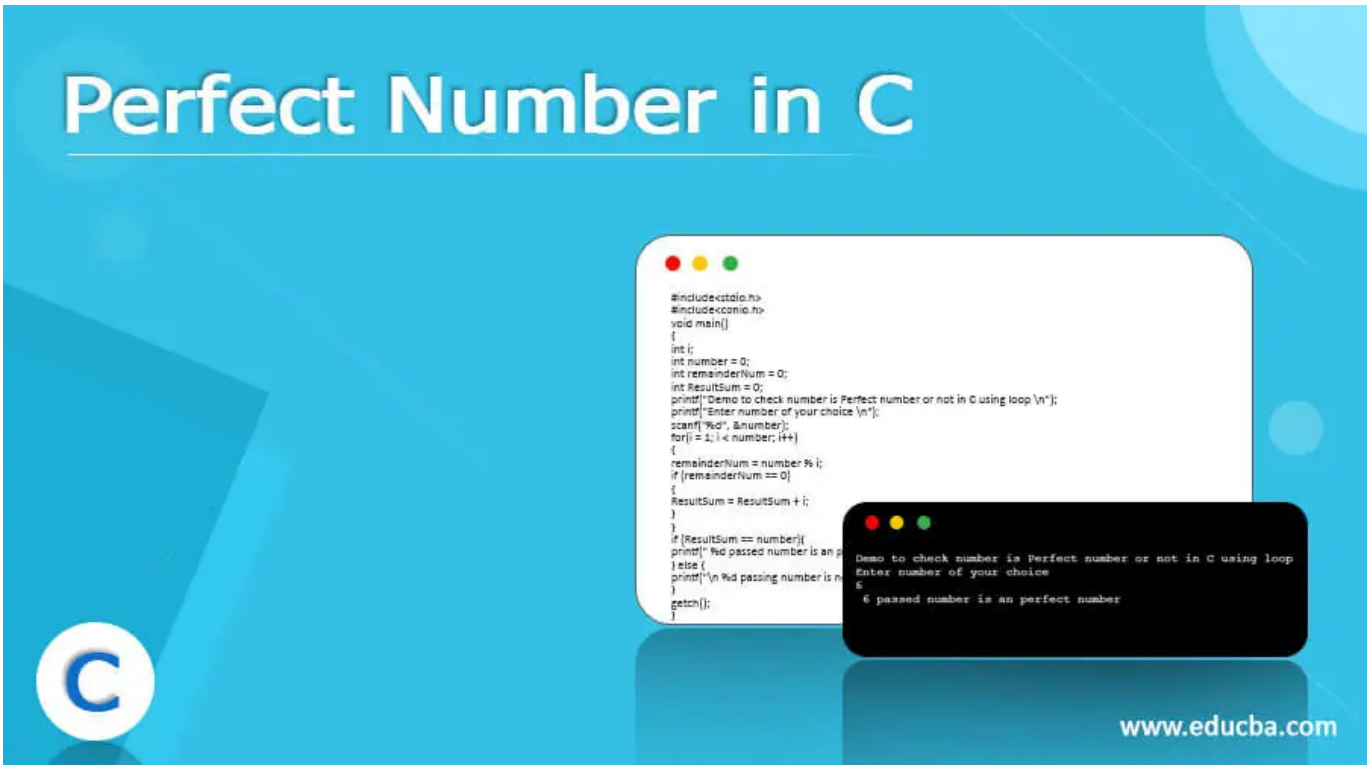
Perfect Number in C

(https://www.educba.com/software-development/)

← (https://www.educba.com/leap-year-program-in-c/)

→ (https://www.educba.com/anagram-program-in-c/)



Definition of C Perfect Number

In General, we have Perfect number, a number whose sum is equal to the sum of its div the number cannot be included. It is a positive integer, we can write the logic in any language which can give us the Perfect number, or also we can check if the number is perfect or not. It

should be a positive integer, but also keep in mind that the number should be divisible by all the numbers we are using, after that the sum should be equal to the number we want to check. If any of the rules break, then those numbers cannot come under the Perfect number category. In C we can write a program to check whether the passing number is a perfect number or not, also we can check this from the series of numbers we have. In mathematical term, its definition will always be the same, but we can implement the logic in any way we want.

Logic behind Perfect number

As we have already discussed, that Perfect number, in general, is the sum of all the divisors of the number but we cannot include the number itself. IF the sum is the same as the number then we can say it is a Perfect Number. But if we talk about the logic, then it will be the same in C or in any other programming language, but the thing is we just have to write the correct logic in order to check whether the number is Perfect number or not. In this section of the tutorial we will see how we can check the number is Perfect or not, to make this simple we will see one simple example to understand it better see below;

1) Suppose the number to be check is 6: The user has provided one number that is ‘6’, now we have to think and calculate by which other numbers we can divide ‘6’ and get the remainder as ‘0’. If the remainder is zero, then we can say that ‘6’ is divisible by that number and we can take this number into consideration to get the sum by using some more other numbers.

In the case of ‘6’. let’s first start with number 1, so it will be divisible by 1.

2) now 2, it will also divide the number 6.

3) Now 3 it will also do the same.

4) and then we can check for other values as well if the sum is not yet meet.

5) But in our case it is already 6 by using 1,2 and 3. So that we can say 6 is a Perfect number.

We also have many numbers which are not Perfect number let’s take a look at number 4 itself;

it would be divisible by 1, and 2 no other number but the sum of 1+2 =3 that means it is not a Perfect number because the sum of the division is not matching with the number we want to check. So by the use of C language, we can write a simple program where we can test the number is Perfect number or not. In the coming section of the tutorial, we will see what steps we need to follow in order to write the program in C language which will check the number for us, and make it simple for beginners to understand.

How to Check Perfect number in C?

As now we already know that what are Perfect numbers are, we just need to now focus on the steps that we need to take up in order to get our program working and check if the number is Perfect number or not. We can do this in C language by the use of loops and also we need to check that whether the number is divisible or not, with more conditions, let's try to have closer look at the steps needed to create the program in C language are as follows see below;

1) We will make us of for loop in order to get the result here. This for loop will be going to iterate less than the number because we do not have to consider the number itself only the values which are less than it and should be positive.

2) Inside the loop we will calculate the remainder by taking the mod of it. In C we can do this by following the below piece of code, for reference see below;

e.g. :

remainder = number_to_check % loop_value (i);

As you can see we are trying to divide the number we have passes with the for loop 'i' value, also we are trying to store that value inside the temp variable we have that is remainder.

3) If the value of the remainder is '0' then it means the number is divided by the given number passed.

4) If then we can move to the other value.

5) Inside the if block we have to keep checking the sum of the divisible numberers, for this we can make a temporary variable that will calculate the sum and ready with the result.

6) at the end we have to check if the sum is equal to the number we have passed to che
Perfect number, if the value matched then we will return true if both numbers do not match then it means the given number passed is not Perfect number.

7) return result and exit

Examples

(https://www.educba.com/software-development/)

Simple example to implement Perfect number in C.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i;
int number = 0;
int remainderNum = 0;
int ResultSum = 0;
printf("Demo to check number is Perfect number or not in C using loop \n");
printf("Enter number of your choice \n");
scanf("%d", &number);
for(i = 1; i < number; i++)
{
remainderNum = number % i;
if (remainderNum == 0)
{
ResultSum = ResultSum + i;
}
}
if (ResultSum == number){
printf(" %d passed number is an perfect number ", number);
} else {
printf("\n %d passing number is not perfect number", number);
}
```

```
getch();
}
```


Output:

Conclusion

By the use of it, we can make a check to a number if it is a Perfect number or not. The definition for Perfect number will always be the same, only the difference will be in writing the logic and

the language we used to implement this.

Recommended Articles

This is a guide to Perfect Number in C. Here we discuss the Definition, syntax, How to check Perfect Number in C? examples with code implementation. You may also have a look at the following articles to learn more –

1. DB2 row_number (https://www.educba.com/db2-row_number/)
2. SQL ROW_NUMBER (https://www.educba.com/sql-row_number/)
3. JavaScript Number (https://www.educba.com/javascript-number/)
4. DB2 row_number (https://www.educba.com/db2-row_number/)