**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

Other Armstrong numbers: 370,371,407 etc.

**In general definition:**

Those numbers which sum of its digits to power of number of its digits is equal to that number are known as Armstrong numbers.

Example 1: 153

Total digits in 153 is 3

And 1^3 + 5^3 + 3^3 = 1 + 125 + 27 = 153

Example 2: 1634

Total digits in 1634 is 4

And 1^4 + 6^4 + 3^4 +4^4 = 1 + 1296 + 81 + 64 =1634

Examples of Armstrong numbers: 1, 2, 3, 4, 5, 6, 7, 8, 9, 153, 370, 371, 407, 1634, 8208, 9474, 54748, 92727, 93084, 548834, 1741725

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

**The time complexity of a program that determines Armstrong number is**: O (Number of digits)

Code 2:

1. **Write a c program for Armstrong number**

2. **C program for Armstrong number generation**

3. **How to find Armstrong number in c**

4. **Code for Armstrong number in c**

#include<stdio.h>

int main(){

    int num,r,sum,temp;

    int min,max;

    printf("Enter the minimum range: ");

    scanf("%d",&min);

    printf("Enter the maximum range: ");

    scanf("%d",&max);

    printf("Armstrong numbers in given range are: ");

    for(num=min;num<=max;num++){

         temp=num;

         sum = 0;

         while(temp!=0){

             r=temp%10;

             temp=temp/10;

             sum=sum+(r\*r\*r);

         }

         if(sum==num)

             printf("%d ",num);

    }

    return 0;

}

Sample output:

Enter the minimum range: 1

Enter the maximum range: 200

Armstrong numbers in given range are: 1 153

Code 3:

1. **Armstrong number in c using for loop**

#include<stdio.h>

int main(){

    int num,r,sum=0,temp;

    printf("Enter a number: ");

    scanf("%d",&num);

    for(temp=num;num!=0;num=num/10)

{

         r=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: 370

370 is an Armstrong number

Logic of Armstrong number in c

Code 4:

1. **C program to print Armstrong numbers from 1 to 500**

2. **C program for finding Armstrong numbers**

#include<stdio.h>

int main(){

    int num,r,sum,temp;

    for(num=1;num<=500;num++){

         temp=num;

         sum = 0;

         while(temp!=0){

             r=temp%10;

             temp=temp/10;

             sum=sum+(r\*r\*r);

         }

         if(sum==num)

             printf("%d ",num);

    }

    return 0;

}

Output:

1 153 370 371 407