr[2];  
cin >> arr[3];  
cin >> arr[4];  
cin >> arr[5];  
.....  
  
cout << arr[0];  
cout << arr[1];  
cout << arr[2];  
cout << arr[3];  
cout << arr[4];  
cout << arr[5];  
.....
```



# Array [ ]

## •Loops and Array

```
for (int i = 0; i < size; i++)  
{  
    cin >> arr[i];  
}
```

```
for (int i = 0; i < size; i++)  
{  
    cout << arr[i];  
}
```





Any question !!



# Array [ ]

- mirror

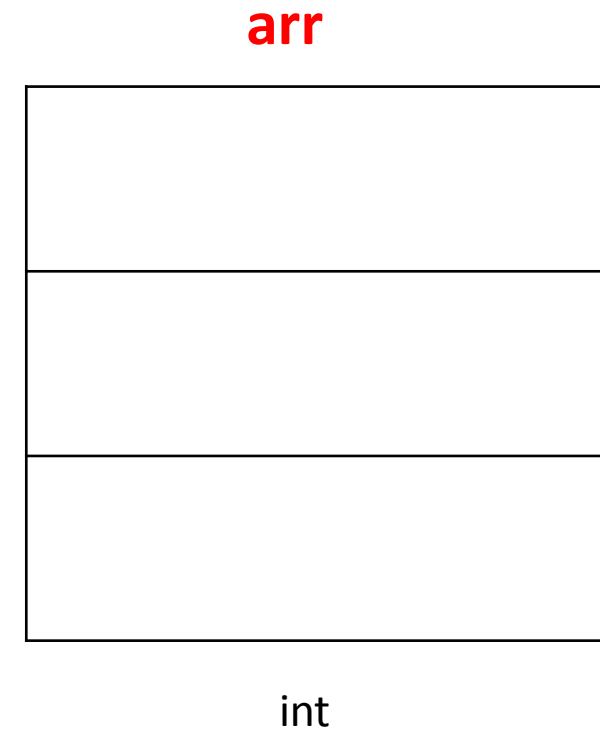
Now I need to get 3 Numbers from the user and  
print them out like mirror

# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```

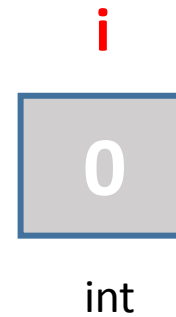


# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```

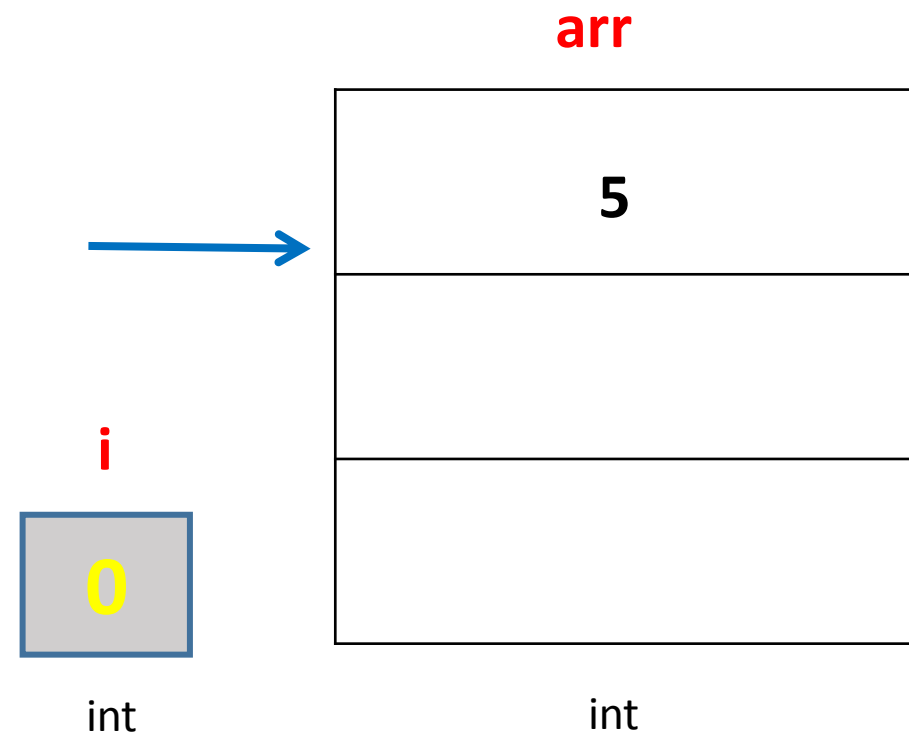


# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```

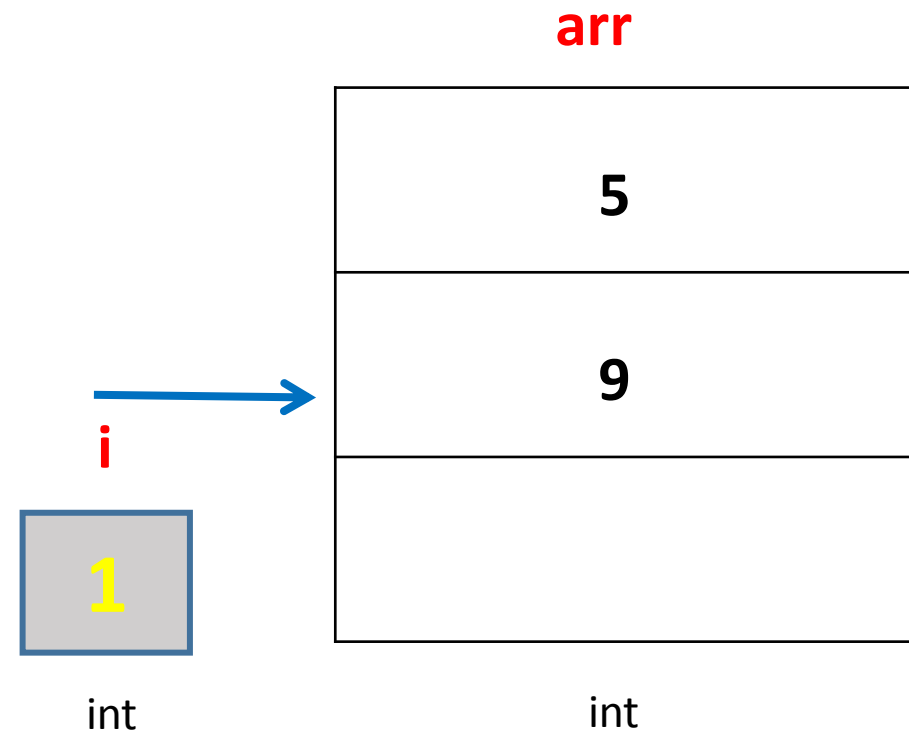


# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```

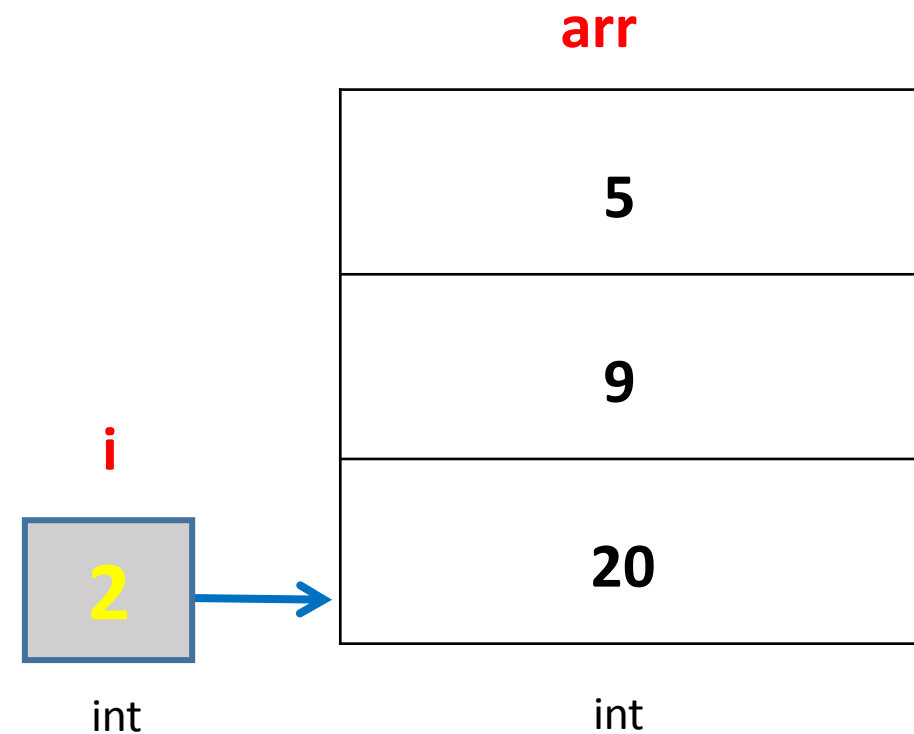


# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```



# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```

**arr**

5
9
20

int



**i**

2
---

int



# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```



**i**  
2  
int

**arr**

5
9
20

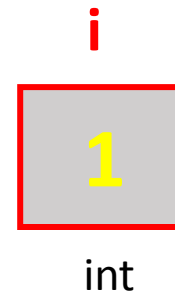
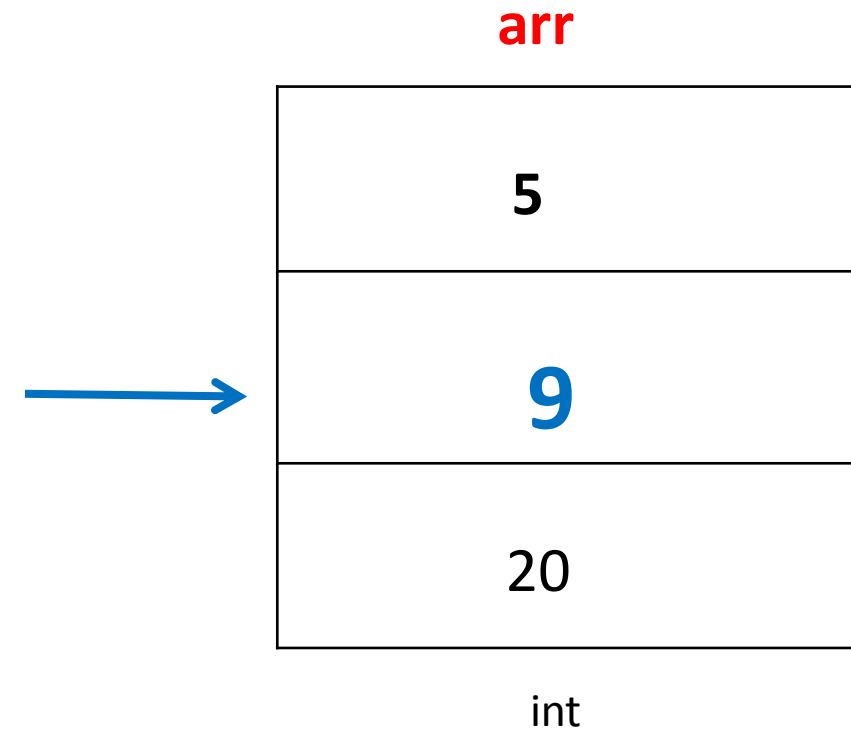
int

# Array [ ]

```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```

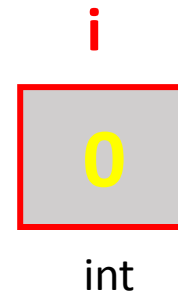
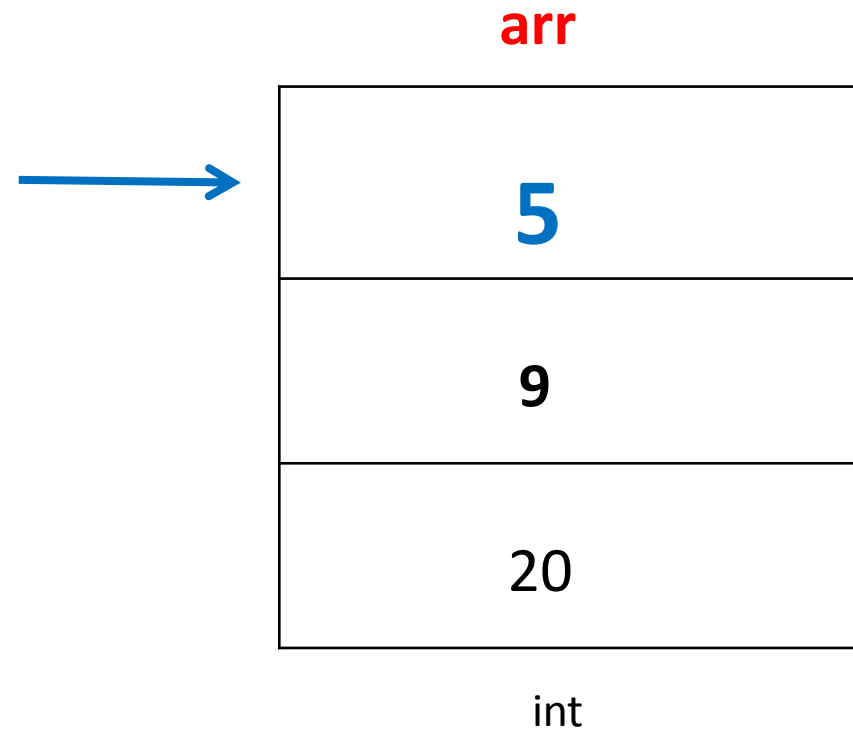


# Array [ ]

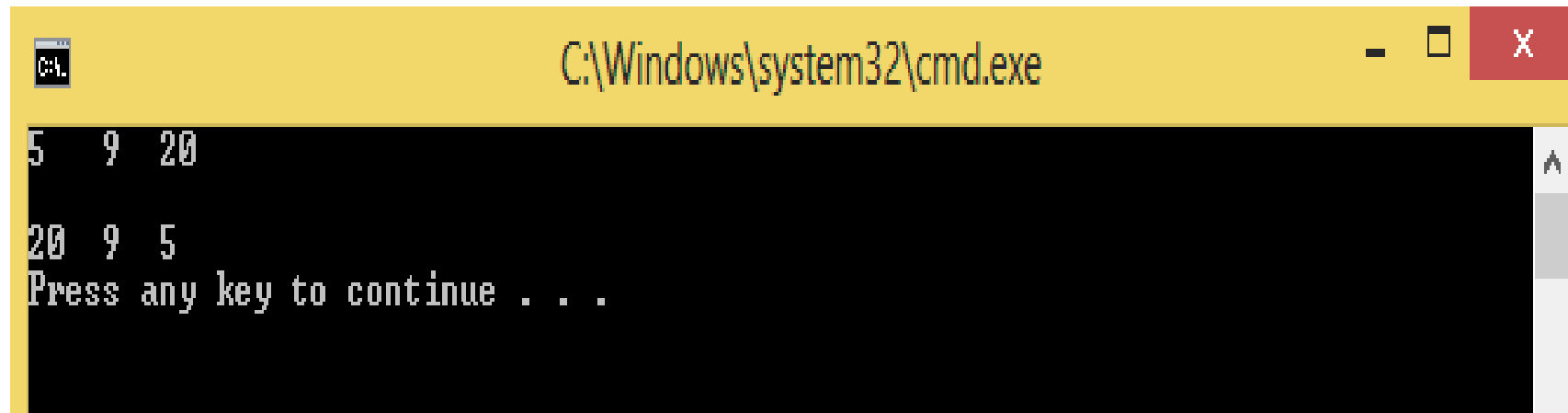
```
#include <iostream>
using namespace std;
int main ()
{
    int arr [3] ;

    for (int i=0 ; i<3 ;i++)
    {
        cin >> arr[i];
    }
    for (int i=2 ; i>=0; i-- )
    {
        cout << arr[i]<< endl;
    }

    return 0;
}
```



# Array [ ]



A screenshot of a Windows command prompt window with a yellow title bar. The title bar contains the text "C:\Windows\system32\cmd.exe" and standard window controls. The command prompt shows the following text:

```
5 9 20  
20 9 5  
Press any key to continue . . .
```

Any question !!



# Array [ ]

- **Linear search**

given a sequence of numbers and value  $k$  you have to determine if  $k$  exists in the sequence or not.

input :

input starts with  $t$  ( $t < 100$ ) denotes the number of test cases

each test case starts with a number  $n$  ( $n < 1000$ ) denotes the number of elements of the sequence followed by  $n$  elements then the number  $k$

output :

For each test case, output one line in the format "**Case  $x$** " where  $x$  is the test case number, followed by "YES" if  $k$  exists in the element or "NO" if not .

# Array [ ]

- Linear search

given a sequence of numbers and value  $k$  you have to determine if  $k$  exists in the sequence or not.

input :

input starts with  $t$  ( $t < 100$ ) denotes the number of test cases

each test case starts with a number  $n$  ( $n < 1000$ ) denotes the number of elements of the sequence followed by  $n$  elements then the number  $k$

output :

For each test case, output one line in the format "**Case  $x$** " where  $x$  is the test case number, followed by "YES" if  $k$  exists in the element or "NO" if not .

# Array [ ]

- Linear search

**Sample input :**

2

2

1      5

5

3

1      4      5

5

**Sample output :**

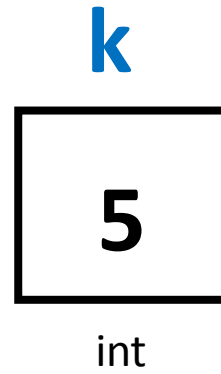
Case 1 :YES

Case 2 :NO



# Array [ ]

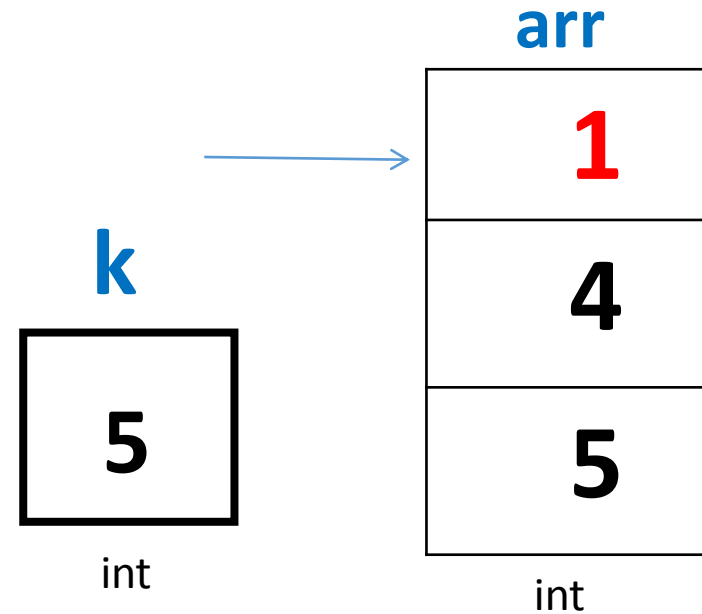
- Linear search



# Array [ ]

- Linear search

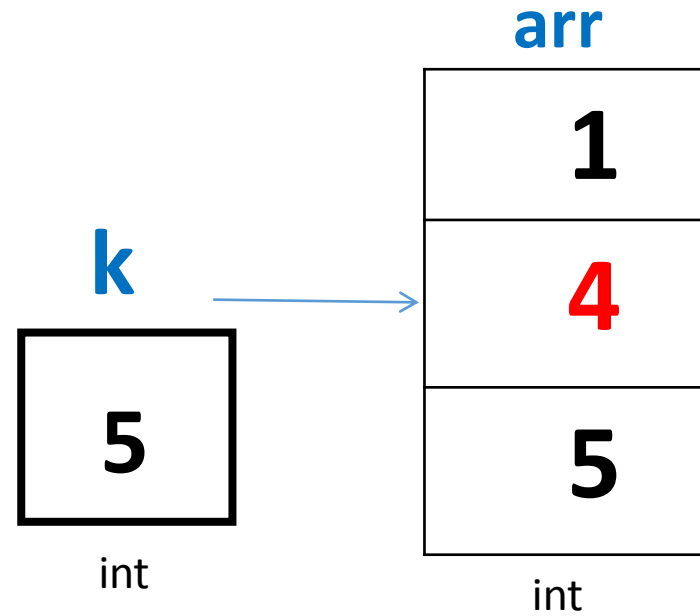
**false**



# Array [ ]

- Linear search

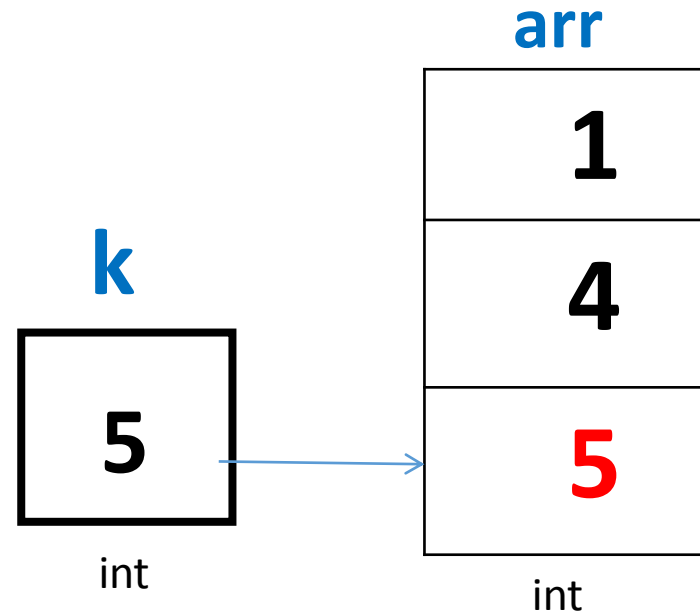
**false**



# Array [ ]

- Linear search

True



# Array [ ]

- Linear search

```
#include <iostream>

using namespace std;

int main()
{
    int testcases,n,k;
    int arr[1010];
    cin>>testcases;
    for(int i=1;i<=testcases;i++)
    {
        cin>>n;
        for(int j=0;j<n;j++)
        {
            cin>>arr[j];
        }
        cin>>k;
        bool found=false;
        for(int j=0;j<n;j++)
        {
            if(arr[j]==k)
            {
                found=true;
                break;
            }
        }
        cout<<"Case " <<i<<" ";
        if(found==true) cout<<"YES"<<endl;
        else cout<<"NO"<<endl;
    }
    return 0;
}
```

Any question !!





# It's Your Turn!

# Array [ ]

## •Summation search

given n numbers and a value k you have to determine if there exists two numbers such that their sum is equal to k or not

input :

the input contains several test cases

each test case start with n ( $n < 1000$ ) the number of elements

followed by n elements then the value k

input terminates with end of file .

output :

For each test case, output one line in the format "**Case x** " where x is the test case number, followed by "YES" if there exists two numbers their sum is equal to k or "NO" if not.





# Array [ ]

## •Summation search

given  $n$  numbers and a value  $k$  you have to determine if there exists two numbers such that their sum is equal to  $k$  or not

input :

the input contains several test cases

each test case start with  $n$  ( $n < 1000$ ) the number of elements followed by  $n$  elements then the value  $k$

input terminates with end of file.

output :

For each test case, output one line in the format "**Case x**" where  $x$  is the test case number, followed by "YES" if there exists two numbers their sum is equal to  $k$  or "NO" if not.

# Array [ ]

## •Summation search

### Sample input :

4

1 4 3 5

9

3

1 5 6

4

### Sample output :

Case 1 :YES

Case 2 :NO

# Array [ ]

- Summation search

```
int main()
{
    int n,k,tc=1;
    int arr[1010];
    while(cin>>n)
    {
        for(int i=0;i<n;i++)
        {
            cin>>arr[i];
        }
        cin>>k;
        bool found=0; // 0 means false
        for(int i=0;i<n;i++)
        {
            for(int j=i+1;j<n;j++)
            {
                if(arr[i]+arr[j]==k)
                {
                    found=1; // 1 means true
                    break;
                }
            }
            if(found==1) break;
        }

        cout<<"Case "<<tc<<" ";
        if(found==1) cout<<"YES"<<endl;
        else cout<<"NO"<<endl;

        tc++;
    }
    return 0;
}
```

Any question !!



# Thank You

