

Page No.			
Date			

* Assignment No. 8 *

Aim:-

write C++ program for storing binary number using doubly linked list. write function to:

- ① to compute 1's & 2's complement.
- ② Add two binary number.

Objective:-

- ① to understand the concept of linked list.
- ② to implement the concept of doubly-linked list.
- ③ to implement function for computing 1's & 2's complement of binary number.

Hardware / Software required:-

Computer lab, Eclipse with C++.

Theory:-

Binary Number:-

Binary number is a number, expressed in the base-2 numeral system or binary numeral system, a method of representation or mathematical expression which uses only two number or bit called 0 and 1's complement.

The one's complement of a binary number is the value obtained by inverting all the bit in the binary representation of the number (swapping 0's and 1's).

1010 \rightarrow 1's complement = 0101

1100 \rightarrow 1's complement = 0011

2's Complement:-

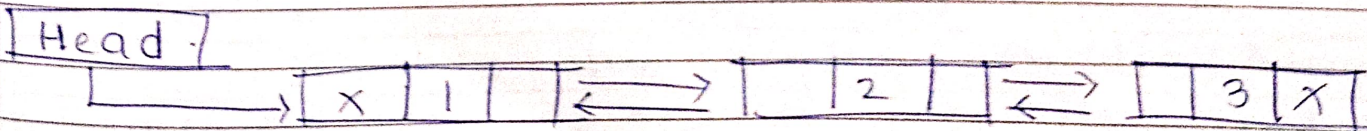
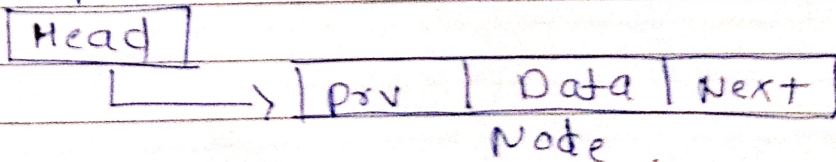
1st complement of binary number is 1, added to the 1's complement of the binary number. In the 2's complement representation of binary number, the msb represent the sign with 0 '0' used for plus sign and 1 '1' used for a minus sign. the remaining bit are used to represent magnitude.

$$0111 \rightarrow 2's \text{ complement} = 1001$$

$$1100 \rightarrow 2's \text{ complement} = 0100$$

Doubly linked list:-

Doubly linked list is a complex type of linked list in which a node contains, a pointer to the previous as well as the next node in the sequence therefore, in a doubly linked list, a node consists of three parts. Node data, pointer to the next node in the sequence & pointer to the previous node in the sequence.



Memory Representation of a doubly linked list:

Generally, doubly linked list consumes, more space for every node and therefore, calls more expansive basic operations such as insertion

& deletion however, we can easily manipulate the element of the list since the list maintain pointer in both the direction.

Algorithm:-

Step 1 - Start.

Step 2: Define a function to compute 1's complement.

int onesComplement(int n) {

int number_of_bits = floor(log₂(n)+1);

return (1 << number_of_bits - 1) ^ n;

}

Step 3: Define a function to compute 2's complement of given binary number. (ones complement + 1)

Step 4: Define a function to add two binary no.

1 + 1 = 0 with carry 1

~~1 + 0 = 1~~

0 + 0 = 0

Step 5: Input a binary number and call the functions to perform a desired operation.

Step 6: Display the desired output.

Step 7: Stop.

Conclusion:-

In this way, we have studied & implemented doubly linked list for binary addition & to compute 1's & 2's complement of the same.

©
M