Условие:

Создайте перегруженную функцию maxc(), которая находит максимум для двух переменных типа целые (int), вещественные (double), символьные (char).

* С помощью перегруженных функций (без классов)
* С помощью перегруженных конструкторов или методов класса

Исходный код программы:

#include **<iostream>  
using namespace** std;  
**int** maxc(**int** a, **int** b)  
{  
 **if** (a > b)  
 **return** a;  
 **else  
 return** b;  
}  
**double** maxc(**double** a, **double** b)  
{  
 **if** (a > b)  
 **return** a;  
 **else  
 return** b;  
}  
**char** maxc(**char** a, **char** b)  
{  
 **if** (a > b)  
 **return** a;  
 **else  
 return** b;  
}  
**class** Max {  
 **char** cmax;  
 **int** imax;  
 **double** dmax;  
  
**public**:  
 Max(**int** a, **int** b)  
 {  
 **if** (a > b)  
 imax = a;  
 **else** imax = b;  
 }  
 Max(**char** a, **char** b)  
 {  
 **if** (a > b)  
 cmax = a;  
 **else** cmax = b;  
 }  
 Max(**double** a, **double** b)  
 {  
 **if** (a > b)  
 dmax = a;  
 **else** dmax = b;  
 }  
 **int** GetIntMax() { **return** imax; }  
 **double** GetDoubleMax() { **return** dmax; }  
 **char** GetCharMax() { **return** cmax; }  
};  
  
**int** main()  
{  
 cout << **"a."** << **"\n"**;  
 cout << **"maxc(1,2)="** << maxc(1, 2) << **"\n"**;  
 cout << **"maxc(1.0,2.0)="** << maxc(1.0, 2.0) << **"\n"**;  
 cout << **"maxc('a','b')="** << maxc(**'a'**, **'b'**) << **"\n"**;  
 cout << **"b."** << **"\n"**;  
 cout << **"Max(1,2).GetIntMax()="** << Max(1, 2).GetIntMax() << **"\n"**;  
 cout << **"Max(1.0,2.0).GetDoubleMax()="** << Max(1.0, 2.0).GetDoubleMax() << **"\n"**;  
 cout << **"Max('a','b').GetCharMax()="** << Max(**'a'**, **'b'**).GetCharMax() << **"\n"**;  
}

Скриншоты программы:

