

variable:

-kono khabar jodi kono koutate rakha hoi thle sei khabar hcce data r j koutar moddhe rakha hoi sei kouta ta holo variable

Insertion operator - << (cout<<)

tag dia suru howa kichu ke pre pocessore director bole . (#include<iostream>
meaning: C++ pre processor to include this iostream file

header file -> pre-return code .Basic input output function.

(iostream, stdio.h, math.h,)

using namespace st; -> to use std(standard) form of input/output(1/0)

ASCII Value :

A = 65 , B = 66 Y = 89 , Z = 90
a = 97 , b = 98 y = 121 , z = 122
0 = 48 , 1 = 49 8 = 56 , 9 = 57

int:

simple number

float:

(.)doshomik type number...maybe decimal number bole.like real number(3.14)

char:

#, @, aA egula use er jonne

Comment:

1.// -single line comment. (Example: This is a single line comment.)

2./* _____ */ -multi line comment. (Example: This is not a single line comment.
This is a multi line comment.*/)

Sentence Break:

printf("_____/n"); or scanf("_____/n");

sum rule:

jdi emon lekha hoi j -

```
int sum=a+b
printf(" The sum is: %d", sum);    tahole int main(){ er por j int likhe variable lekha hoi
sekhane sum lekha lagbe na.
```

```
int sum=a+b
printf(" The sum is : sum");    eta poriborte emon lekha jai printf("The sum is %d",a+b); eikhetreo int
main(){ er por int likhe sum lekha lagbe na...
```

MOD:

-mod er mansomuho shudu int e kaj kore ,real number e kaj kore na.....

Force Type:

- Example:

```
int a= (int) 1.9999999;
printf("%d",a);
```

Priority:

!, *, /, %, +-, <> <=, == !=, &&, ||, =

Hierarchy of operators :

BODMAS -> Brackets, Order, Division, Multiplition, Addition, Substitution ...

In real life math BODMAS works -> B/O/D,M/A,S

In programming life (C++) BODMAS works -> left to right but BODMAS rule...

Convert Priority:

- 1.int operation int= int
- 2.int operation float= float
- 3.float operation float= float

Associativity rule:

- calculate (*,/,%,+,-,=) left to right
- Example : $4*3/6*2$
- Answer :4

True-False:

- 1.True : 1(2,3,4,5,-1,-2,-3,-4)
- 2.False : 0

Assignment Operators:

1. =
2. +=
3. *=
4. /=
5. %=

Example:

$a=a+b$ / $a+=b$

Modulus Operator :

- * $a \% b = a$ [$a < b$]
- * $a \% a = 0$
- * $a \% (-b) = a \% b$
- * $(-a) \% b = -[a \% b]$

Type casting :

Example:

```
int x;  
cin >> x;  
float y = (float) x;  
cout << y;
```

Ternary Operator:

Law:

```
condition ? True[printf("___"):False("___");
```

Switch rule:

```
switch (int/char next word) {  
    case 1/a : printf("____");  
    break;  
    case 2/b : printf("____");  
    break;  
    case 3/c : printf("____");  
    break;  
    default: printf("____");  
}
```

Note:jodi break use na kortam thle sob gulai print hoto tobe j num input kora hoto sei num soho porer num gula print hoto,kintu ager num gula print hoto na.....

Nested if else:

```
if(__condition){  
    printf("____\n");  
    if(__condition__){  
        printf("____\n");  
    } else {  
        printf("____\n");  
    }  
} else {  
    printf("____\n");  
}
```

===

```
*i++ : post increment  
*++i : pre increment  
*i-- : post increment  
*--i : pre increment
```

Introduction of post increment: Current value of i is preserved temporarily i will get incremented before next statement is executed.

Example:

```
int a = 10;
int b;

b = a++ ;
cout<<a" "<<b<<endl;
```

Output: a=11, b=10

Introduction of pre increment : Increment i instantly

Example:

```
int a = 10;
int b;

b = a++ ;
cout<<a" "<<b<<endl;
```

Output: a=11, b=10

```
###
int i=1;
while(i<=5) {
    printf("%d SHRE \n",i);
    i++;
}
```

as same as

```
int i=1;
while(i<=5) {
    printf("SHRE \n");
    i++;
}
```

Loop:

1. for loop-

```
for( increment ; condition ;updation ) {
    printf("%d \n",i);
}
```

Example: for(int i=1 ; i<=5 ; i++) {

printf("%d \n",i); [tbe eikhane int i=1 e ciale i=1 lekha jabe tobe seikhetre for er uporer

line e int i; bole dite hobe...]

2. while loop-

```

int increment;
while( condition ){
    printf("%d \n",i);
    updation;
}

```

Example: int i=1;

```

while(i<=5) {
    printf("%d \n",i);
    i++;
}

```

3. do while loop-

```

int increment;
do{
    printf("%d \n", i);
    updation;
} while ( condiotion );

```

Example: int i=1;

```

do {
    printf("%d \n", i);
    i++
} while (i<=5);

```

Break Loop Rule:

```

- for(int increment ; condition ; updation) {
    if (i==n){
        break;
    }
    printf("%d \n",i);
}

```

Example: for(int i=1 ; i<=5 ; i++) {

```

    if (i==3) {
        break;
    }
    printf("%d \n", i)
}

```

Array Structure:

```

int/float/cha ____varieable____[] = {value,value,____};

```

- int main function e seser print e ja thakbe tai argument[eta nijer mote]
- function e bracket e ja thakbe seta hcece parameter

calculate the sum of 2 number(Q)

Initialization of Arrays:

data_type array_name[how_much_array];
int marks[3] = {10,20,30} or int marks[] = {10,20,30} ----direct value initialization er jnne,kheyal rakhte
hobe equal er pore jeno second bracket use kora hoi

2D Array/Matrix Array:

initialization-

```
int A[row][colom] ={  
    {a[0][0] a[0][1]}  
    {a[1][0] a[1][1]}  
    {a[2][0] a[2][1]}  
}
```

or

```
A[0][0]=5;  
A[0][1]=6;  
A[1][0]=21;  
A[1][1]=12;
```

Error and solve:

-uninitialized =&

Array is a Pointer:

```
int*ptr= & arr[0]; or int*ptr =arr;
```

String with null character:

```
char name[] = ('S','H','O','V','O','N','\0'); or char name[] = "SHOVON";
```

eikhane c-compiler automatic \0 input kore dibe..

String format specifier:

```
char name[]=SHRE;  
printf("%s",name);
```

and

```
char name[50];  
scanf("%s",name);
```

- %s input dile automatic \0 [null character] active kore dei.....

String function:

1. gets(str) - input a string [even multiword] [Dangerous % outdated]
2. puts(str) - output a string
3. fgets(str) - stops when n-1 chars input or new line is entered

Structure:

```
char str[100];  
fgets(str,n,file); [ or gets(str),,but eta Dangerous and outdated]  
puts(str);
```

eikhane puts(str) dile aladavabe \n dite hoina.automatic new line hoye jai....
file e name dibo stdin

```
char str[] = char * str {is equal to}
```

condition:

```
# char str[]-eta change kora jaina {cannot be reinitialized}
```

```
#char *str- eta change kora jai {can be reinitialized}
```

Example:

```
char *canChange = " C programming";
puts(canChange);
canChange = "Progrmming";
puts(canChange);
```

```
char cannotChange[] = "C Programming";
puts(cannotChange);
cannotChange = "Programming"; [Expression must be a modifiable lvalue
```

<string.h>---

Example:

```
#include<stdio.h>
#include<string.h>
```

```
int main()
{
    char name[]="SHRE";
    int length = strlen(name);
    printf("Length is :%d",length);
    return 0;
}
```

eikhatre strlen shudu character count korbe but \0[null character] count korbe na....amar mote, strlen k input korar jnne int te convert kore nite hoi... sejonno direct strlen(____) k print na kore onno ekta int(____) e convert kore nite hobe....amar mote,ei jonno alada vabe loop/function creat korte hoi na...

strcpy structure:

```
strcpy(newstr,oldstr);
```

Example:

```
#include<stdio.h>
#include<string.h>
int main()
{
    char oldstr[] = "FF";
    char newstr[] = "COC";
    strcpy (newstr,oldstr);
    puts(newstr);
}
```

strcat structure: [cat=concatenates/concating]

```
strcat(firststr,secondstr);
```

Example:

```
#include<stdio.h>
```

```
#include<string.h>
```

```
int main()
```

```
{
```

```
    char first[100] = "Clash of ";
```

```
    char second[] = "Clans";
```

```
    strcat(first,second);
```

```
    puts(first);
```

```
}
```

eikhane first[] e 100 dewa hoise karon first store e first char soho second char er output ese store hobe...jodi first e store space na dewa hoi thle first er j koita char lekha hobe sei koita char er space nibe and output e oi koita char thakbe bole second char first char e ese store korte parbe na sejonno beshi kore storage no{100} dia dite hobe....tobe second char e store num dite hobe na karon second char to first char e ese store hobe....

strcmp structure:

```
strcmp(firstcmp,secondcmp);
```

-compare 2 strings & returns a value....

The value are:

1. 0 - string equal
2. Positive - [First > Second] {scan with ASCII number}
3. Negative - [First < Second] {scan with ASCII number}

Example:

```
#include<stdio.h>
```

```
#include<string.h>
```

```
int main()
```

```
{
```

```
    char firststr[] = "Apple";
```

```
    char secondstr[] = "Banana";
```

```
    printf("%d",strcmp(firststr,secondstr));
```

```
}
```

tobe eikhane ager string er moton value print hobe na...printf er sahajje value print korte hobe seta khewal rakhte hobe....

ASCII number compare er khetre first e fast word math check korbe jodi same hoi thle next word check

korbe...jar ASCII number beshi se win(bujhar jnne)...

Structure:

Example:

```
#include<stdio.h>
#include<string.h>

// user defined
struct student{
    int roll;
    float cgpa;
    char name[100];
};

int main()
{
    struct student s1;
    s1.roll=365;    {s1.roll-eikhane . holo dot operator]
    s1.cgpa=4.00;
    //s1.name= "SHOVON"    [jehetu struct user define e char namep[100]; use kora hoyese sejonno eta
next e change kora jai na{ager theory.....sejonno strcpy dia copy kore bosano hoise.....
    strcpy(s1.name, "SHOVON");

    printf("Student Name =%s\n",s1.name);
    printf("Student roll =%d\n",s1.roll);
    printf("Student cgpa =%f\n",s1.cgpa);

    return 0;
}
```

Array of Struckture:

Example:

```
#include<stdio.h>
#include<string.h>

// user defined
struct student{
    int roll;
    float cgpa;
    char name[100];
```

```
};

int main()
{
    struct student CSE[100];
    CSE[0].roll = 365;
    CSE[0].cgpa = 4.00;
    strcpy(CSE[0].name,"SHOVON");

    printf("Name = %s\n",CSE[0].name);
    printf("Roll = %d\n",CSE[0].roll);
    printf("cgpa = %f\n",CSE[0].cgpa);

    return 0;
}
```

Inalization of Structure:

```
struct student{
    int roll;
    float cgpa;
    char name[100];
};

int main()
{
    struct student s1 = {roll,cgpa,name};
    _____ ---
    return 0;
}
```

Example:

```
#include<stdio.h>
#include<string.h>

// user defined
struct student{
    int roll;
    float cgpa;
    char name[100];
};

int main()
{
    struct student s1= {365,4.00,"SHOVON"};

    printf("Name = %s\n",s1.name);
    printf("Roll = %d\n",s1.roll);
    printf("cgpa = %f\n",s1.cgpa);

    return 0;
}
```

Pointer of Structure:

Example:

```
#include<stdio.h>
#include<string.h>

// user defined
struct student{
    int roll;
    float cgpa;
    char name[100];
};

int main()
{
    struct student s1= {365,4.00,"SHOVON"};

    struct student *ptr = &s1;
    printf("Student roll=%d\n",(*ptr).roll);
    printf("Student all=%d\n",(*ptr)); {all print er jonne}

return 0;
}
```

Exchange of pointer in structure:

(*ptr).code = ptr->code

Typedef Keyword:

-used to creat alias for data types....

Structure:

```
typedef struct ComputerScienceandEndineering{
    char name[100];
    int roll;
    float cgpa;
```

```
} cse ;
```

Example:

```
#include<stdio.h>
#include<string.h>

typedef struct ComputerScienceandEndineering{
    char name[100];
    int roll;
    float cgpa;
} cse ;

int main()
{
    cse s1;
    s1.roll= 365;
    s1.cgpa = 4.00;
    strcpy(s1.name, "SHOVON");

    printf("Roll is =%d\n",s1.roll);
    printf("cgpa is =%f\n",s1.cgpa);
    printf("Name is =%s\n",s1.name);
    return 0;
}
```

Volatile memory: RAM

Non-Volatile memory: Hard-Disk

File:

Opening a file-

```
FILE *ptr;
ptr= fopen("filename" , "mode");
```

Closing a file-

```
fclose(fp);
```


Opening File mode:

```
# "r" - open to read          {jodi file creat na hoi seikhetre null ese jabe}  
# "rb" - open to read in binary {jodi file creat na hoi seikhetre null ese jabe}  
# "w" - open to write          {jodi file creat na hoi seikhetre nije file creat hoye jabe,eikhetre old data  
delete hoye jabe and new data input hoye jabe}  
# "wb" - open to write in binary {jodi file creat na hoi seikhetre nije file creat hoye jabe,eikhetre old data  
delete hoye jabe and new data input hoye jabe}  
# "a" - open to append          {old data save rekhe old data er sathe new data input er jonne}
```

Reading from a file:

structure:

```
char ch;  
fscanf(fptr,"%c",&ch);  
printf("Character = %c",ch);
```

Write a file:

structure:

```
char ch = 'S';    {na likhleole cole}  
fprintf(fptr,"%c",ch);
```

Read & Write a char: {only for character}

```
# fgetc(fptr)  
# fputc('A',fptr)
```

EOF-End of File

Function for Dynamic Memory Allocation(DMA):

-takes number of bytes to be allocated & returns a pointer of type void...

-Header file: #include<stdlib.h>

malloc() - memory allocation

Structure:

```
ptr = (int*)malloc(5*sizeof(int));
```

Example:

```
#include<stdio.h>
#include<stdlib.h>

int main()
{
    int*ptr;
    ptr = (int *) malloc (5 * sizeof(int));

    ptr[0] = 1;
    ptr[1] = 2;
    ptr[2] = 3;
    ptr[3] = 4;
    ptr[4] = 5;

    for(int i=0;i<5;i++){
        printf("%d\n",ptr[i]);
    }
    return 0;
}
```

calloc() - continuous allocation

initializes with 0 {input korar sathe sathe array er vitore 0(null number) creat hoye jai...sejonno null programe creat korar jonne calloc() use korte hobe....}

structure:

```
ptr = (float *) calloc (5 , sizeof(float));
```

Example:

```
#include<stdio.h>
#include<stdlib.h>

int main()
{
    float*ptr;
    ptr = (float *) calloc (5 , sizeof(float));
    for(int i=0;i<5;i++){
        printf("%f\n",ptr[i]);
    }
    return 0;
}
```

free() -

we use it to free memory that is allocated using malloc & calloc

Structure:

```
free(ptr)
```

Example:

```
#include<stdio.h>
#include<stdlib.h>
```

```

int main()
{
    int*ptr;
    int n;
    printf("Enter n:");
    scanf("%d",&n);

    ptr = (int *) calloc (n , sizeof(int));
    for(int i=0;i<n;i++){
        printf("%d\n",ptr[i]);
    }
    free(ptr);
    ptr = (int *) calloc (2 , sizeof(int));
    for(int i=0;i<2;i++){
        printf("%d\n",ptr[i]);
    }
    return 0;
}

```

realloc() - re-allocation

reallocate (increase or decrease) memory using the same pointer & size....

Structure:

```
ptr = realloc (ptr,newsiz);
```

Example:

```
#include<stdio.h>
```

```
#include<stdlib.h>
```

```

int main()
{
    int*ptr;
    ptr = (int *) calloc (5 , sizeof(int));
    printf("Enter numbers(5) :");
    for (int i=0;i<5;i++){
        scanf("%d\n",&ptr[i]);
    }
    ptr = realloc(ptr,8);
    printf("Enter numbers(8) :");
    for (int i=0;i<8;i++){
        scanf("%d\n",&ptr[i]);
    }
    //print
    for(int i=0;i<8;i++){
        printf("Number %d is %d\n",i,ptr[i]);
    }
    return 0;
}

```

= -> ei operator er somoi operation hoi right to left.....

Composite number and prime number :

- * n has no extra factor only 1 & n -> prime
- * n ha extra factor except 1 & n -> composite

Break -> Condition true howar sathe sathe sekhane loop end hoye jabe jodio loop er aro condition thake.....

Continue -> If you want to skip a particular round...

while loop convert to for loop:

while loop ->

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

```
int main()
{
    int i=1;
    while(i<=n){
        cout<<i<<" ";
        i++;
    }

    return 0;
}
```

convert to while loop ->

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

```
int main()
{
    int i=1;
    for(;i<=n;){
        cout<<i<<" ";
        i++;
    }

    return 0;
}
```

note:

while loop er somoi while(i<=n) lekha lagbe kintu for loop er belai for(;i<=n;) lekha lagbe....for loop er parenthesis er er open e statement terminator dite hobe and close er somoi terminator dite hobe.....

Function:
Structure:

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

```
void basic(){
    cout<<"SHRE";
    return;
```

```
int main()
{
    basic();
    return 0;
}
```

*** Function call er khetre main function er upor call kora function thakte hbe. abar oi call kora funtion jdi onno kono function ke call kore sekhetreo oi onno funtion ti prothom call kora function er upore thaka lagbe . nahole Error type emon kichu show korbe.

Ei problem solve korte chaile j funtion ti call funtion er nice ba onno kothau rakhte caibo sei function tir return type + function name(); eivabe call kora funciton er vitore rekhe tar por oi function k call korte hbe...

tarpur onno function er khetreo same kaj kora lagbe...

Example:

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

```
void CR(){
    cout<<"Clash Royal\n";
    void BB();
    BB();
    return;
}
```

```

int main(){
    void COC();
    COC();
    return 0;
}

void COC(){
    cout<<"Clash of Clans\n";
    void CR();
    CR();
    return;
}

void FF(){
    cout<<"Free Fire \n";
    return;
}

void BB(){
    cout<<"Boom Beach\n";
    FF();
    return;
}

```

Scope of variable :

Example:

```

int main()
{
    int a;
    cin>>a;

    for(int i=1; i<=a; i++){
    }
}

```

eikhane for loop er vitore int i initialize kora hoise eta shudu matro for loop er vitore access kora jabe...main function er vitore access hobe na...
bola jai int i jeta initialize kora hoise setar power only for loop er vitore...

Formal parameters & Actual parameters :

Example:

```
int main()
{
    int a=2;
    int b=4;

    swap(a,b);

}
```

* Formal parameters : a & b

* Actual parameters : 2 & 4 [a,b er value hce Actual parameters]

Pointer :

Example in C :

```
int main()
{
    int a=5;
    printf("%p",&a);
    return 0;
}
```

eikhane data format hisabe %p use hobe....pointer e data format hisabe %p use hoi address janar jnne....tbe jdi data format hisabe %p er bodole %d use kora hoi seikhetre kichu num show korbe tobe seta address na..address tokhoni kichu num hisabe dekhabe jekhona data format hisabe %p use kora hobe...

Again,

```
int main()
{
    int a=5;
    int* x= &a;
    printf("%p",x);
    printf("%p",&x);
    printf("%p",*x);
    return 0;
}
```

Eikhane,

printf("%p",x); -> address of a.
printf("%p",&x); -> address of x.
printf("%p",*x); -> x er vitore j address ta ase tar valur mane a er value.

Again2,

```
int main()
{
    int a=5;
    int* x= &a;
    *x=7;
    printf("%p",a);
    return 0;
}
```

Output : 7....

karon , eikhane pointer onno kono address er (a no int) value k change kortece...See Again1 more more to gain knowledge...

Pass by reference or basic ++ or jnne College Wallah C++ video or 7:49 hour theke watch....

Arrays :

Accessing Elements of arrays using square brackets...

Example: `arr[2];`

- * Array -> List
- * 2D Array -> Table
- * 2D Array -> Array of Array

Strings:

* Initialization and Declaration jdi eksathe kora hoi thle jkno sentence print kora jai orthat word er por gap(space) thakleo se gap(space) print hobe sathe gap (space) er porer word o print hobe...
Tobe jodi prothome Declaration kore pore initialization kora hoi thle prothome word ta print hobe but space+porer kono word print hobe na...
ei problem solve er jnne syntax hcece :
 getline(cin,s)

Example: int main()
{
 string s;
 getline(cin,s);
 cout<<s;

```
    return 0;  
}
```

Build in function:

push_back -> only 1 ta character add kora possible, not multiple character.

Formula : s.push_back('a');

pop_back -> word er last theke only 1 ta character delete howe jabe...

Formula : s.pop_back();

append -> etar dara full word with space with multiple word add kora i...tbe single quotation hbe na...Double quotation use korte hbe...

Formula : s.append("Clans");

clear -> full string empty kore dei...

+(plus) operator -> kono kichu samne ba pisone add korar jonno with space..

Formula : s = s + " Clash";

or,

s= "Free " s;

to_string -> int k strings e convert kore...tokhon oi strings er sathe jekono kichu add kora,reverse kora sob kora jai...

stoi -> strings convert to int...orthat string k int e convert korar jnne..

Formula : int x = stoi(s);