ASSINGMENT 1:

Roll no : 17f-8195.

Name : Muhammad usama.

Section :B

Task 1::static working:

#include<iostream>

usingnamespace std;

void chack\_static(int&x);

int main()

{

int x=0;

chack\_static(x);

chack\_static(x);

chack\_static(x);

cout<<"value of x : "<<x<<endl;

system("pause");

return 0;

}

void chack\_static(int&x)

{

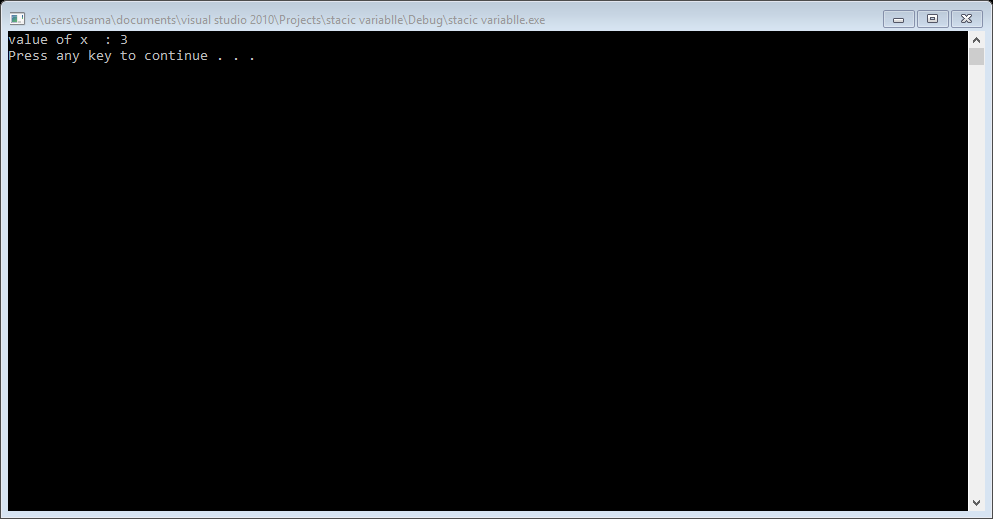
staticint i=0; //static variable initialize only once.

i++;

x=i;

}

Result:



Task: 1::static workig::

#include<iostream>

usingnamespace std;

staticint x=0; //we do same process to check with x and y.

int y=0; //they behave same. because we declare both in global.

void fun(); /\*if we use static in file means its scope only in

this file. and cannot be accessable in any other file \*/

int main()

{

x++;

y++;

cout<<"in main x : "<<x<<endl;

cout<<"in main y : "<<y<<endl;

fun();

system("pause");

return 0;

}

void fun()

{

x++; /\* they show same behaviour if we declare both in global.

woring of static variable is different. if we declare in

function. as we do in above programe.\*/

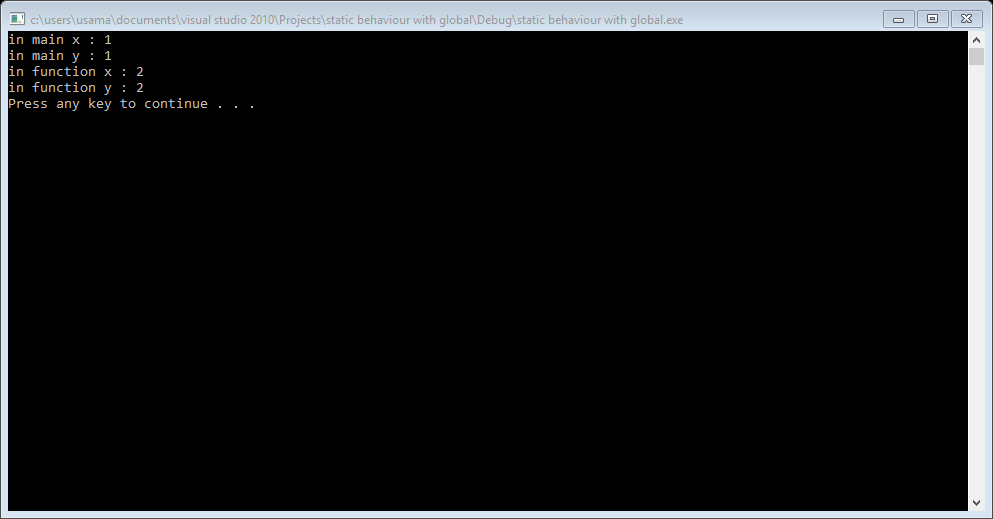
y++;

cout<<"in function x : "<<x<<endl;

cout<<"in function y : "<<y<<endl;

}

Result:



Task 2::auto working.

#include<iostream>

usingnamespace std;

int main()

{

auto a=5; /\*this auto automaticaly make it integer type variable.

and auto scope is local.\*/

auto x=5.322; /\*this auto make automaticaly a double or float type variable.

and auto scope is local.\*/

auto y=x\*3; /\*auto made y auto as tpe of variable that tpe of value store in y.

and auto scope is local.\*/

cout<<"answer of y is : "<<y<<endl;

auto ch='a'; /\*auto made ch charater variable automaticaly.

and auto scope is local.\*/

auto str="my name is usama"; /\*auto made str string type variable.

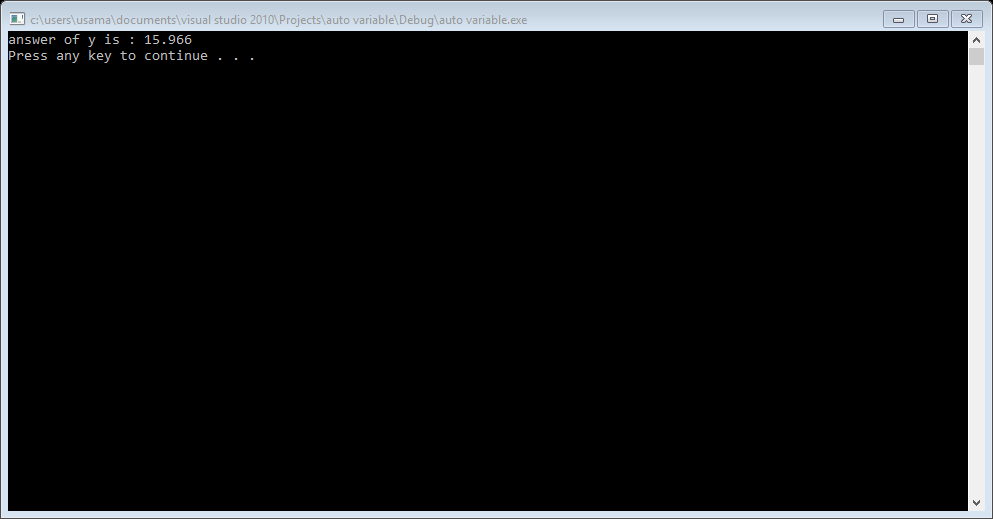
and auto scope is local.\*/

system("pause");

return 0;

}

Result:



Task 3::extern working::

#include<iostream>

usingnamespace std;

void disp();

int main()

{

externint x;

cout<<x<<endl;

disp();

system("pause");

return 0;

}

int x=10; /\*it must declare any where in the programe.

if extern declare in main even you declare it in the

function.if you not declare it error occure undeclare x.

and extern scope is global. It can be shared amoung multiply files \*/

void disp()

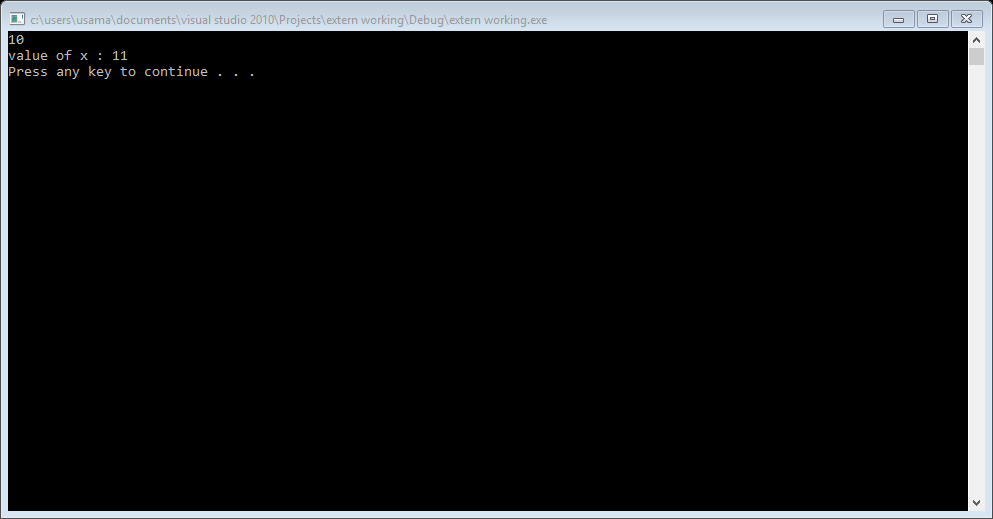
{

x++;

cout<<"value of x : "<<x<<endl;

}

Result:



Task 4::register ::

#include<iostream>

usingnamespace std;

int main()

{

registerint x=10; //this register show that memory allocate to x in registers ma

be. register is use to define local variable

x=12;

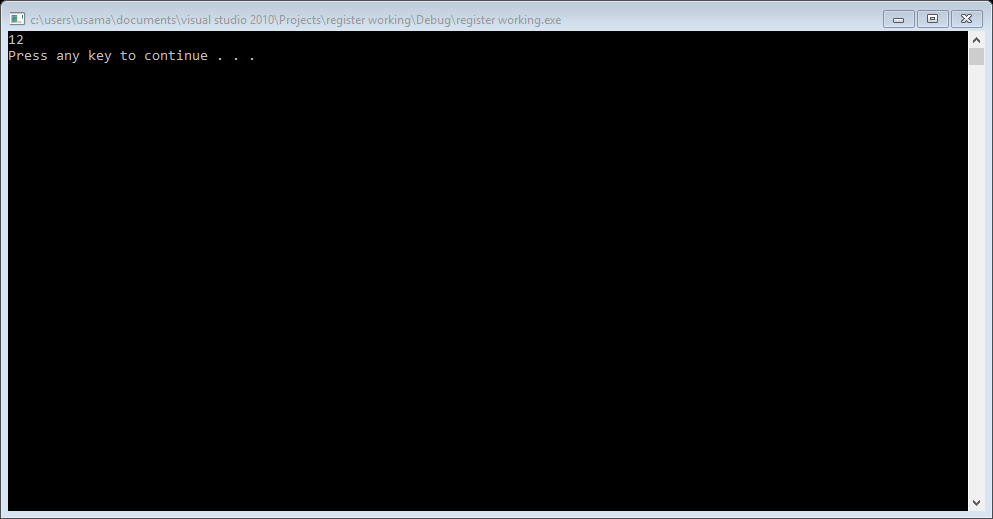
cout<<x<<endl;

system("pause");

return 0;

}

Result:



Task 5:: linear search::

#include<iostream>

usingnamespace std;

void linear\_search(int list[],int size,int key);

int main()

{

int list[]={0};

int size=0,key=0;

cout<<"enetr the size of list : ";

cin>>size;

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

{

cout<<"enter "<<i+1<<" element : ";

cin>>list[i];

}

cout<<"enter the key that you want to search : ";

cin>>key;

linear\_search(list,size,key);

system("pause");

return 0;

}

void linear\_search(int list[],int size,int key)

{

int \*adress=0;

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

{

if(list[i]==key)

{

cout<<"your number is found : "<<list[i]<<endl;

}

adress=&list[i];

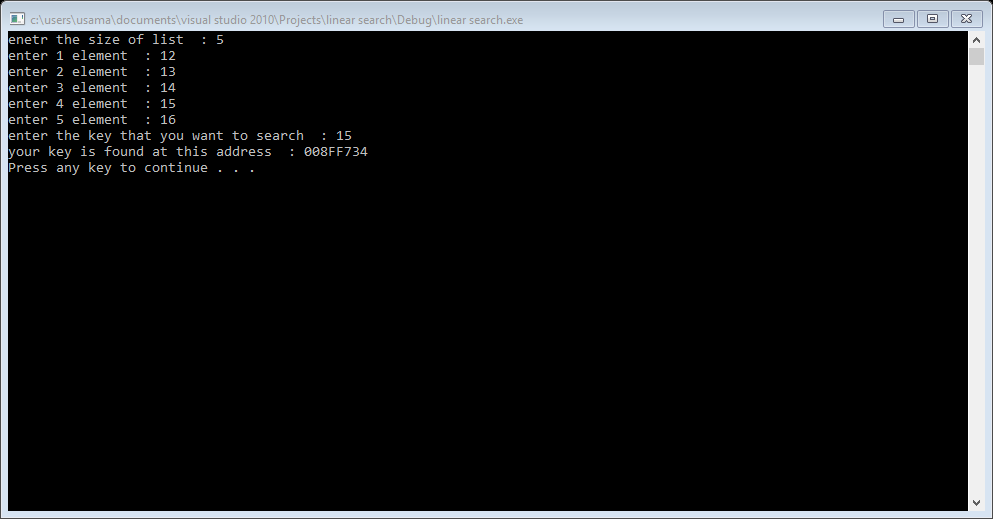
cout<<"your key is found at this address : "<<adress<<endl;

break;

}

}

Result:



Task 6::

#include<iostream>

#include<time.h>

#include<fstream>

#include<iomanip>

#include<time.h>

usingnamespace std;

void writing\_file();

void read\_file();

void bubble\_sort();

void selection\_sort();

void writing\_section\_sort\_file();

void writing\_bubble\_sort\_file();

void count\_random\_file();

void binary\_search();

void linear\_search();

ifstream read;

ofstream write;

int readf[1000]={0};

int count[100000]={0};

int key=0;

int fkey=0;

int stime=0,btime=0,bsearch\_time=0,linear\_time=0;

char selection=0;

int main()

{

writing\_file();

read\_file();

count\_random\_file();

cout<<"what type of sorting you want \npress b for bubble sort \npress s for selection sort\npress A for both type of sorting"<<endl;

cin>>selection;

if(selection=='s'||selection=='S')

{

selection\_sort();

writing\_section\_sort\_file();

}

elseif(selection=='b'||selection=='B')

{

bubble\_sort();

writing\_bubble\_sort\_file();

}

elseif(selection=='A'||selection=='a')

{

selection\_sort();

writing\_section\_sort\_file();

bubble\_sort();

writing\_bubble\_sort\_file();

}

else

{

cout<<"you press invalid key"<<endl;

}

cout<<"press b for binary search \npress l linear search"<<endl;

cin>>selection;

if(selection=='b'||selection=='B')

{

binary\_search();

}

elseif(selection=='l'||selection=='L')

{

linear\_search();

}

else

{

cout<<"invalid option"<<endl;

}

if(fkey==key)

{

cout<<"your no seached that is : "<<fkey<<endl;

}

else

{

cout<<"your key not found"<<endl;

}

system("pause");

return 0;

}

void writing\_file()

{

srand(time(0));

write.open("random\_data.txt");

if(write.is\_open())

{

for(int i=0; i<1000; i++)

{

write<<setw(7)<<rand()%1000;

}

}

else

{

cout<<"file not open"<<endl;

}

write.close();

}

void read\_file()

{

read.open("random\_data.txt");

if(read.is\_open())

{

for(int i=0; i<1000; i++)

{

read>>readf[i];

}

}

else

{

cout<<"file not open "<<endl;

}

read.close();

}

void selection\_sort()

{

int i=0,smalli=0,temp=0;

for(i=0; i<1000; i++)

{

smalli=i;

for(int j=i; j<1000; j++)

{

if(readf[j]<readf[smalli])

{

smalli=j;

}

}

temp=readf[smalli];

readf[smalli]=readf[i];

readf[i]=temp;

}

stime=time(0);

cout<<"the time required for selection sort : "<<stime<<" ms"<<endl;

}

void bubble\_sort()

{

int temp=0;

int size=1000;

for(int iter=1; iter<size; iter++)

{

for(int i=0; i<size-iter; i++)

{

if(readf[i]>readf[i+1])

{

temp=readf[i+1];

readf[i+1]=readf[i];

readf[i]=temp;

}

}

}

btime=time(0);

cout<<"time taken by bubble sort : "<<btime<<endl;

}

void writing\_section\_sort\_file()

{

write.open("selection\_sort.txt");

if(write.is\_open())

{

write<<"this sorting is done by selection sort"<<endl;

for(int i=0; i<1000; i++)

{

write<<setw(7)<<readf[i];

}

}

else

{

cout<<"file not open"<<endl;

}

write.close();

}

void writing\_bubble\_sort\_file()

{

write.open("bubble\_sort.txt");

if(write.is\_open())

{

write<<"this sorting is done bubble sort"<<endl;

for(int i=0; i<1000; i++)

{

write<<setw(7)<<readf[i];

}

}

else

{

cout<<"file not open"<<endl;

}

write.close();

}

void count\_random\_file()

{

read.open("random\_data.txt");

if(read.is\_open())

{

int i=0;

while(!read.eof())

{

read>>count[i];

i++;

}

cout<<"total integer enter in random file : "<<i<<endl;

}

else

{

cout<<"file not open "<<endl;

}

read.close();

}

void linear\_search()

{

cout<<"this searching is going to done by linear search"<<endl;

cout<<"enter key that you seach"<<endl;

cin>>key;

for(int i=0; i<1000; i++)

{

if(key==readf[i])

{

fkey=key;

break;

}

}

linear\_time=time(0);

cout<<"time taken by linear search : "<<linear\_time<<" ms"<<endl;

}

void binary\_search()

{

cout<<"this searching is going to done by binary search method"<<endl;

int size=1000;

int fist=0,last=0,mid=0;

last=size-1;

cout<<"eneter key that yoy want to seach"<<endl;

cin>>key;

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

{

mid=(fist+last)/2;

if(readf[mid]==key)

{

fkey=key;

}

elseif(readf[mid]<key)

{

fist=mid+1;

}

elseif(readf[mid]>key)

{

last=mid-1;

}

else

{

cout<<"invalid key"<<endl;

}

}

bsearch\_time=time(0);

cout<<"time taken by binary search : "<<bsearch\_time<<" ms"<<endl;

}

Result::

