

# **Correction TD 6 : Heritage Multiple Compte Bancaire**

```
#include<iostream.h>
```

```
/*
class
{
public:
();
virtual ~();
};

::() {cout << "    ";
::~() {cout << "    ";
*/
```

## **class Account**

```
{
public:
float Balance;
public:
Account();
virtual ~Account();
virtual void deposit(float);
};

Account::Account(){cout << " Account " << endl; Balance=0;}
Account::~Account(){ cout << " Balancing account, pay "
                  << Balance << " to the client" << endl; }
void Account::deposit(float f){Balance+=f;}
```

## **class Deposit\_Account : virtual public Account**

```
{

public:

Deposit_Account();
virtual ~Deposit_Account();
virtual void deposit(float);
};

Deposit_Account::Deposit_Account(){cout << " Deposit_Account " << endl;}
Deposit_Account::~Deposit_Account(){Balance-=100;}
void Deposit_Account::deposit(float f){Balance+=f-1;}
```

## **class Deposit\_Account\_w\_Creditcard : virtual public Deposit\_Account**

```
{
public:
Deposit_Account_w_Creditcard();
virtual ~Deposit_Account_w_Creditcard();
};
```

```
Deposit_Account_w_Creditcard::Deposit_Account_w_Creditcard()
{cout << " Deposit_Account_w_Creditcard " << endl;}
Deposit_Account_w_Creditcard::~Deposit_Account_w_Creditcard()
{}
```

```

class Remunerative_Account : virtual public
Account
{
public:
virtual void deposit(float);
Remunerative_Account();
virtual ~Remunerative_Account();
};

Remunerative_Account::Remunerative_Account()
    {cout << " Remunerative_Account "<< endl;}
Remunerative_Account::~Remunerative_Account(){Balance*=1.1;}
void Remunerative_Account::deposit(float f)
{Balance+=f; Balance*=1.01;}


class Remunerative_Deposit_Account
    : virtual public Remunerative_Account, virtual public Deposit_Account
{

public:
virtual void deposit(float);
Remunerative_Deposit_Account();
virtual ~Remunerative_Deposit_Account();
};

Remunerative_Deposit_Account::Remunerative_Deposit_Account()
    {cout << " Remunerative_Deposit_Account "<< endl;}
Remunerative_Deposit_Account::~Remunerative_Deposit_Account()
    {}
void Remunerative_Deposit_Account::deposit(float f){Balance+=f+1;}


class
Remunerative_Deposit_Account_w_Creditcard
    : virtual public Deposit_Account_w_Creditcard,
      virtual public Remunerative_Deposit_Account
//    : virtual public Remunerative_Deposit_Account,
//    virtual public Deposit_Account_w_Creditcard
    {

public:
Remunerative_Deposit_Account_w_Creditcard();
virtual ~Remunerative_Deposit_Account_w_Creditcard();
};

Remunerative_Deposit_Account_w_Creditcard::Remunerative_Deposit_Account_w_Creditcard
    ()
    {cout << " Remunerative_Deposit_Account_w_Creditcard "<< endl;}
Remunerative_Deposit_Account_w_Creditcard::~Remunerative_Deposit_Account_w_Creditcard
    () {}

```

## main()

```
{
    /* cout <<"----- opening of a Deposit account-----" <<endl;
    Deposit_Account *RA
        = new Deposit_Account;
    RA->deposit(400);
    cout <<"----- closing of a Deposit account-----" <<endl;
    RA->end();
    delete RA;
    */
    cout <<"----- opening of a remunerative deposit account-----" <<endl;
    Remunerative_Deposit_Account *RDA
        = new Remunerative_Deposit_Account();
//    RDA->Deposit_Account::deposit(800); //cas heritage non virtuel
    RDA->deposit(800);

    cout << " RDA balance " << RDA->Deposit_Account::Balance << endl;
    cout <<"----- closing of a remunerative deposit account-----" <<endl;
    delete RDA;

    cout <<"----- opening of a remunerative deposit account with CC-----"
<<endl;
    Remunerative_Deposit_Account_w_Creditcard *RDACC
        = new Remunerative_Deposit_Account_w_Creditcard();
    RDACC->deposit(800);
    cout <<"----- closing of a remunerative deposit account with CC-----"
<<endl;
    delete RDACC;
}

/* Heritage non virtuel, cas RDA seulement
-----
----- opening of a remunerative deposit account-----
Account
Remunerative_Account
Account
Deposit_Account
Remunerative_Deposit_Account
RDA balance 800
----- closing of a remunerative deposit account-----
Balancing account, pay 700 to the client
Balancing account, pay 0 to the client
*/

```

## /\*Heritage virtuel, cas complet

-----

```
----- opening of a remunerative deposit account-----
Account
Remunerative_Account
Deposit_Account
Remunerative_Deposit_Account
RDA balance 801
----- closing of a remunerative deposit account-----
Balancing account, pay 771.1 to the client
----- opening of a remunerative deposit account with CC-----
```

```
Account
Deposit_Account
Deposit_Account_w_Creditcard
Remunerative_Account
Remunerative_Deposit_Account
Remunerative_Deposit_Account_w_Creditcard
----- closing of a remunerative deposit account with CC-----
Balancing account, pay 781.1 to the client
```

```
*/  
/*
```

## **le même avec ordre d'héritage inverse pour RDACC**

```
----- opening of a remunerative deposit account-----
Account
Remunerative_Account
Deposit_Account
Remunerative_Deposit_Account
RDA balance 801
----- closing of a remunerative deposit account-----
Balancing account, pay 771.1 to the client
----- opening of a remunerative deposit account with CC-----
Account
Remunerative_Account
Deposit_Account
Remunerative_Deposit_Account
Deposit_Account_w_Creditcard
Remunerative_Deposit_Account_w_Creditcard
----- closing of a remunerative deposit account with CC-----
Balancing account, pay 771.1 to the client
```

```
*/
```