Currency Class

Currency Name, Fractional Name, Whole Parts, Fractional Parts

Get and Sets

Overload add and subtract (only on same type of currency) (fractional rolls into whole)

Polymorphic construction, initialization lists

Wallet Class

# of currency types

If currency type is in wallet

Add/Remove money by currency

Check if currency type is in wallet

Add/remove to/from that currency type

Remove all currency

Check if wallet is empty

Main

User adds a maximum of 5 different currency types entering the whole and fractional names

(Enums for the currency types to be generated?)

Able to add or subtract as many times as they want (if able)

Output total value and names of currencies in the wallet

Able to empty the wallet and print relevant information about the operation

Emphasis on user interactivity

Initialization Lists:

Call the parents constructor (that requires parameters) in the child’s constructor

Class Foo {

Public:

Foo(int x)

{

Std::cout << x << std::endl;

}

Class Bar : public Foo {

Public:

Bar(int x, int y) : Foo (this->x)

{

Std::cout << y << std::endl;

}

Result:

X

y