Decimal Equivalent of Binary Linked List

Given a singly linked list of 0s and 1s find its decimal equivalent.

```
Input  : 0->0->0->1->1->0->0->1->0
Output : 50

Input  : 1->0->0
Output : 4
```

Decimal Value of an empty linked list is considered as 0.

Initialize result as 0. Traverse the linked list and for each node, multiply the result by 2 and add node's data to it.

```
// C++ Program to find decimal value of
// binary linked list
#include<iostream>
using namespace std;

/* Link list Node */
struct Node
{
    bool data;
    struct Node* next;
};

/* Returns decimal value of binary linked list */
int decimalValue(struct Node *head)
{
    // Initialized result
    int  res = 0;

    // Traverse linked list
    while (head != NULL)
    {
        // Multiply result by 2 and add
        // head's data
        res = (res  << 1) + head->data;

        // Move next
        head = head->next;
    }
    return res;
}

// Utility function to create a new node.
Node *newNode(bool data)
{
    struct Node *temp = new Node;
    temp->data = data;
    temp->next = NULL;
    return temp;
}

/* Drier program to test above function*/
int main()
{
    /* Start with the empty list */
    struct Node* head = newNode(1);
    head->next = newNode(0);
    head->next->next = newNode(1);
    head->next->next->next = newNode(1);

    cout << "Decimal value is "
         << decimalValue(head);

    return 0;
}
```

Output :

```
Decimal value is 11
```