026 -
2'S COMPLEMENT IN C

The 2s complement in C is generated from the 1s complement in C.

As we know that the 1s complement of a binary number is created by transforming bit 1 to 0 and 0 to 1; the 2s complement of a binary number is generated by adding one to the 1s complement of a binary number.

DETAILED ALGORITHM

First, we input the number of bits, and it gets stored in the 'n' variable.

After entering the number of bits, we declare character array, i.e., char binary[n+1], which holds the binary number.

The 'n' is the number of bits which we entered in the previous step; it basically defines the size of the array.

We declare two more arrays, i.e., onescomplement[n+1], and twoscomplement[n+1].

The onescomplement[n+1] array holds the ones complement of a binary number while the twoscomplement[n+1] array holds the two's complement of a binary number.

Initialize the carry variable and assign 1 value to this variable.

After declarations, we input the binary number.

Now, we simply calculate the one's complement of a binary number. To do this, we create a loop that iterates throughout the binary array, for(int i=0;i<n;i++).

In for loop, the condition is checked whether the bit is 1 or 0. If the bit is 1 then onescomplement[i]=0 else onescomplement[i]=1.

In this way, one's complement of a binary number is generated.

After calculating one's complement, we generate the 2s complement of a binary number.

To do this, we create a loop that iterates from the last element to the starting element. In for loop, we have three conditions:

> If the bit of onescomplement[i] is 1 and the value of carry is 1 then we put 0 in twocomplement[i].

> If the bit of onescomplement[i] is 0 and the value of carry is 1 then we put 1 in twocomplement[i] and 0 in carry.

> If the above two conditions are false, then onescomplement[i] is equal to twocomplement[i].