

TP 2

Consignes :

Suivre les conventions suivantes pour nommer les différents éléments de votre code :

- **Classes** : le nom d'une classe commence par une majuscule, ainsi que toutes les autres parties du nom (par ex. : `CompteEtudiant`);
- **Attributs et variables** : le nom d'un attribut d'une classe, ou d'une variable locale, commence par une minuscule, alors que les autres parties du nom commencent par une majuscule (par ex. : `firstName`);
- **Fonctions et méthodes** : le nom d'une fonction suit la même règle que celle de la classe (par ex. : `GetFirstName()`).
- **Constantes** : le nom d'une constante est tout en majuscule avec des tirets bas séparant les différentes parties du nom (par ex. : `NOMBRE_DE_TRUCS`).
- **Utiliser Doxygen** et commenter vos fichiers pour créer une documentation HTML de votre code. Consulter <http://www.doxygen.org>. Voir la courte introduction sur <http://lugatgt.org/2002/05/30/using-doxygen/>

Sujet :

Une banque gère des comptes bancaires de clients dont on connaît le prénom, le nom et la date de naissance. La banque propose différents type de comptes bancaires : COURANT, CODEVI, CEL, EPARGNE qui se caractérisent par un taux de rémunération, un solde minimum possible et un solde maximum possible. Un client peut évidemment posséder plusieurs comptes.

Quand le client crée un compte(d'un certain type), il dépose une certaine somme d'argent qui constitue son solde initial. Il peut ensuite effectuer les opérations suivantes :

- dépôt : ajouter de l'argent sur le compte
- retrait : retirer de l'argent sur le compte
- transfert : transférer de l'argent d'un de ses comptes sur un de ses autres comptes

Une opération (création, dépôt, retrait, transfert) ne peut être effectuée que si le solde des différents comptes mis en jeu dans l'opération et leurs caractéristiques le permette. Une opération qui ne peut pas être effectuée est toujours amicalement refusée.

Pour chaque compte, on peut consulter le relevé des 5 dernières opérations effectuées sur ce compte.

Question 1

Analyser le sujet pour identifier les différents objets et leurs interactions avec les autres objets. Déterminer les différentes classes, leurs attributs et leurs méthodes qui permettent de prendre en compte tout le sujet (voir aussi l'exemple ci-dessous). Dessiner un modèle UML qui résume l'ensemble des classes du sujet et leurs interactions.

Question 2

Implémenter toutes ces classes.

Appliquer le principe d'encapsulation à toutes les classes de votre projet. Implémenter tous les accesseurs utiles pour l'utilisation de ces classes. Bien faire la distinction entre les méthodes qui devraient être "const" et les autres.

Implémenter tous les constructeurs utiles pour l'initialisation des objets utilisés dans votre projet. En particulier, on sera attentif à la validité des valeurs qui ont été affectées aux attributs de chaque objet après leur création. Utiliser au mieux la syntaxe d'initialisation avec ":". Implémenter des destructeurs si nécessaire.

Utiliser des tableaux alloués dynamiquement. Ces tableaux pourront éventuellement contenir des pointeurs : les objets pourront alors être alloués au fur et à mesure des besoins. Modifier les constructeurs et destructeurs si nécessaire.

Question 3 (à faire s'il reste du temps)

La banque rémunère tous les comptes (qui ont un taux de rémunération non nul) de la façon suivante. Les intérêts d'un compte sont calculés tous les 15 jours selon la formule suivante :

$\text{intérêts} = (\text{solde courant} * \text{taux de rémunération}) / 24.$

Ces intérêts ne sont pas versés sur le compte mais cumulés avec les intérêts calculés les quinzaines précédentes dans un enregistrement de cumul d'intérêt.

Une fois par an (à la fin de l'année), les intérêts sont reversés sur le compte et l'enregistrement de cumul d'intérêt remis à zéro.

Implémenter l'ensemble des méthodes et attributs qui permettrait de calculer les intérêts d'une quinzaine (processus utilisé par la banque une fois tous les quinze jours) et de reverser les intérêts cumulés (processus utilisé une fois par an).

Exemple

Voici un exemple de programme principal qui pourrait utiliser un tel ensemble de classes :

```
#include "banque.h"
#include <cstdlib>
int main(){
    using namespace BANQUE;
    // Une Banque
    Banque BNP; BNP.Initialiser("BNP");

    // Ses différents types de comptes
    BNP.CreerUnTypeDeCompte("CHEQUE",-500,9999999,0);
    BNP.CreerUnTypeDeCompte("EPARGNE",15,9999999,1);
    BNP.CreerUnTypeDeCompte("LIVRET A",15,15300,1.25);
    BNP.CreerUnTypeDeCompte("CEL",300,15300,0.75);
    BNP.CreerUnTypeDeCompte("CODEVI",50,6000,1.25);

    // Ses clients
    BNP.AjouterUnClient("Clement","Fouque",30,6,1987);
    BNP.AjouterUnClient("Amjad","Abou-Assali",19,12,1986);
    BNP.AjouterUnClient("Sofiane","Khalfallah",31,5,1986);
    BNP.AjouterUnClient("Cuong","Duc Dang",18,11,1990);
    BNP.AjouterUnClient("Sergio","Rodriguez",24,8,1987);

    // Ses comptes
    BNP.CreerUnCompte("CODEVI","Clement","Fouque",400);
    BNP.CreerUnCompte("CHEQUE","Clement","Fouque",15);
```

```

BNP.CreerUnCompte( "LIVRET A", "Clement", "Fouque", 2000);
BNP.CreerUnCompte( "CHEQUE", "Amjad", "Abou-Assali", 3000);
BNP.CreerUnCompte( "CEL", "Amjad", "Abou-Assali", 1000);
BNP.CreerUnCompte( "EPARGNE", "Amjad", "Abou-Assali", 200);
BNP.CreerUnCompte( "CHEQUE", "Sofiane", "Khalfallah", 500);
BNP.CreerUnCompte( "LIVRET A", "Sofiane", "Khalfallah", 800);
BNP.CreerUnCompte( "CODEVI", "Sofiane", "Khalfallah", 6000);
BNP.CreerUnCompte( "CHEQUE", "Cuong", "Duc Dang", 15000);
BNP.CreerUnCompte( "CODEVI", "Cuong", "Duc Dang", 150);
BNP.CreerUnCompte( "EPARGNE", "Cuong", "Duc Dang", 1500);
BNP.CreerUnCompte( "CHEQUE", "Sergio", "Rodriguez", 4000);
BNP.CreerUnCompte( "CEL", "Sergio", "Rodriguez", 4000);
BNP.CreerUnCompte( "CHEQUE", "Sergio", "Rodriguez", 6000);

// Quelques opérations
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Deposer(3000);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(100);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(40);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(50);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(100);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(20);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(1500);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(140);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(200);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(200);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").Retirer(400);
BNP.GetCompte( "CHEQUE", "Clement", "Fouque").
TransfererVers( 200, BNP.GetCompte( "CODEVI", "Clement", "Fouque" ));
// Affichage
BNP.Afficher();
std::system( "pause" );
return 0;
}

```

Ci-dessous un exemple de l'affichage qui pourrait être généré :

```

----- BNP -----
CLIENTS :
Clement Fouque (30/06/1987)
Amjad Abou-Assali (19/12/1986)
Sofiane Khalfallah (31/05/1986)
Cuong Duc Dang (18/11/1990)
Sergio Rodriguez (24/08/1987)
COMPTE :
=====
COMPTE NUM 0
PROPRIETAIRE :Clement Fouque (30/06/1987)
compte CODEVI remunere au taux de 1.250000 %
PLAFOND :6000.000000 SOLDE MINIMUM AUTORISE :50.000000
SOLDE ACTUEL :600.000000
HISTORIQUE :
15/03/2010 -- reception transfert 200.00 du compte CHEQUE 1
15/03/2010 --          creation 400.00
=====
=====
COMPTE NUM 1
PROPRIETAIRE :Clement Fouque (30/06/1987)

```

compte CHEQUE PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :-500.00
SOLDE ACTUEL :65.00

HISTORIQUE :

15/03/2010 -- transfert 200.00 vers le compte CODEVI 0
15/03/2010 -- retrait 400.00
15/03/2010 -- retrait 200.00
15/03/2010 -- retrait 200.00
15/03/2010 -- retrait 140.00

=====

=====

COMPTE NUM 2

PROPRIETAIRE :Clement Fouque (30/06/1987)

compte LIVRET A remunere au taux de 1.25 %

PLAFOND :15300.00 SOLDE MINIMUM AUTORISE :15.00

SOLDE ACTUEL :2000.00

HISTORIQUE :

15/03/2010 -- creation 2000.00

=====

=====

COMPTE NUM 3

PROPRIETAIRE :Amjad Abou-Assali (19/12/1986)

compte CHEQUE PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :-500.00

SOLDE ACTUEL :3000.00

HISTORIQUE :

15/03/2010 -- creation 3000.00

=====

=====

COMPTE NUM 4

PROPRIETAIRE :Amjad Abou-Assali (19/12/1986)

compte CEL remunere au taux de 0.75 %

PLAFOND :15300.00 SOLDE MINIMUM AUTORISE :300.00

SOLDE ACTUEL :1000.00

HISTORIQUE :

15/03/2010 -- creation 1000.00

=====

=====

COMPTE NUM 5

PROPRIETAIRE :Amjad Abou-Assali (19/12/1986)

compte EPARGNE remunere au taux de 1.00 %

PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :15.00

SOLDE ACTUEL :200.00

HISTORIQUE :

15/03/2010 -- creation 200.00

=====

=====

COMPTE NUM 6

PROPRIETAIRE :Sofiane Khalfallah (31/05/1986)

compte CHEQUE PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :-500.00

SOLDE ACTUEL :500.00

HISTORIQUE :

15/03/2010 -- creation 500.00

=====

```
=====
COMPTE NUM 7
PROPRIETAIRE :Sofiane Khalfallah (31/05/1986)
compte LIVRET A remunere au taux de 1.25 %
PLAFOND :15300.00 SOLDE MINIMUM AUTORISE :15.00
SOLDE ACTUEL :800.00
HISTORIQUE :
15/03/2010 --      creation 800.00
=====
=====

COMPTE NUM 8
PROPRIETAIRE :Sofiane Khalfallah (31/05/1986)
compte CODEVI remunere au taux de 1.25 %
PLAFOND :6000.00 SOLDE MINIMUM AUTORISE :50.00
SOLDE ACTUEL :6000.00
HISTORIQUE :
15/03/2010 --      creation 6000.00
=====
=====

COMPTE NUM 9
PROPRIETAIRE :Cuong Duc Dang (18/11/1990)
compte CHEQUE PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :-500.00
SOLDE ACTUEL :15000.00
HISTORIQUE :
15/03/2010 --      creation 15000.00
=====
=====

COMPTE NUM 10
PROPRIETAIRE :Cuong Duc Dang (18/11/1990)
compte CODEVI remunere au taux de 1.25 %
PLAFOND :6000.00 SOLDE MINIMUM AUTORISE :50.00
SOLDE ACTUEL :150.00
HISTORIQUE :
15/03/2010 --      creation 150.00
=====
=====

COMPTE NUM 11
PROPRIETAIRE :Cuong Duc Dang (18/11/1990)
compte EPARGNE remunere au taux de 1.00 %
PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :15.00
SOLDE ACTUEL :1500.00
HISTORIQUE :
15/03/2010 --      creation 1500.00
=====
=====

COMPTE NUM 12
PROPRIETAIRE :Sergio Rodriguez (24/08/1987)
compte CHEQUE PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :-500.00
SOLDE ACTUEL :4000.00
HISTORIQUE :
15/03/2010 --      creation 4000.00
=====
```

=====

COMPTE NUM 13

PROPRIETAIRE :Sergio Rodriguez (24/08/1987)

compte CEL remunere au taux de 0.75 %

PLAFOND :15300.00 SOLDE MINIMUM AUTORISE :300.00

SOLDE ACTUEL :4000.00

HISTORIQUE :

15/03/2010 -- creation 4000.00

=====

=====

COMPTE NUM 14

PROPRIETAIRE :Sergio Rodriguez (24/08/1987)

compte CHEQUE PLAFOND :9999999.00 SOLDE MINIMUM AUTORISE :-500.00

SOLDE ACTUEL :6000.00

HISTORIQUE :

15/03/2010 -- creation 6000.00

=====

Appuyez sur une touche pour continuer...