

Write a program to show the toll charges to be levied on a vehicle that crosses a toll booth, based on in which state it is registered. Get input from the user about the state in which the vehicle is registered.

State Code Karnataka K Kerala R Tamil Nadu T Andhra Pradesh A Telangana G

Toll Charges Rs. 200 Rs. 250 Rs. 300 Rs. 240 Rs. 220

```
#include <stdio.h>
int main() {
    char code;
    scanf("%c",&code);
    switch(code){
        case 'K':
            printf("Total Charges: 200");
            break;
        case 'R':
            printf("Total Charges: 250");
            break;
        case 'T':
            printf("Total Charges: 300");
            break;
        case 'A':
            printf("Total Charges: 240");
            break;
        case 'G':
            printf("Total Charges: 220");
            break;
    }
    return 0; }
```

Operators in C

An operator is used to perform either a mathematical operation or a logical operation between two values (constants or variables).

Types:

- ➤ Arithmetic operators
- ➤ Increment or decrement operators
- ➤ Relational operators
- ➤ Logical operators
- ➤ Bitwise operators
- ➤ Assignment operators
- ➤ Conditional operators
- ➤ Special operators

Arithmetic operators

14

Operator Program

Output

+ Addition	<pre>#include<stdio.h> void main() { int a=30,b=40, c; c=a+b; printf("%d", c); }</pre>	
- Subtraction	<pre>#include<stdio.h> void main() { int a=30,b=40, c; c=a-b; printf("%d", c); }</pre>	
* Multiplication	<pre>#include<stdio.h> void main() { int a=30,b=40, c; c=a*b; printf("%d", c); }</pre>	
/ Division	<pre>#include<stdio.h> void main() { int a=30,b=40, c; c=a/b; printf("%d", c); }</pre>	
% Modulus	<pre>#include<stdio.h> void main() { int a=30,b=40, c; c=a%b; printf("%d", c); }</pre>	

Increment or Decrement operator

Operator Program Output

++ Pre-Increment	#include<stdio.h> void main() { int a=30,b; b=++a; printf("%d", b); }	
++ Post-increment	#include<stdio.h> void main() { int a=30,b; b=a++; printf("%d", b); }	
-- Pre-decrement	#include<stdio.h> void main() { int a=30,b; b=--a; printf("%d", b); }	
-- Post-decrement	#include<stdio.h> void main() { int a=30,b; b=a--; printf("%d", b); }	

What is the output for the given program?

```
#include<stdio.h> void main()
{
int a, b, c; a=20;
b=30; c=a+b; a=c-b; b=a*c; c=b/a; a=++c; b=a++;
c=--b;
a=c--; printf("%d",a);
}
```

Output: _____

Relational operators

Operator Program

Output

	<pre>#include<stdio.h> void main() { int x=40,y=60; if(x==y) printf("True"); else printf("False"); }</pre>	
!= Not Equal	<pre>#include<stdio.h> void main() { int x=40,y=60; if(x!=y) printf("True"); else printf("False"); }</pre>	
> Greater than	<pre>#include<stdio.h> void main() { int x=40,y=60; if(x>y) printf("True"); else printf("False"); }</pre>	
< Less than	<pre>#include<stdio.h> void main() { int x=40,y=60; if(x<y) printf("True"); else printf("False"); }</pre>	
>= Greater than or equal to	<pre>#include<stdio.h> void main() { int x=40,y=60; if(x>=y) printf("True"); else printf("False"); }</pre>	

<= Less than or equal to	#include<stdio.h> void main() { int x=40,y=60; if(x<=y) printf("True"); else printf("False"); }	
---------------------------------------	--	--

Logical operators

17

Operator Program

Output

	#include<stdio.h> void main() { int x=1,y=2; if(x&& y) printf("Both are non zero numbers"); else printf("At least one of the numbers is 0"); }	
&& Logical AND	#include<stdio.h> void main() { int x=1,y=0; if(x&& y) printf("Both are non zero numbers"); else printf("At least one of the numbers is 0"); }	

	<pre>#include<stdio.h> void main() { int x=1,y=0,z=5; if(x&& y&&5) printf("Both are non zero numbers"); else printf("At least one of the numbers is 0"); }</pre>	
 Logical OR	<pre>#include<stdio.h> void main() { int x=1,y=0; if(x y) printf("There is at least one non-zero number"); else printf("All are 0s"); }</pre>	
	<pre>#include<stdio.h> void main() { int x=0,y=0; if(x y) printf("There is at least one non-zero number"); else printf("All are 0s"); }</pre>	
	<pre>#include<stdio.h> void main() { int x=0,y=0,z=9; if(x y z) printf("There is at least one non-zero number"); else printf("All are 0s"); }</pre>	

Operator Program Output

! Logic Reversal	<pre>#include<stdio.h> void main() { int a=5, b=6; if(!(a&&b)) printf("1"); else printf("0"); }</pre>	
	<pre>#include<stdio.h> void main() { int a=5, b=0; if(!(a&&b)) printf("1"); else printf("0"); }</pre>	
	<pre>#include<stdio.h> void main() { int a=5, b=0,c=8; if(!(a&&b&&c)) printf("1"); else printf("0"); }</pre>	
	<pre>#include<stdio.h> void main() { int a=5, b=0; if(!(allb)) printf("1"); else printf("0"); }</pre>	
	<pre>#include<stdio.h> void main() { int a=0, b=0; if(!(allb)) printf("1"); else printf("0"); }</pre>	
	<pre>#include<stdio.h> void main() { int a=0, b=0, c=1; if(!(allblc)) printf("1"); else printf("0"); }</pre>	

Operations are carried out bit-by-bit. Operator Program
Binary Representation Output

Bitwise operators

& Binary AND	<pre>#include<stdio.h> void main() { int a=14, b=10, c; c=a&b; printf("%d",c); }</pre>	00001110 = 14 00001010 = 10	
Binary OR	<pre>#include<stdio.h> void main() { int a=12, b=11, c; c=a b; printf("%d",c); }</pre>	00001100 = 12 00001011 = 11	
^ Binary XOR	<pre>#include<stdio.h> void main() { int a=4, b=6, c; c=a^b; printf("%d",c); }</pre>	00000100 = 4 00000110 = 6	
~ Binary 1's complement	<pre>#include<stdio.h> void main() { int a=7,b; b=~a; printf("%d",b); }</pre>	00000111 = 7	
<< Binary Left Shift	<pre>#include<stdio.h> void main() { int a=3,b; b=a<<1; printf("%d",b); }</pre>	00000011 = 3	
>> Binary Right Shift	<pre>#include<stdio.h> void main() { int a=15,b; b=a>>2; printf("%d",b); }</pre>	00001111 = 15	

Assignment operators

20

Operator Program

Output

	<pre>#include <stdio.h> void main() { int a=4,b=5,c; c=a+b; printf("%d",c); }</pre>	
+= Add and assign	<pre>#include <stdio.h> void main() { int a=4,b=8; b+=a; printf("%d",b); }</pre>	
-= Subtract and assign	<pre>#include <stdio.h> void main() { int a=4,b=8; b-=a; printf("%d",b); }</pre>	
= Multiply and assign	<pre>#include <stdio.h> void main() { int a=4,b=8; b=a; printf("%d",b); }</pre>	

<div>/=</div> <div>Divide and assign</div>	<pre>#include <stdio.h> void main() { int a=4,b=8; b/=a; printf("%d",b); }</pre>	
<div>%=</div> <div>Modulus and Assign</div>	<pre>#include <stdio.h> void main() { int a=4,b=11; b%=a; printf("%d",b); }</pre>	

Operator Program

Binary Representation Output

<div><<=</div> <div>Left Shift and assign</div>	<pre>#include <stdio.h> void main() { int a=8; a<<=3; printf("%d",a); }</pre>	00001000 = 8	
---	---	--------------	--

<div>>>=</div> <div>Right Shift and assign</div>	<pre>#include <stdio.h> void main() { int a=8; a>>=3; printf("%d",a); }</pre>	00001000 = 8	
<div>&=</div> <div>Bitwise AND and assign</div>	<pre>#include <stdio.h> void main() { int a=5,b=6; b&=a; printf("%d",b); }</pre>	00000101 = 5 00000110 = 6	

= Bitwise OR and assign	<pre>#include <stdio.h> void main() { int a=5,b=6; bl=a; printf("%d",b); }</pre>	00000101 = 5 00000110 = 6	
^= Bitwise Exclusive OR and assign	<pre>#include <stdio.h> void main() { int a=9,b=11; b^=a; printf("%d",b); }</pre>	00001001 = 9 00001011 = 11	

Conditional Operators

Operator Program

Output

	<pre>#include <stdio.h> void main() { int a=9,b=10; a>b?printf("%d",a-b):printf("%d",b-a); }</pre>	
?: Conditional operation	<pre>#include <stdio.h> void main() { int a=15,b=10; a>b?printf("%d",a-b):printf("%d",b-a); }</pre>	
	<pre>#include <stdio.h> void main() { int a=15,b=10; a>b?(a+=b):(b*=a); printf("a = %d\n",a); printf("b = %d",b); }</pre>	

Operator Program

Special operators

22

sizeof() Size of a variable	<pre>#include <stdio.h> void main() { int a; char b; float c; printf("Size of an int is %lubytes.\n",sizeof(a)); printf("Size of a char is %lubyte.\n",sizeof(b)); printf("Size of a float is %lubytes.\n",sizeof(c)); }</pre> <p>Output: Size of an int is 4 bytes. Size of a char is 1 byte. Size of a float is 4 bytes.</p>
--------------------------------	--

<pre>#include<stdio.h> int main() { int a; 10=a; printf("%d",a); }</pre>		<pre>#include<stdio.h> int main() { int a; a=10+10*2%10-20*3; printf("%d",a); }</pre>	
<pre>#include <stdio.h> void main() { int i= -1 ,j; j= i && i++; printf("%d %d",i,j); }</pre>		<pre>#include<stdio.h> void main() { int i=10; printf("%d",i+1); printf("\n%d",i++); printf("\n%d",i); }</pre>	

<pre>#include <stdio.h> int main() { int i; i=!(0==0)?1:0; printf("%d",i); return 0; }</pre>		<pre>#include <stdio.h> int main() { int a=0,b=5,c; c=!a>b; printf("%d",c); }</pre>	
<pre>#include<stdio.h> void main() { int i=3,j=6; printf("%d",i++ +j); }</pre>		<pre>#include<stdio.h> void main() { printf("%d",2^3); }</pre>	
<pre>#include<stdio.h> int main() { int i=- 2; printf("%d",i); return 0; }</pre>		<pre>#include<stdio.h> int main() { int i=12; i=i>>4; printf("%d",i); return 0; }</pre>	

Module 3

if else statements

- if statement
- if statement with 1 instruction
- if statement with more than 1 instruction ➤ if else statements
- Nested if else statements
- Ways of writing if else statements
- else if clause
- Logical operators in if else
- Avoid these commonly made mistakes ➤ Conditional operator in place of if else

Value of the expression if(6)

if(-4)

if(0.2)

if('a')

if(0)

if(-1+1) if(5==5) if(5==6)

if(5>2)

if((5>8) if(5>4&&6>8) if(6>4||4>6)

Program

Output

<pre>#include<stdio.h> void main() { int a=45; if(a>40) printf("Excess"); }</pre>	
<pre>#include<stdio.h> void main() { int a=39; if(a>40) printf("Excess"); }</pre>	
<pre>#include<stdio.h> void main() { int a=39; if(a>40) printf("1"); if(a<40) printf("2"); if(a=40) printf("3"); if(a==40) printf("4"); }</pre>	

if statement with more than 1 instruction

<pre>#include<stdio.h> void main() { int a=39; if(a>40) printf("1"); if(a<40) printf("2"); if(a=40) printf("3"); if(a==41) printf("4"); printf("%d",a); }</pre>	
---	--

Program

Output

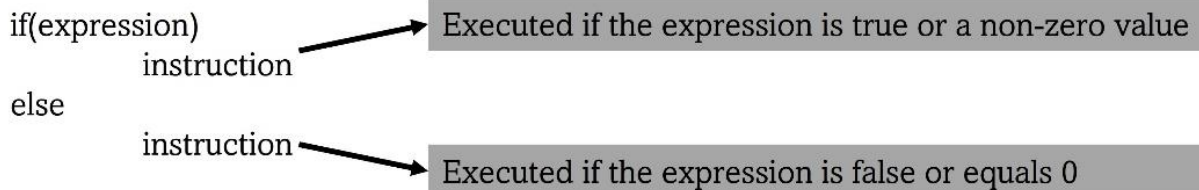
<pre>#include<stdio.h> void main() { int a=39; if(a<40) { a+=a; printf("%d",a); } }</pre>	
--	--

<pre>#include<stdio.h> void main() { int a=40; if(a<40) { a+=a; printf("%d",a); } if(a==40) { a*=a; printf("%d",a); } }</pre>	
---	--


```
#include<stdio.h> void
main()
{
int a=40; if(a<40) {
a+=a;
printf("%d",a); }
if(a=10) {
a*=a;
printf("%d",a); }
}
```

if else statements

28



Program:

A student will be allowed to take part in the test if his attendance percentage is more than 80%. Write a program to show whether the student will be allowed to take the test or not. Get the total number of classes and the number of classes attended by the student from the user.

<pre>#include<stdio.h> int main() { int tot,att; int p; printf("Enter the total number of classes: "); scanf("%d",&tot); printf("Enter the number of classes attended by the student: "); scanf("%d",&att); p=att/tot*100; printf("Attendance percentage is %d, so ",p); if(p>80) printf("allow."); else printf("don't allow."); }</pre>	<pre>#include<stdio.h> int main() { float tot,att; float p; printf("Enter the total number of classes: "); scanf("%f",&tot); printf("Enter the number of classes attended by the student: "); scanf("%f",&att); p=att/tot*100; printf("Attendance percentage is %f, so ",p); if(p>80) printf("allow."); else printf("don't allow."); }</pre>
<pre>Enter the total number of classes: 100 Enter the number of classes attended by the student: 90 Attendance percentage is 0, so don't allow.</pre>	<pre>Enter the total number of classes: 100 Enter the number of classes attended by the student: 90 Attendance percentage is 90.000000, so allow.</pre>
<pre>Enter the total number of classes: 100 Enter the number of classes attended by the student: 70 Attendance percentage is 0, so don't allow.</pre>	<pre>Enter the total number of classes: 100 Enter the number of classes attended by the student: 70 Attendance percentage is 70.000000, so don't allow.</pre>

Program:

Write a program to show the square of a number if the number is greater than 10, or the cube of the number if the number is less than 10. Get the number as input from the user.

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

```
#include<math.h>
```

```
int main()
```

```
{
```

```
int num;
```

```
printf("Enter a number: "); scanf("%d", &num);
```

```
if(num>10)
```

```

{
num=pow(num,2);
printf("Square of the number is: %d",num);
} else {
num=pow(num,3);
printf("Cube of the number is: %d",num);
} }

```

Exercise Programs:

1. Write a program to show the toll charges to be levied on a vehicle that crosses a toll booth, based on in which state it is registered. Get input from the user about the state in which the vehicle is registered.

State Code Karnataka K Kerala R Tamil Nadu T Andhra Pradesh A Telangana G

Toll Charges Rs. 200 Rs. 250 Rs. 300 Rs. 240 Rs. 220

2. Write a program to show whether a student is qualified in an exam based on the percentage of marks scored by him. If the student is qualified, show 1 in the output. If the student is not qualified, show 0 in the output. Get the marks obtained, maximum marks, and the percentage to qualify the exam from the user.

Program:

Input 1 (Graduation level) Post-Graduate Post-Graduate
Under-Graduate Under-Graduate

```
#include <stdio.h> void main()  
{
```

Input 2 (Experience in years) More than 10 years
Less than or equal to 10 years More than 10 years

Less than or equal to 10 years

Output (Salary) Rs. 45000/- Rs. 35000/- Rs. 40000/- Rs.
30000/-

```
char grad;  
int ex;  
printf("Enter P for post-graduate\nEnter U for  
undergraduate\n"); scanf("%c",&grad);  
printf("Experience in years: ");  
scanf("%d",&ex);  
if(grad=='P')  
{  
if(ex>10)  
{  
printf("Salary is Rs. 45000/-");  
} else {  
printf("Salary is Rs. 35000/-"); }  
} }
```

Program

Ways of writing if else statements

```
} else
```

```
{
if(ex>10)

{
printf("Salary is Rs. 40000/-");
} else {
printf("Salary is Rs. 30000/-"); }
30
```

Output

<pre>#include <stdio.h> void main() { int a=5; if(a==5) printf("5"); }</pre>	
<pre>#include <stdio.h> void main() { int a=5; if(a==5) { a+=4; printf("%d",a); } }</pre>	

<pre>#include <stdio.h> void main() { int a=6; if(a==5) printf("Equal"); else printf("Not Equal"); }</pre>	
<pre>#include <stdio.h> void main() { int a=6; if(a==5) { a-=3; printf("%d",a); } else { a+=3; printf("%d",a); } }</pre>	

else if clause

Program:

Write a program to show whether a student has passed the exam or failed the exam based on the given table.

Input (Grade) A

B

C

D

E

F

G

Output (Result) Pass

Pass

Pass

Pass

Fail

Fail

Fail

31

<pre>#include <stdio.h> void main() { int a=6; if(a==5) printf("Not Equal"); else { a+=3; printf("%d",a); } }</pre>	
<pre>#include <stdio.h> void main() { int a=6; if(a=5) { a+=5; printf("%d",a); } else printf("%d",6); }</pre>	

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

```
void main()
```

```
{
```

```

char grade;
printf("Enter the grade (A or B or C or D or E or F or G):
"); scanf("%c",&grade);
if(grade=='A')
printf("Pass");
else if(grade=='B')
printf("Pass");
else if(grade=='C')
printf("Pass");
else if(grade=='D')
printf("Pass");
else if(grade=='E')
printf("Fail");
else if(grade=='F')
printf("Fail");
else if(grade=='G')
printf("Fail");

}

```

Operator Program

Output

Logical operators in if else

32

&& Logical AND	#include <stdio.h> void main() { int a=5,b=6,c=7,d=9; if((a<b)&&(c<d)) printf("1"); else printf("0"); }	
Logical OR	#include <stdio.h> void main() { int a=5,b=6,c=7,d=9; if(a>b c<d) printf("1"); else printf("0"); }	

! Logical NOT	<pre>#include <stdio.h> void main() { int a=5,b=6,c=7,d=9; if(!(a>b c<d)) printf("1"); else printf("0"); }</pre>	
---------------	---	--

Program: Write a program to show the ticket price for the circus based on the age of the audience.

Input (Age) Age < 10
10 <=Age<40 40<=Age<=120

Output (Ticket Price) Rs. 100
Rs. 150
Rs. 200

```
#include <stdio.h> void main()
{
int age;
printf("Age is: "); scanf("%d", &age);
if(age>120) printf("Undefined"); else
if(age>=40) printf("200");
else if(age>=10)
printf("150"); else
if(age<10) printf("100");
}
```

```
#include <stdio.h> void main()
{
int age;
printf("Age is: "); scanf("%d", &age);
if(age<10)
printf("100"); if(age>=10&&age<40)
printf("150"); if(age>=40&&age<=120)
printf("200");
if(age>120) printf("Undefined");
}
```

Program:

Write a program to show whether a candidate can be selected for the job position based on the given eligibility criteria.

Input (Graduation) B.E

B.E

B.E

B.E

M.E

M.E

M.E

M.E

```
#include <stdio.h> void main()
```

```
{
```

Input (Age) 25<Age<30 25<Age<30 30<=Age<40

30<=Age<40 27<Age<32 27<Age<32 32<=Age<42

32<=Age<42

Input (Experience) <5

>5

>8

<8

<5

>5
>8
<8

Output (Result) Reject Select Select Reject Reject Select
Select Reject

33

```
char g;  
int a,e;  
printf("Enter B for B.E\nEnter M for M.E\n");  
scanf("%c",&g);  
printf("Enter age: ");  
scanf("%d",&a);  
printf("Enter experience: ");  
scanf("%d",&e);
```

```
if((g=='B'&&a>25&&a<30&&e>5)||  
(g=='B'&&a>=30&&a<=40&&e>8)||  
(g=='M'&&a>27&&a<32&&e>5) ||  
(g=='M'&&a>=32&&a<=42&&e>8))  
printf("Select"); else printf("Reject");  
}
```

Exercise Program:

Write a program to show salary range for a candidate based on his performance in the interview.

Input (Score in Aptitude) Max Mark - 60 >30

>30

>30

>30

>30

>30

>=15 and >=15 and >=15 and >=15 and >=15 and
>=15 and

<15 <15 <15 <15 <15 <15

Input Input (Score in Coding) (Graduation)

Max Mark - 20

>10 M.E >10 B.E

>=5 and <=10 M.E >=5 and <=10 B.E <5 M.E <5 B.E

>10 M.E

>10 B.E >=5 and <=10 M.E >=5 and <=10 B.E

<5 M.E

<5 B.E >10 M.E >10 B.E

>=5 and <=10 M.E >=5 and <=10 B.E <5 M.E <5 B.E

Problem:

Input Size : $N \leq 1000000$

Sample Testcase :

INPUT

2 4

OUTPUT

4

2) Given the radius of a circle:
Find its area

Sample Testcase:

Input:

4

Output:

50.24

3) Given numbers A,B find A^B .

Input Size : $1 \leq A \leq 5 \leq B \leq 50$

Sample Testcase :

INPUT

3 4

OUTPUT

81

4) Given a number N and an array of N elements, find the Bitwise OR of the array elements.

Input Size : $N \leq 100000$

Sample Testcase :

INPUT

2

2 4

OUTPUT

6

5) Given 3 numbers A,B,C print 'Yes' if they can form the sides of a right angled triangle, otherwise 'No'.

Input Size : $A,B,C \leq 100000$

Sample Testcase :

INPUT

3 4 5

OUTPUT

Yes

6) Given 3 numbers N , L and R. Print 'yes' if N is

between L and R else print 'no'.

Sample Testcase : 1

INPUT

3

2 6

OUTPUT

yes

Sample Testcase : 2

INPUT

8

2 6

OUTPUT

no

7) Given 2 numbers N and M add both the numbers and check whether the sum is odd or even.

Sample Testcase :

INPUT

9 2

OUTPUT

odd

Sample Testcase :

INPUT

11 23

OUTPUT

even

Sample Testcase :

INPUT

-9 2

OUTPUT

odd

8) Given 2 numbers N, M find the GCD of N and M. If it cannot be found for given number(s) then print -1

Sample Testcase :

INPUT
10 5
10 = 1,2,5,10
5 = 1,5
OUTPUT
5

Sample Testcase :

INPUT
10 11
OUTPUT
-1

9) In a garage the service man takes 10 minutes to service one car. If there are N cars in garage and X is number of minutes after which one person arrives, Calculate how much time last person has to wait in garage. (Print answer in minutes)

Input Description:

You are given Two numbers 'N' and 'X'

Output Description:

Waiting time of last person

Sample Input :

4 5

Sample Output :

15

Find out the perfect number using c program

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

```
int main(){  
    int n,i=1,sum=0;  
    printf("Enter a number: ");  
    scanf("%d",&n);  
    while(i<n){  
        if(n%i==0)  
            sum=sum+i;  
        i++;  
    } if(sum==n)
```

```

    printf("%d is a perfect number",i);
else
    printf("%d is not a perfect number",i);
return 0;
}

```

Sample output:

Enter a number: 6

6 is a perfect number

Code 2:

1. C program to find perfect numbers
2. C perfect number code
3. Perfect number program in c language

```

#include<stdio.h>
int main(){
    int n,i,sum;
    int min,max;
    printf("Enter the minimum range: ");
    scanf("%d",&min);

    printf("Enter the maximum range: ");
    scanf("%d",&max);
    printf("Perfect numbers in given range is: ");
    for(n=min;n<=max;n++){
i=1; sum = 0;

        while(i<n){
            if(n%i==0)
                sum=sum+i;
            i++;
        }

        if(sum==n)
            printf("%d ",n);
    }

    return 0; }

```


Sample output:

Enter the minimum range: 1

Enter the maximum range: 20

Perfect numbers in given range is: 6

Code 3:

3. C program to print perfect numbers from 1 to 100

```
#include<stdio.h>
int main(){
int n,i,sum;

    printf("Perfect numbers are: ");
    for(n=1;n<=100;n++){
i=1; sum = 0;

while(i<n){

if(n%i==0)
    sum=sum+i;
i++; }

    if(sum==n)
        printf("%d ",n);
}

return 0; }
```

Output:

Perfect numbers are: 6 28

Definition of perfect number or what is perfect number?

Perfect number is a positive number which sum of all positive divisors excluding that number is equal to that number. For example 6 is perfect number since divisor of 6 are 1, 2 and 3. Sum of its divisor is

$$1 + 2 + 3 = 6$$

Note: 6 is the smallest perfect number.

Next perfect number is 28 since $1 + 2 + 4 + 7 + 14 = 28$

Some more perfect numbers: 496, 8128

Check the given number is Armstrong number or not using c program

Code 1:

1. Warp to check a number is Armstrong
2. C program to check whether a number is Armstrong or not
3. Simple c program for Armstrong number
4. Armstrong number in c with output

```
#include<stdio.h>
int main(){
    int num,r,sum=0,temp;
    printf("Enter a number: ");
    scanf("%d",&num);
    temp=num;
    while(num!=0){
        r=num%10;
        num=num/10;
        sum=sum+(r*r*r);
    }
    if(sum==temp)
        printf("%d is an Armstrong number",temp);
    else
        printf("%d is not an Armstrong number",temp);
    return 0;
}
```

Sample output:

Enter a number: 153

153 is an Armstrong number

Definition of Armstrong number or what is an Armstrong number:

Definition according to c programming point of view:
Those numbers which sum of the cube of its digits is

equal to that number are known as Armstrong numbers. For example 153 since $1^3 + 5^3 + 3^3 = 1 + 125 + 9 = 153$

Check given number is prime number or not using c program

Definition of prime number:

A natural number greater than one which has not any other divisors except 1 and itself is called prime number. In other word we can say which has only two divisors 1 and number itself. For example: 5

Their divisors are 1 and 5.

Note: 2 is only even prime number.
Logic for prime number in c

We will take a loop and divide number from 2 to number/2. If the number is not divisible by any of the numbers then we will print it as prime number.

Example of prime numbers

: 2, 3, 5, 7, 11, 13, 17, 19,
23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79,
83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137,
139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193,
197, 199 etc.

```
#include<stdio.h>
int main(){
int num,i,count=0;
```

```
printf("Enter a number: ");
scanf("%d",&num);
for(i=2;i<=num/2;i++){
if(num%i==0){
count++;

break;
}
else
}

if(count==0 && num!= 1)
printf("%d is a prime number",num);
printf("%d is not a prime number",num);
return
0;
}
```

Sample Output:

Enter a number: 5
5 is a prime number

Write a program to check the palindrome or not

TestCase:1

Input:121

Output: Palindrome

TestCase:2

Input:12321

Output: Palindrome

TestCase: 3

Input: 12341

Output: Not a Palindrome

Write a program to find an Armstrong number

TestCase 1:

Input: 153

Output: Armstrong number

TestCase2:

Input: 1634

Output: Armstrong number

TestCase3:

Input: 123

Output: Not an Armstrong number

2. Write a program to find the given number is prime or not

TestCase: 1

Input: 11

Output: Prime

TestCase: 2

Input: 22

Output: Not a Prime

TestCase:3

Input: 1

Output: Not a Prime