TEAM:  
18BIT0143 SAUARBH SRIVASTAVA

18BIT0105 ABHIMANYU SINGH

Q.4

#include<iostream>

#include <fstream>

using namespace std;

struct node

{

string name;

node \*next;

node \*prev;

};

node\* head1 = NULL;

node\* head2 = NULL;

int isprime(int n)

{

if(n==1||n==2)

return 1;

for(int i=2;i<=n/2;i++)

{

if(n%i==0)

return 0;

else

continue;

}

return 1;

}

void insert1( string name)

{

if (head1 == NULL)

{

node\* new\_node = new node;

new\_node->name = name;

new\_node->next = new\_node;

new\_node->prev = new\_node;

head1 = new\_node;

return ;

}

node\* last = head1->prev;

node\* new\_node = new node;

new\_node->name = name;

new\_node->next = head1;

head1->prev = new\_node;

new\_node->prev = last;

last->next = new\_node;

}

void insert2( string name)

{

if (head2 == NULL)

{

node\* new\_node = new node;

new\_node->name = name;

new\_node->next = new\_node;

new\_node->prev = new\_node;

head2 = new\_node;

return ;

}

node\* last = head2->prev;

node\* new\_node = new node;

new\_node->name = name;

new\_node->next = head2;

head2->prev = new\_node;

new\_node->prev = last;

last->next = new\_node;

}

void Create()

{

ofstream F;

F.open("names.txt");

string name;

char choice;

do

{

cout<<"Enter Name: "<<endl;

cin>>name;

F<<name<<endl;

cout<<"More(Y/N)";cin>>choice;

}

while(choice!='N');

F.close();

}

void Inlist()

{

ifstream F;

F.open("names.txt");

char name[20];int count=0;

while(F.getline(name,20))

{

count++;

if(isprime(count))

{

insert1(name);

}

else

{

insert2(name);

}

cout<<endl;

}

F.close();

}

void display(node \*head)

{

node \*temp = head;

int flag = 0;

if(head==NULL)

{

cout<<"Empty list";

}

else

{

while(flag!=1)

{

if(temp->next!=head)

{

cout<<temp->name<<endl;

temp = temp->next;

}

else

{

flag = 1;

}

}

cout<<temp->name<<endl;

}

}

node\* middle()

{

int flag=0;

node\* temp1=head2,\*temp2=head2;

while(flag!=1)

{

if((temp2->next)->next==head2)

{

flag=1;

break;

}

else if(temp2->next!=head2)

{

temp2=(temp2->next)->next;

temp1=temp1->next;

}

else

{

flag=1;

break;

}

}

return temp1;

}

int count()

{

int c=1,flag=0;

node\*temp=head1;

while(flag!=1)

{

if(temp->next!=head1)

{

c++;

temp=temp->next;

}

else

{

flag=1;

}

}

return c;

}

int count2()

{

int c=1,flag=0;

node\*temp=head2;

while(flag!=1)

{

if(temp->next!=head2)

{

c++;

temp=temp->next;

}

else

{

flag=1;

}

}

return c;

}

void del()

{

node\*start;

start=middle();

cout<<"\nStarter: "<<start->name;

int m=count();

cout<<"\nCount: "<<m;

while(start!=NULL)

{

node\*temp=start;

if(start->next==start)

{

delete start;

head2=NULL;

break;

}

for(int i=0;i<m;i++)

{temp=temp->next;

}

if(start->next!=start)

{

start=temp->next;

(temp->prev)->next=temp->next;

(temp->next)->prev=temp->prev;

}

else

start=NULL;

temp->next=NULL;

temp->prev=NULL;

delete temp;

}

}

int main()

{

Create();

Inlist();

cout<<"Prime\n ["<<count()<<" numbers]: "<<endl;

display(head1);

cout<<"---------------------------------"<<endl;

cout<<"Not Prime\n ["<<count2()<<" numbers]: "<<endl;

display(head2);

cout<<"---------------------------------"<<endl;

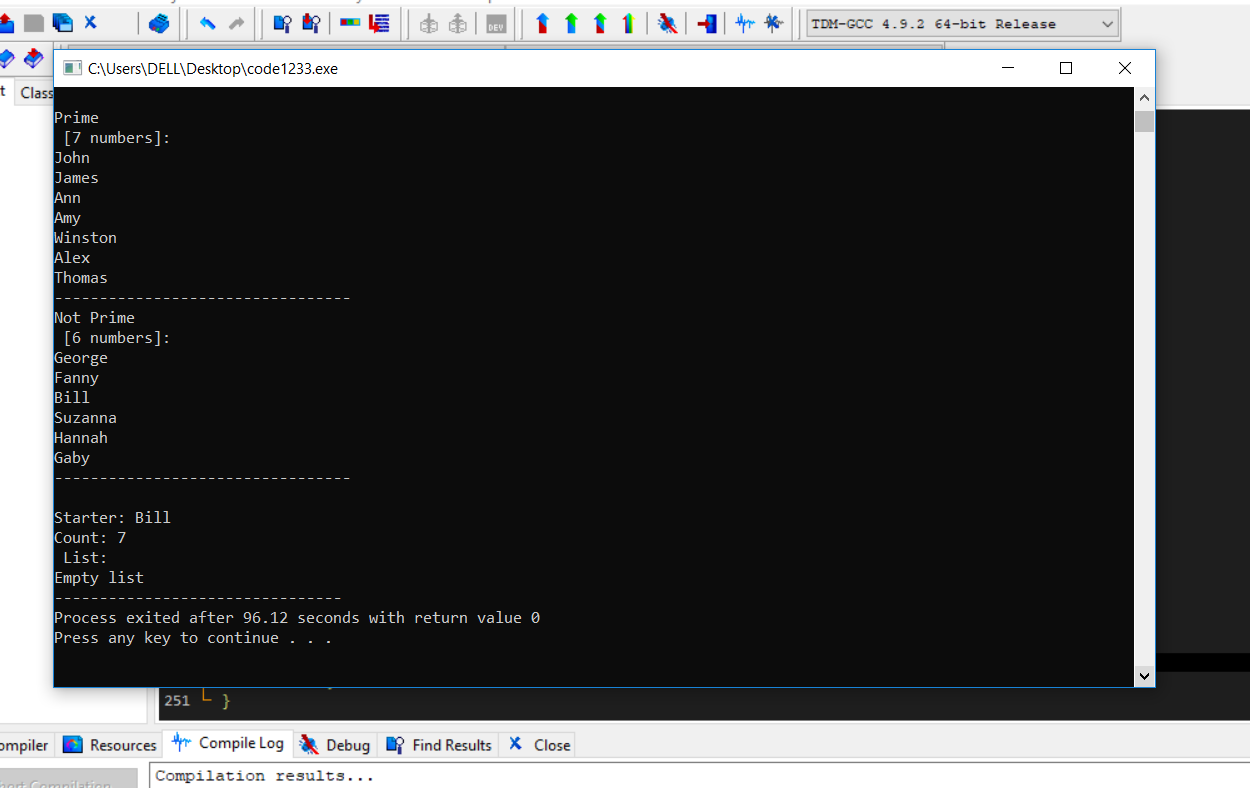
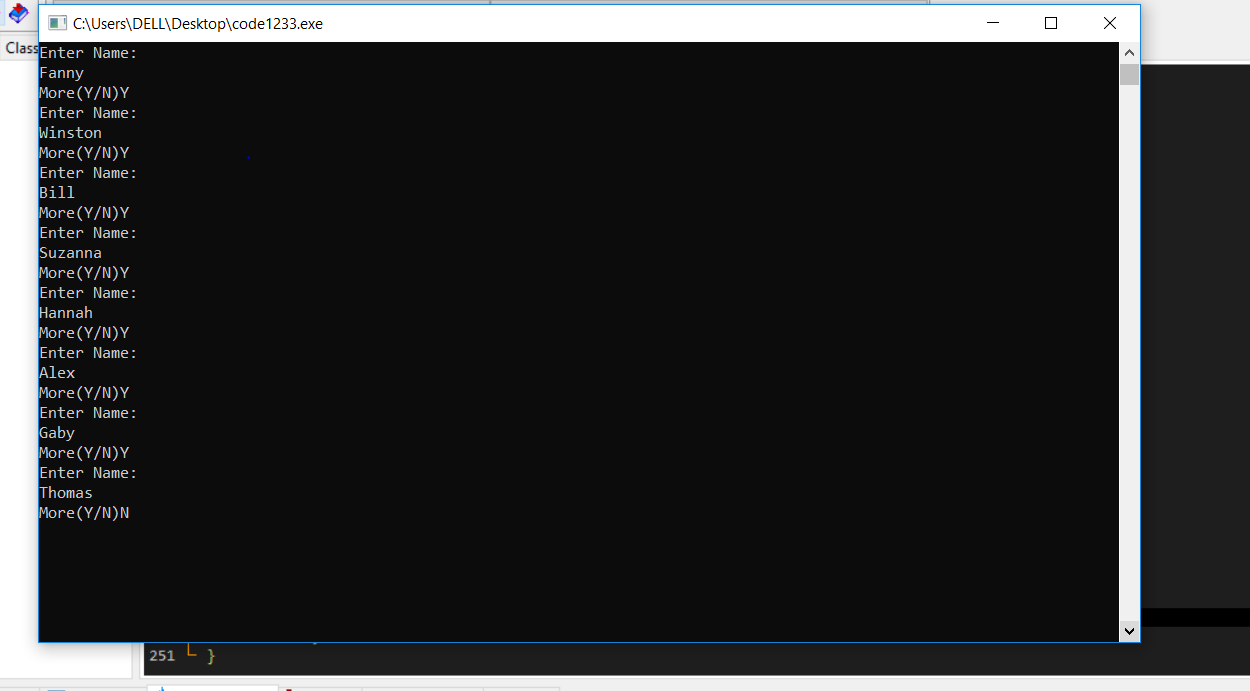
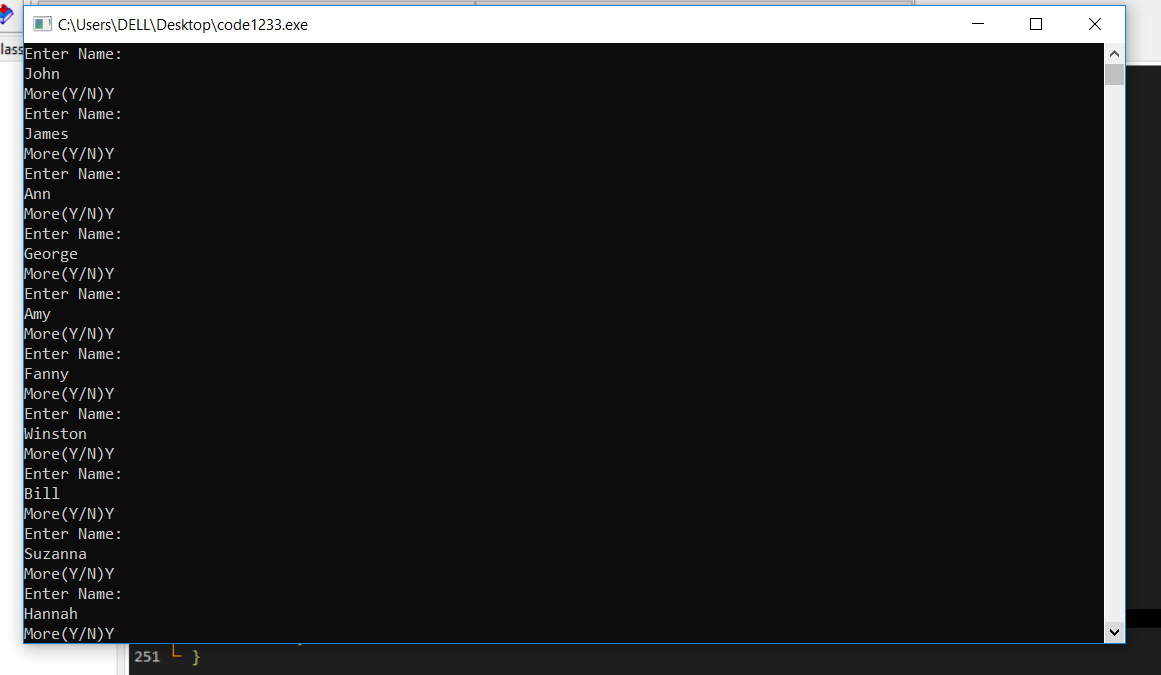
del();

cout<<"\n List: "<<endl;

display(head2);

return 0;

}



Q.22

#include<iostream>

using namespace std;

struct node

{

string course;

string name;

node \*right;

node \*down;

};

node \*head\_stud = NULL;

node \*head\_course = NULL;

void insert\_stud(string s)

{

node \*newnode = new node;

newnode->name = s;

if(head\_stud==NULL)

{

head\_stud = newnode;

newnode->right = head\_stud;

}

else

{

node \*temp = head\_stud;

while(temp->right!=head\_stud)

{

temp = temp->right;

}

temp->right = newnode;

newnode->right = head\_stud;

}

}

void insert\_course(string s)

{

node \*newnode = new node;

newnode->course = s;

if(head\_course==NULL)

{

head\_course = newnode;

newnode->down = head\_course;

}

else

{

node \*temp = head\_course;

while(temp->down!=head\_course)

{

temp = temp->down;

}

temp->down = newnode;

newnode->down = head\_course;

}

}

void display\_stud()

{

node \*temp = head\_stud;

int flag = 0;

while(flag!=1)

{

if(temp->right!=head\_stud)

{

cout<<temp->name<<endl;

temp = temp->right;

}

else

{

flag = 1;

}

}

cout<<temp->name<<endl;

}

void display\_course()

{

node \*temp = head\_course;

int flag = 0;

while(flag!=1)

{

if(temp->down!=head\_course)

{

cout<<temp->course<<endl;

temp = temp->down;

}

else

{

flag = 1;

}

}

cout<<temp->course<<endl;

}

int main()

{

int n,m;

string course\_code,stud\_name;

cout<<"Enter Number of students"<<endl;

cin>>n;

cout<<"Enter Number of courses"<<endl;

cin>>m;

for(int i=0;i<n;i++)

{

cout<<"Enter Student Name"<<endl;

cin>>stud\_name;

insert\_stud(stud\_name);

for(int j=0;j<m;j++)

{

cout<<"Enter Course Code"<<endl;

cin>>course\_code;

insert\_course(course\_code);

}

}

display\_stud();

cout<<endl;

display\_course();

return 0;

}

