Dynamic memory allocation problems:

1)primitive datatypes:

#include<iostream>

using namespace std;

int main(){

int \*p=new int;

cout<<"enter the values";

cin>>\*p;

cout<<"the values are:";

cout<<\*p<<endl;

delete []p;

{

float \*p=new float;

cout<<"enter the float";

cin>>\*p;

cout<<"the values are:";

cout<<\*p<<endl;

delete []p;

}

{

char \*p=new char;

cout<<"enter the char";

cin>>\*p;

cout<<"the values are:";

cout<<\*p<<endl;

delete []p;

}

{

double \*p=new double;

cout<<"enter the values";

cin>>\*p;

cout<<"the values are:";

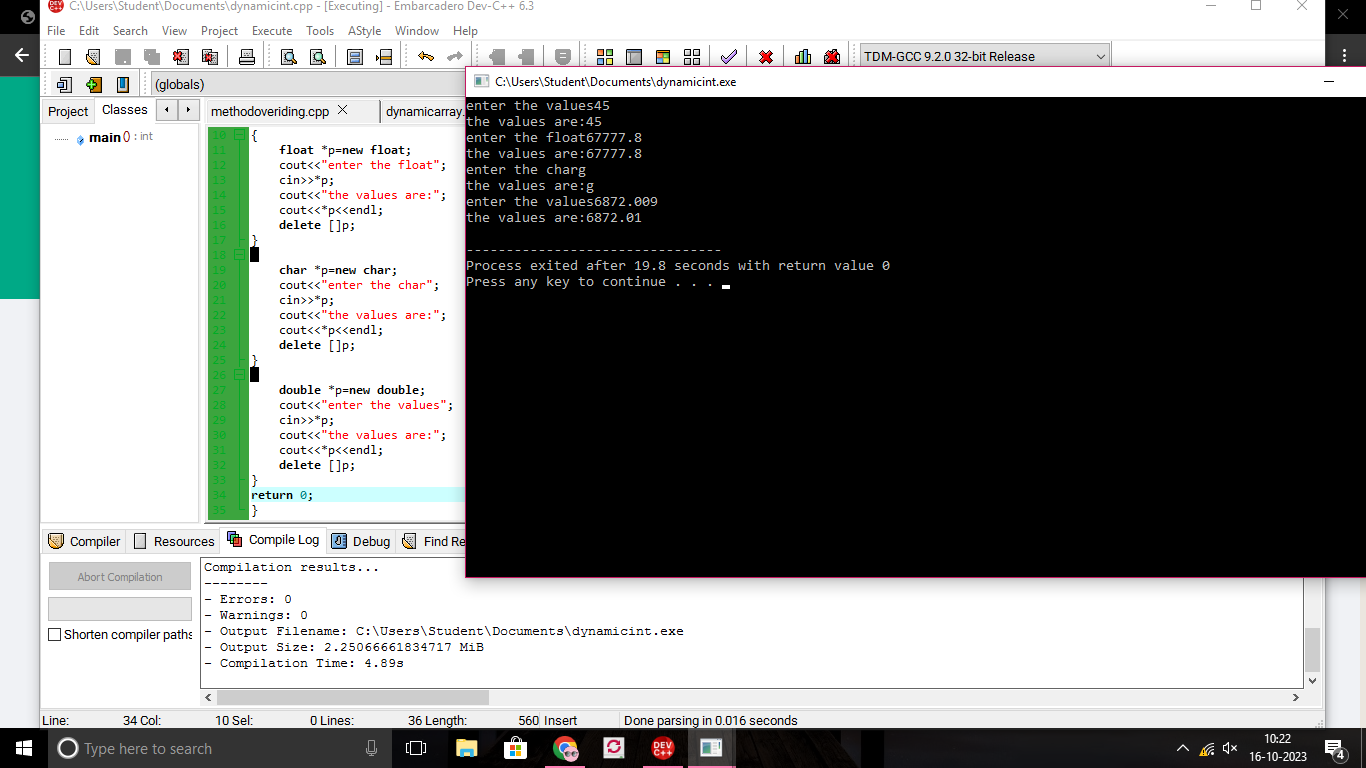
cout<<\*p<<endl;

delete []p;

}

return 0;

}



2) dynamic memory allocation for user defined datatype like array:

Array integer:

#include<iostream>

using namespace std;

int main(){

int n,i;

cout<<"enter the num";

cin>>n;

int \*p=new int[n];

cout<<"enter the values\n";

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

{

cin>>\*(p+i);

}

cout<<"the values are";

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

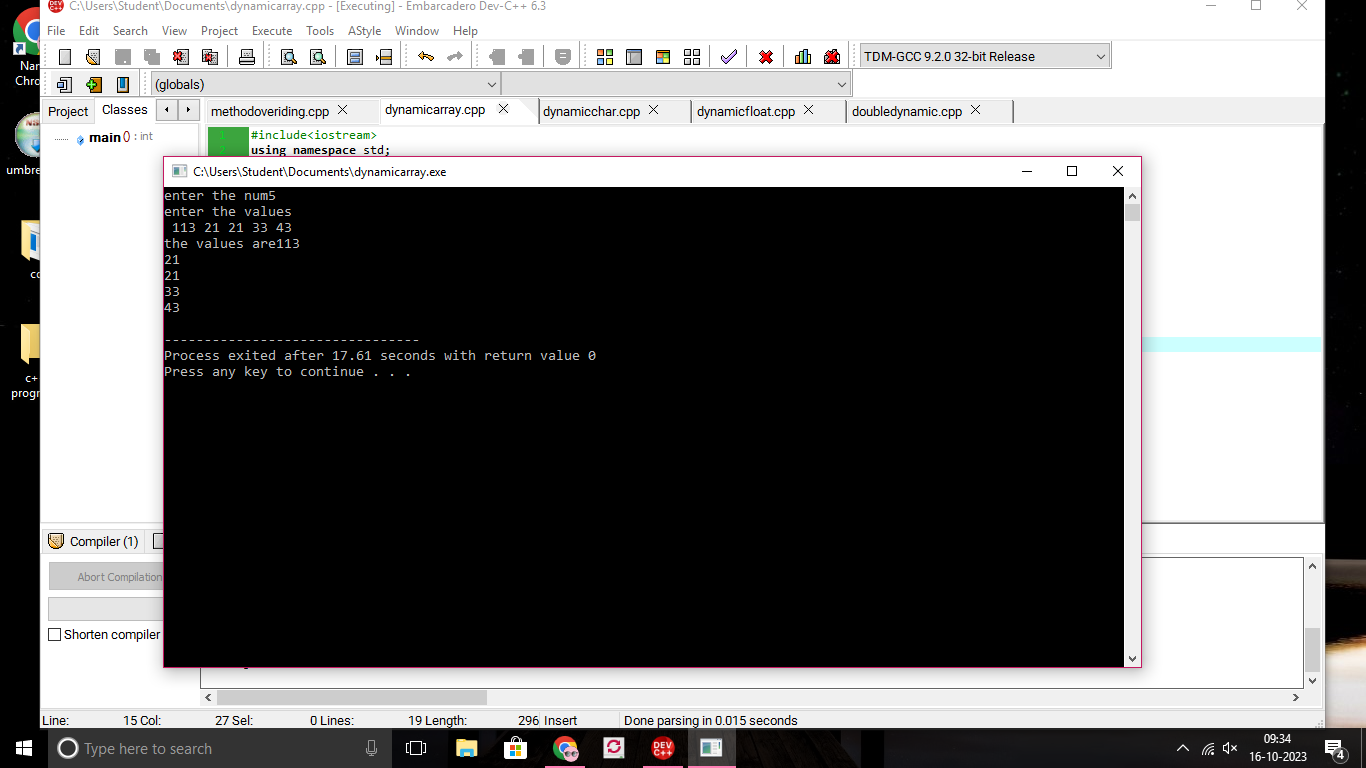
cout<<\*(p+i)<<endl;

}

delete []p;

return 0;

}



3)array character:

#include<iostream>

using namespace std;

int main(){

int n;

int i;

cout<<"enter the num";

cin>>n;

char \*p=new char[n];

cout<<"enter the values\n";

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

{

cin>>\*(p+i);

}

cout<<"the values are";

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

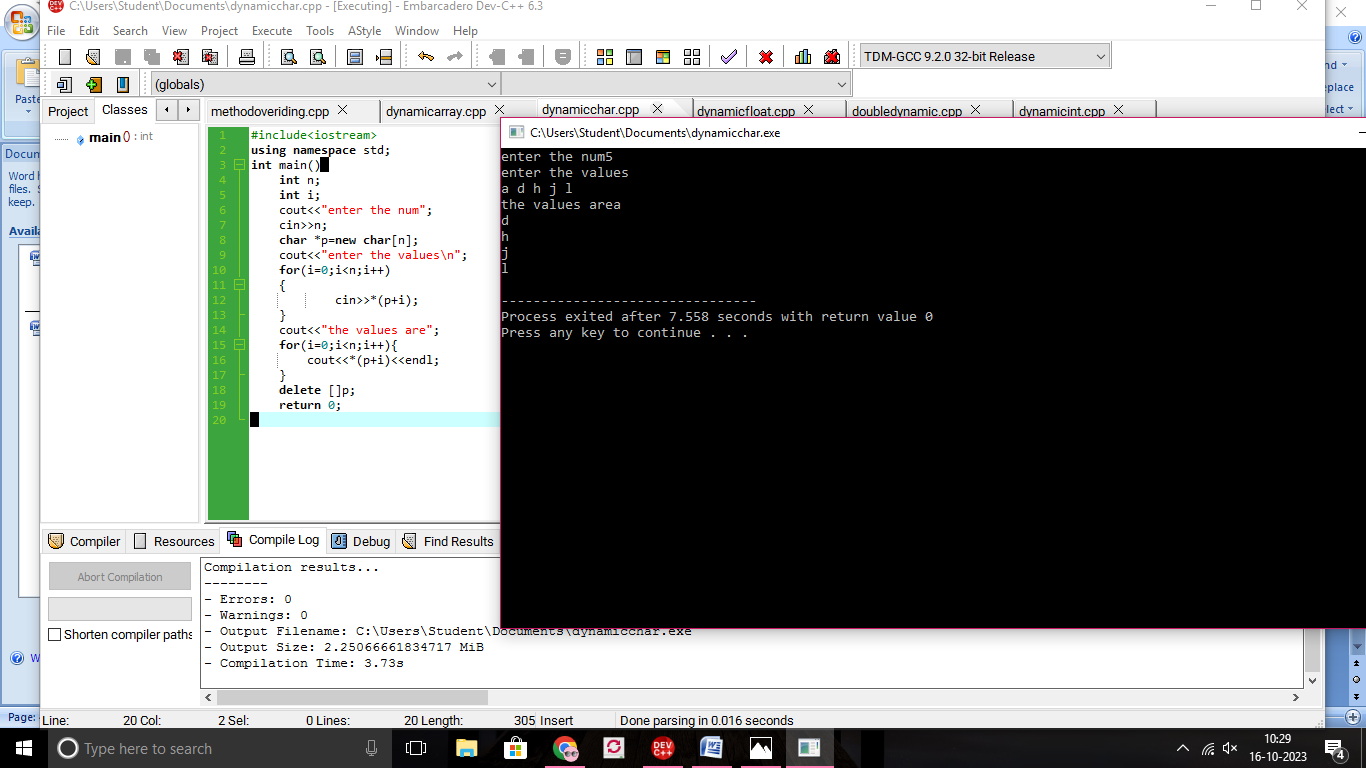
cout<<\*(p+i)<<endl;

}

delete []p;

return 0;

}



4)array float:

#include<iostream>

using namespace std;

int main(){

int n,i;

cout<<"enter num of array:";

cin>>n;

float \*p=new float[n];

cout<<"ener the values";

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

{

cin>>\*(p+i);

}

cout<<"the values are:";

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

{

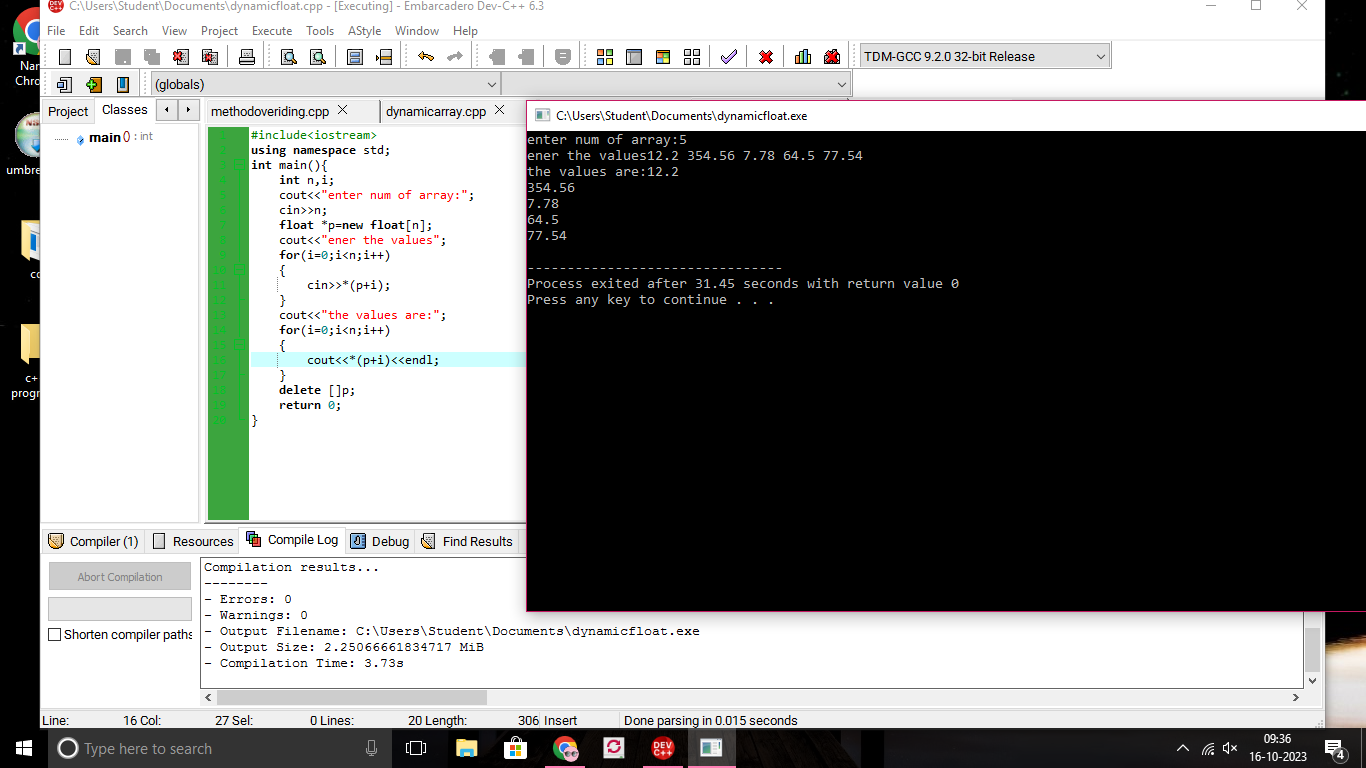
cout<<\*(p+i)<<endl;

}

delete []p;

return 0;

}



5)array double:

#include<iostream>

using namespace std;

int main(){

int n,i;

cout<<"enter num of array:";

cin>>n;

double \*p=new double[n];

cout<<"ener the values";

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

{

cin>>\*(p+i);

}

cout<<"the values are:";

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

{

cout<<\*(p+i)<<endl;

}

delete []p;

return 0;

}

