**Practical No:-8**

**Name - BANKHELE YASH DATTATRAY**

**Roll No -04**

**Exam Seat No - S190842009**

Title:-

Write c++ program for storing binary number using doubly linked lists. Write functions:

a) To compute 1ns and 2ns complement

b) Add two binary numbers.

INPUT:-

#include<iostream>

using namespace std;

class binary;

class node

{

public:

node \*prev;

bool n;

node\*next;

node()

{

prev=next=NULL;

}

node(bool b)

{

n=b;

prev=next=NULL;

}

friend class binary;

};

class binary

{

node \*start;

public:

binary()

{

start=NULL;

}

void generateBinary(int no);

void displayBinary();

void onesComplement();

void twoscomplement();

binary operator +(binary n1);

bool addBitAtBegin(bool val)

{

node \*nodee=new node(val);

if(start==NULL)

{

start=nodee;

}

else

{

nodee->next=start;

start->prev=nodee;

start=nodee;

}

return true;

}

};

void binary::generateBinary(int no)

{

bool rem;

node \*p;

rem=no%2;

start=new node(rem);

no=no/2;

while(no!=0)

{

rem=no%2;

no=no/2;

/\*

if(start==NULL)

{

start=new node(rem);

// cout<<" Start prev: "<<start->prev;

// cout<<" Start next: "<<start->next ;

}

else

{

\*/

p=new node(rem); p->next=start; start->prev=p;

// cout<<" Start prev: "<<start->prev->n;

// cout<<" p->n"<<p->n;

start=p;

//}

}

}

void binary::displayBinary()

{

node \*t;

t=start;

while(t!=NULL)

{

cout<<t->n;

t=t->next;

}

}

void binary::onesComplement()

{

node \*t;

t=start;

while(t!=NULL)

{

if(t->n==0)

t->n=1;

else

t->n=0;

t=t->next;

}

}

binary binary::operator +(binary n1)

{

binary sum;

node \*a=start;

node \*b=n1.start;

// bit \*s=sum.start;

bool carry=false;

while(a->next!=NULL)

a=a->next;

while(b->next!=NULL)

b=b->next;

while(a!=NULL && b!=NULL)

{

sum.addBitAtBegin((a->n)^(b->n)^carry);

carry=((a->n&& b->n) || (a->n&& carry) || (b->n && carry));

a=a->prev;

b=b->prev;

}

while(a!=NULL)

{

sum.addBitAtBegin(a->n^carry);

a=a->prev;

}

while(b!=NULL)

{

sum.addBitAtBegin(b->n^carry);

b=b->prev;

}

sum.addBitAtBegin(carry);

return sum;

}

void binary::twoscomplement()

{

onesComplement();

bool carry=1;

node \*t;

t=start;

while(t->next!=NULL)

{

t=t->next;

}

while(t!=NULL)

{

if(t->n==1&& carry==1)

{

}

else

t->n=0;

carry=1;

if(t->n==0&& carry==1)

{

}

else

t->n=1;

carry=0;

if(carry==0) break;

t=t->prev;

}

displayBinary();

}

int main()

{

int num,num1;

binary n1,n3,n2;

int choice=1;

do

{

cout<<"\n\n=========Binary Number Operations========\n";

cout<<"1. Generate binary\n2.One's Complement\n3.Two'sComplement\n4.Addition\n5.Exit\nEnter your choice: ";

cin>>choice;

switch(choice)

{

case 1:

cout<<"\nENter Number : ";

cin>>num;

n1.generateBinary(num);

cout<<"\nBinary Representation: ";

n1.displayBinary();

break;

case 2:

cout<<"\nENter Number in decimal form: ";

cin>>num;

n1.generateBinary(num);

cout<<"\nBinary Representation: ";

n1.displayBinary();

cout<<"\nOnes Complement: ";

n1.onesComplement();

n1.displayBinary();

break;

case 3:

cout<<"\nEnter Number in decimal form: ";

cin>>num;

n1.generateBinary(num);

cout<<"\nBinary Representation: ";

n1.displayBinary();

cout<<"\nTwos complement: ";

n1.twoscomplement();

break;

case 4:

cout<<"\nEnter Two Numbers: ";

cin>>num>>num1;

n1.generateBinary(num);

n2.generateBinary(num1);

n1.displayBinary();

cout<<" + ";

n2.displayBinary();

cout<<"= ";

n3=n1+n2;

n3.displayBinary();

}

}

while(choice!=5);

cout<<"\nEnd of Program";

return 0;

}

OUTPUT:-

jaihind@jaihind-ThinkCentre-M60e:~$ g++ yash8.cpp

jaihind@jaihind-ThinkCentre-M60e:~$ ./a.out

=========Binary Number Operations========

1. Generate binary

2.One's Complement

3.Two'sComplement

4.Addition

5.Exit

Enter your choice: 1

ENter Number : 2

Binary Representation: 10

=========Binary Number Operations========

1. Generate binary

2.One's Complement

3.Two'sComplement

4.Addition

5.Exit

Enter your choice: 2

ENter Number in decimal form: 31

Binary Representation: 11111

Ones Complement: 00000

=========Binary Number Operations========

1. Generate binary

2.One's Complement

3.Two'sComplement

4.Addition

5.Exit

Enter your choice: 3

Enter Number in decimal form: 34

Binary Representation: 100010

Twos complement: 011101

=========Binary Number Operations========

1. Generate binary

2.One's Complement

3.Two'sComplement

4.Addition

5.Exit

Enter your choice: 4

Enter Two Numbers: 23 12

10111 + 1100= 100011

=========Binary Number Operations========

1. Generate binary

2.One's Complement

3.Two'sComplement

4.Addition

5.Exit

Enter your choice: 5

End of Program