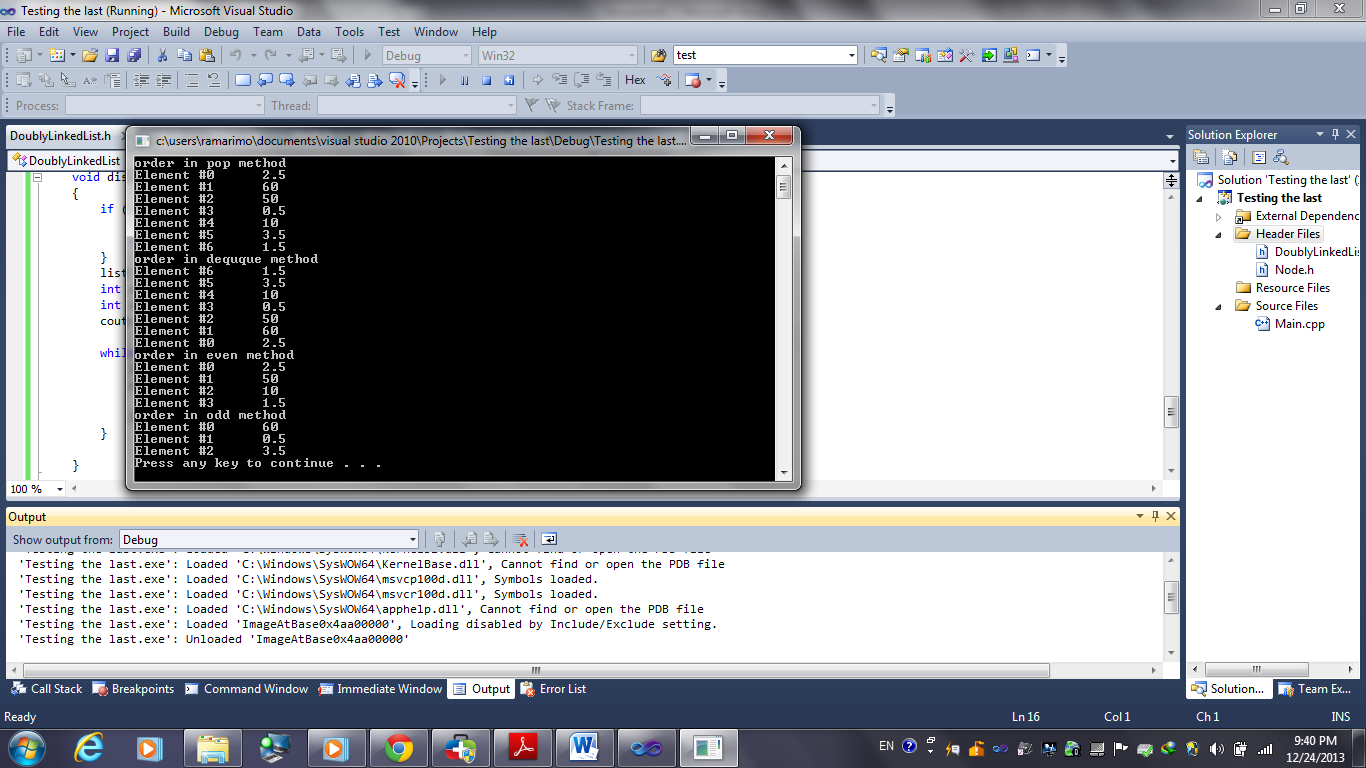
DS Report

Problem 1:

Screenshot of the result:



Sample code :

class DoublyLinkedList{

private:

Node\* head;

Node\* tail;

Node\* list;

int elementCount;

public:

DoublyLinkedList(){

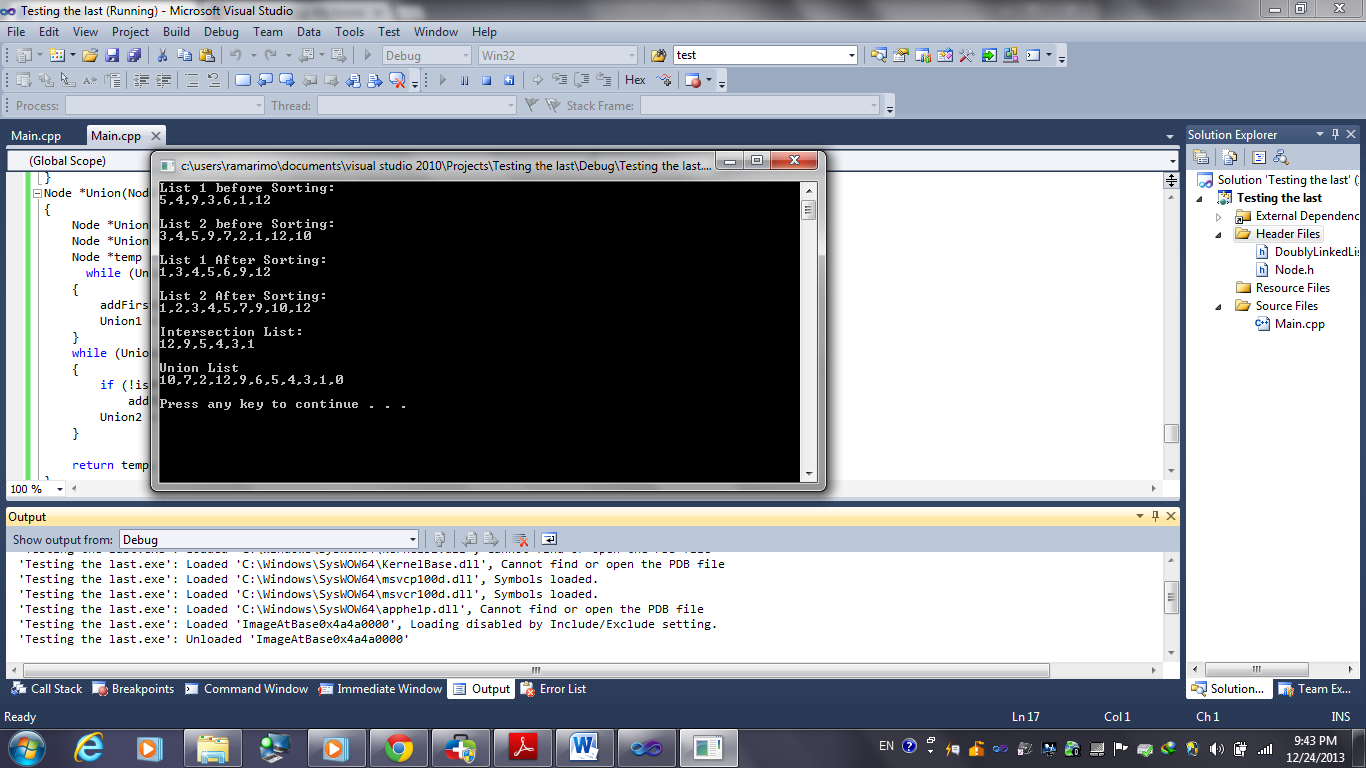
head = tail = list = NULL;

elementCount = 0;

}

Problem 2

Screenshot of the result:



Sample code :

void addLast (Node \*head, int data)

{

Node \*temp = head;

Node \*newHead = new Node();

newHead -> data = data;

while(temp)

{

if(temp -> next == NULL)

{

temp -> next = newHead;

newHead -> next = NULL;

return;

}

temp = temp -> next;

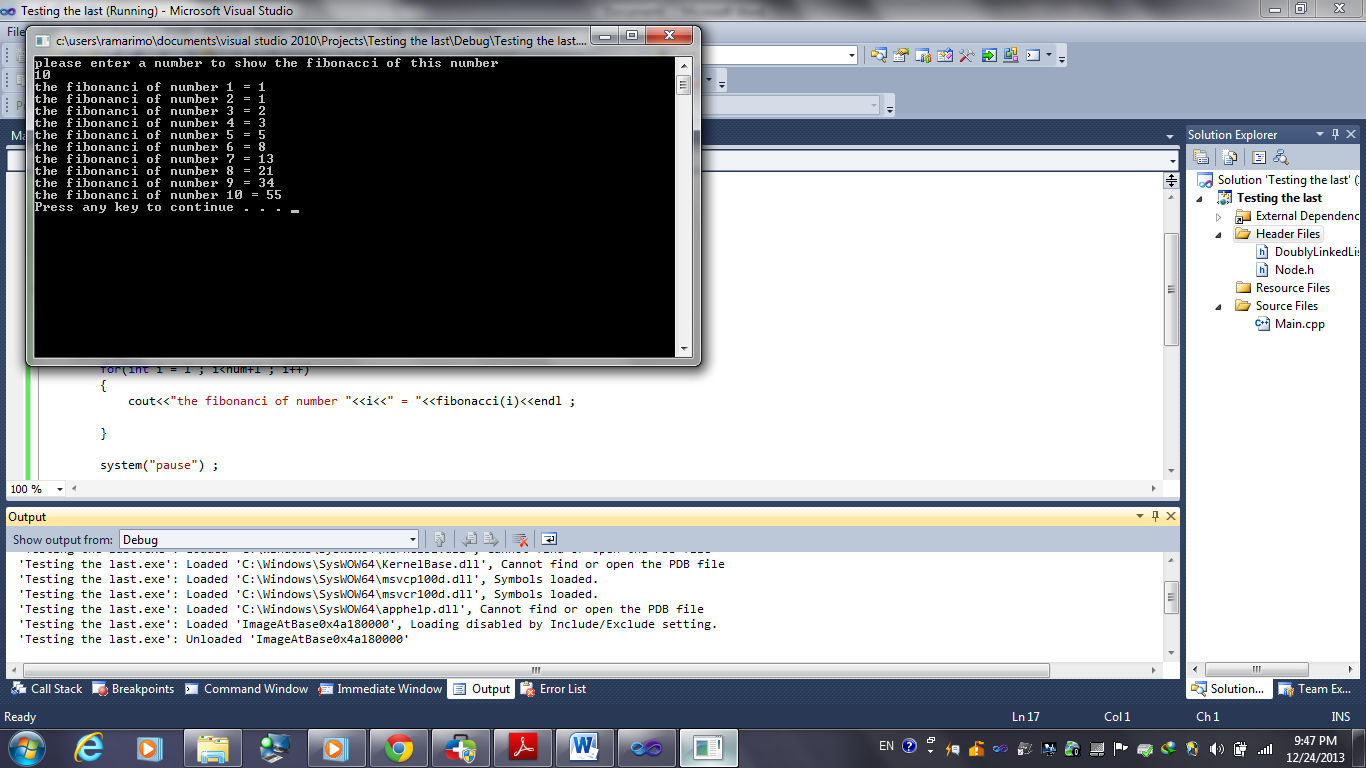
}

}

Problem 3:

Screenshot of the result:

In case of input 10



Sample code:

int fibonacci (int n)

{

if (n==0)

return 0 ;

else if (n==1)

return 1 ;

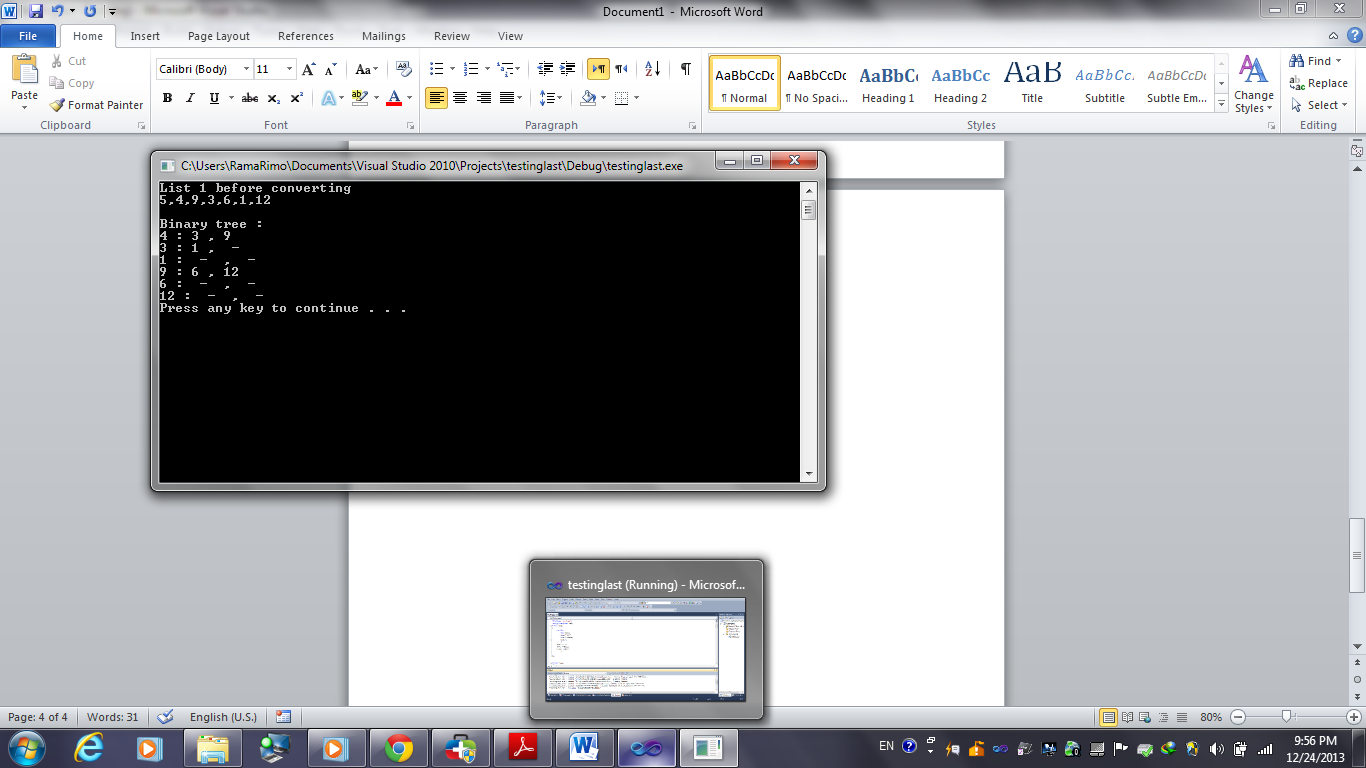
else

return (fibonacci(n-1)+fibonacci(n-2));

}

Problem 4:

Screenshot of the result:



Sample code :

void insertintobts(int value, node\* root)

{

if(value < root->data)

{

if(root->left == NULL)

{

root->left = new node();

root->left->data = value;

}

else

{

insertintobts(value, root->left);

}

}

else if(value >= root->data)

{

if(root->right == NULL)

{

root->right = new node();

root->right->data = value;

}

else

{

insertintobts(value, root->right);

}

}

}