--------------------------------------------------------------------Assignment Name: Perform Bubble Sort

Class: BCA-II Lab: BCA 405(DS)

--------------------------------------------------------------------

#include<iostream.h>

#include<conio.h>

class demo

{

int a[10],i,last,exch,j,n,temp;

public:

void get();

void asc\_sort();

void dec\_sort();

void disp();

};

void demo::get()

{

cout<<"\n Enter the array size:";

cin>>n;

cout<<"\nEnter the array element:";

for(i=1;i<=n;i++)

cin>>a[i];

}

void demo::asc\_sort()

{

last=n;

for(i=0;i<n-1;i++)

{

exch=0;

for(j=0;j<last-1;j++)

{

if(a[j]>a[j+1])

{

temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

}

exch=exch+1;

}

}

if(exch==0)

return;

else

last=last-1;

}

void demo::dec\_sort()

{

last=n;

for(i=0;i<n-1;i++)

{

exch=0;

for(j=0;j<last-1;j++)

{

if(a[j]<a[j+1])

{

temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

}

exch=exch+1;

}

}

if(exch==0)

return;

else

last=last-1;

}

void demo::disp()

{

cout<<"\nThe array element are";

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

cout<<a[i]<<"\t";

}

void main()

{

clrscr();

demo d;

d.get();

d.disp();

d.asc\_sort();

cout<<"\nAfter Ascending Sort:";

d.disp();

d.dec\_sort();

cout<<"\nAfter Descending Sort:";

d.disp();

getch();

}

\*/ Output \*/

Enter the array size: 3

Enter the array element: 12 3 45

The array element are12 3 45

After Ascending Sort:

The array element are3 12 45

After Descending Sort:

The array element are45 12 3

--------------------------------------------------------------------Assignment Name: Perform Selection Sort

Class: BCA-II Lab: BCA 405(DS)

--------------------------------------------------------------------

#include<iostream.h>

#include<conio.h>

class demo

{

int a[10],i, min\_index,j,n,temp;

public:

void get();

void asc\_sort();

void dsc\_sort();

void disp();

};

void demo::get()

{

cout<<"\nEnter the array size:";

cin>>n;

cout<<"\nEnter the array element:";

for(i=1;i<=n;i++)

cin>>a[i];

}

void demo::asc\_sort()

{

for(i=0;i<n-1;i++)

{

min\_index=i;

for(j=i+1;j<n;j++)

{

if(a[j]<a[min\_index])

min\_index=j;

}

if(min\_index!=i)

{

temp=a[min\_index];

a[min\_index]=a[i];

a[i]=temp;

}

}

}

void demo::dsc\_sort()

{

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

{

min\_index=i;

for(j=i+1;j<n;j++)

{

if(a[j]>a[min\_index])

min\_index=j;

}

if(min\_index!=i)

{

temp=a[min\_index];

a[min\_index]=a[i];

a[i]=temp;

}

}

}

void demo::disp()

{

cout<<"\n The array element are";

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

cout<<a[i]<<"\t";

}

void main()

{

clrscr();

demo d;

d.get();

d.disp();

d.asc\_sort();

cout<<"\nAfter ascending sort:";

d.disp();

d.dsc\_sort();

cout<<"\n After Descending sort:";

d.disp();

getch();

}

\*/ Output \*/

Enter the array size:4

Enter the array element:12 3 -45 -6

The array element are12 3 -45 -6

After ascending sort:

The array element are-45 -6 3 12

After Descending sort:

The array element are12 3 -6 -45

------------------------------------------------------------------------

Assignment Name: Implement Insertion Sort

Class: BCA-II Lab: BCA 405(DS)

------------------------------------------------------------------------

#include<iostream.h>

#include<conio.h>

#include<stdlib.h>

#include<math.h>

class insert

{

int n,a[10],temp,ptr,q,i,j,k,key;

public:

void get();

void sort();

void display();

};

void insert::get()

{

cout<<"\nEnter Range:";

cin>>n;

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

a[i]=random(1000);

cout<<"\nElements are :";

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

cout<<a[i]<<"\t";

}

void insert::sort()

{

a[0]=-9999;

for(i=1;i<n;i++)

{

temp=a[i];

ptr=i-1;

while(temp<a[ptr])

{

a[ptr+1]=a[ptr];

ptr--;

}

a[ptr+1]=temp;

}}

void insert::display()

{

cout<<"\nSorted Element using Insertion Sort:";

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

cout<<a[i]<<"\t";

}

void main()

{

clrscr();

insert h;

h.get();

h.sort();

h.display();

getch();

}

\*/ Output \*/

Enter Range:5

Elements are :10 3 335 33 355

Sorted Element using Insertion Sort:3 10 33 335 355

--------------------------------------------------------------------Assignment Name: Implement Linear and Binary Search

Class: MCA I Lab: BCA 405(DS)

--------------------------------------------------------------------

#include<iostream.h>

#include<conio.h>

#include<process.h>

class demo

{

int a[10],i,j,n,f,temp,ele,demo,mid,low,high;

public:

void get();

void sort();

void linear();

void binary();

void dis();

};

void demo::get()

{

cout<<"\n Enter n:";

cin>>n;

cout<<"\nEnter array Elements:";

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

cin>>a[i];

}

void demo::linear()

{

int ele;

cout<<"\nEnter the element to be search";

cin>>ele;

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

{

if(a[i]==ele)

{

cout<<"\nSuccessful search";

cout<<"\nElement is found at position "<<i;

return;

}

}

if(i>n)

{

cout<<"\nUnsuccessful search:";

cout<<"\nElement is not found ";

}

}

void demo::sort()

{

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

{

for(j=0;j<n-1;j++)

{

if(a[j]<a[j+1])

{

temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

}

}

}

}

void demo::binary()

{

cout<<"\nEnter element to be search ";

cin>>ele;

f=0;

low=0;

high=n-1;

while(low<=high)

{

mid=(low+high)/2;

if(a[mid]==ele)

{

f=1;

cout<<"\nElement is found at :"<<mid;

return;

}

else if(a[mid]<ele)

low=mid+1;

else if(a[mid]>ele)

high=mid-1;

}

if(f==0)

cout<<"\n Element is not found:";

}

void demo::dis()

{

cout<<"\n Element are \n";

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

cout<<a[i]<<"\t";

}

void main()

{

clrscr();

demo d;

int ch;

d.get();

d.dis();

cout<<"\n 1:Linear 2:Binary 3:exit\n";

while(ch!=3)

{

cout<<"\nEnter Choice:";

cin>>ch;

switch(ch)

{

case 1: d.linear(); break;

case 2: d.sort();

d.dis();

d.binary(); break;

case 3: exit(0); break;

}

}

getch();

}

\*/ Output \*/

Enter n:3

Enter array Elements:12 3 45

Element are

12 3 45

1:Linear 2:Binary 3:exit

Enter Choice:1

Enter the element to be search 3

Successful search

Element is found at position 2

Enter Choice:2

Element are

45 12 3

Enter element to be search 12

Element is found at :2

Enter Choice:2

Element are

45 12 3

Enter element to be search 56

Element is not found:

Enter Choice:3

--------------------------------------------------------------------Assignment Name: Implement Stack for Integer

Class: BCA-II Lab: BCA 405(DS)

--------------------------------------------------------------------

#include<iostream.h>

#include<conio.h>

#include<process.h>

class stack

{

int s[10],n,top,ele,i;

public:

stack()

{

top=-1;

}

void push();

void dis();

int pop();

int peep();

void change();

};

void stack::push()

{

if(top>=2)

cout<<"\nStack is overflow:";

else

{

cout<<"\nEnter element:";

cin>>ele;

top++;

s[top]=ele;

}

}

void stack::dis()

{

cout<<"\nElements in stack are:\n";

for(i=top;i>=0;i--)

cout<<s[i]<<"\t";

}

int stack::pop()

{

if(top==-1)

{

cout<<"\nUnderflow";

return 0;

}

else

return (s[top--]);

}

int stack::peep()

{

cout<<"\nEnter position:";

cin>>i;

if((top-i+1)<0)

{

cout<<"\nUnderflow";

return 0;

}

else

return (s[top-i+1]);

}

void stack::change()

{

cout<<"\nEnter position ";

cin>>i;

if((top-i+1)<0)

{

cout<<"\nUnderflow";

}

else

{

int n;

cout<<"\nEnter element:";

cin>>n;

s[top-i+1]=n;

}

}

void main()

{

clrscr();

stack s;

int ch;

cout<<"\n1. Push 2.Display 3.Pop 4.Peep 5.Change 6.Exit\n";

while(ch!=6)

{

cout<<"\nEnter ch :";

cin>>ch;

switch(ch)

{

case 1: s.push(); break;

case 2: s.dis(); break;

case 3: int n=s.pop();

if(n>0)

cout<<"\nPop ele is "<<n;

break;

case 4: int m=s.peep();

if(m>0)

cout<<"\nPeep ele is "<<m;

break;

case 5: s.change(); break;

case 6: exit(0);

}

}

getch();

}

\*/ Output \*/

1. Push 2.Display 3.Pop 4.Peep 5.Change 6.Exit

Enter ch :1

Enter element:10

Enter ch :1

Enter element:20

Enter ch :1

Enter element:30

Enter ch :1

Stack is overflow:

Enter ch :2

Elements in stack are:

30 20 10

Enter ch :3

Pop ele is 30

Enter ch :2

Elements in stack are:

20 10

Enter ch :4

Enter position:1

Peep ele is 20

Enter ch :

2

Elements in stack are:

20 10

Enter ch :5

Enter position 1

Enter element:80

Enter ch :2

Elements in stack are:

80 10

Enter ch : 6

--------------------------------------------------------------------Assignment Name: Implement linear queue for integer

Class: BCA-II Lab: BCA 405(DS)

--------------------------------------------------------------------

#include<iostream.h>

#include<conio.h>

#include<process.h>

class queue

{

int f,r,q[10],n,i;

public:

queue()

{

f=r=-1;

}

void insert();

void del();

void dis();

};

void queue::insert()

{

if(r==3)

cout<<"\nOverflow";

else

{

cout<<"\nEnter n";

cin>>n;

if(f==-1)

f=1;

r++;

q[r]=n;

}

}

void queue::del()

{

if(f==0)

{

cout<<"\nUnderflow";

return;

}

else

{

int n;

n=q[f];

if(f==r)

f=r=0;

else

f++;

cout<<"\nDeleted element is "<<n;

}

}

void queue::dis()

{

if(f==0)

cout<<"\nUnderflow";

else

{

cout<<"\nElements in queue are:";

for(i=f;i<=r;i++)

cout<<q[i]<<"\t";

}

}

void main()

{

clrscr();

queue q;

int ch;

cout<<"\n 1.insert 2.display 3.delete 4. exit \n";

while(ch!=4)

{

cout<<"\nEnter ch:";

cin>>ch;

switch(ch)

{

case 1: q.insert(); break;

case 2: q.dis(); break;

case 3: q.del(); break;

case 4:exit(0);

}

}

getch();

}

\*/ Output \*/

1.insert 2.display 3.delete 4. exit

Enter ch:3

Underflow

Enter ch:1

Enter n10

Enter ch:1

Enter n20

Enter ch:1

Enter n30

Enter ch:1

Overflow

Enter ch:2

Elements in queue are:10 20 30

Enter ch:3

Deleted element is 10

Enter ch:2

Elements in queue are:20 30

Enter ch:4

--------------------------------------------------------------------Assignment Name: Write a program to implement singly linked list with operations. i)create ii) insert iii) delete

Class: BCA-II Lab: BCA 405(DS)

--------------------------------------------------------------------

#include<iostream.h>

#include<conio.h>

#include<process.h>

class ll

{

int info,ele;

ll \*node,\*start,\*link,\*move;

public:

ll()

{

start=NULL;

}

void ins\_beg();

void del\_beg();

void disp();

};

void ll::del\_beg()

{

if(start==NULL)

{

cout<<"\n link list is empty" ;

}

else

{

move=start;

start=move->link;

move->link=NULL;

}

}

void ll::ins\_beg()

{

ll \*node1=new ll;

cout<<"\n enter element to insert==>> ";

cin>>ele;

node1->info=ele;

node1->link=NULL;

if(start==NULL)

{

start=node1;

}

else

{

move=start;

start=node1;

node1->link=move;

}

}

void ll::disp()

{

if(start==NULL)

{

cout<<"\n Link list is empty";

}

else

{

move=start;

while(move!=NULL)

{

cout<<" "<<move->info;

move=move->link;

}

}

}

int main()

{

ll l1;

int ch;

clrscr();

cout<<"\n enter 1 for insert 2 for disp 3 for del 4 for exit ";

while(ch!=4)

{

cout<<"\n enter choice";

cin>>ch;

switch(ch)

{

case 1:l1.ins\_beg();break;

case 2:l1.disp();break;

case 3:l1.del\_beg();break;

case 4:exit(0);

}

}

clrscr();

getch();

return(0);

}

/\*

enter 1 for insert 2 for disp 3 for del 4 for exit

enter choice1

enter element to insert==>> 10

enter choice3

enter choice1

enter element to insert==>> 10

enter choice1

enter element to insert==>> 20

enter choice2

20 10

enter choice3

enter choice2

10

enter choice4

\*/