Université de Nice-Sophia Antipolis — 20 décembre 2017
Polytech ELEC4

Mini-Projet C++ : Matrices

---

1) Complétez votre classe `MatriceCreuse` commencée en DS avec la méthode **void** `setValeur(`**int** `i,` **int** `j,` **double** `v)`, son constructeur de copie, la surcharge de l'opérateur d'affectation et toutes autres méthodes que vous jugerez utiles. Remarque : si vous le souhaitez, à la place de la liste, vous pouvez utiliser un vecteur (classe générique `vector`) pour mémoriser la suite de triplets.

2) Ajoutez à la classe `MatriceCreuse` la méthode `tauxRemplissage` qui renvoie le taux de remplissage d'éléments différents de 0 de la matrice creuse courante. Ce taux est compris en 0 et 1.

2) Écrivez la classe abstraite `MatriceAbstraite` qui possédera, entre autres, les méthodes abstraites `getValeur` et `setValeur`. Votre classe `MatriceCreuse` héritera de `MatriceAbstraite`.

3) Écrivez la classe `Matrice`, héritière de `MatriceAbstraite`, qui implémente une matrice $M \times N$ à l'aide d'un tableau à deux dimensions de réels double. Vous écrirez toutes les méthodes nécessaires.

4) Vous gérerez des exceptions lorsque les valeurs des indices, pour accéder à un élément d'une matrice, ne sont pas dans le bon intervalle.

5) Ajoutez à la classe abstraite `MatriceAbstraite` la méthode `identite` qui initialise la matrice courante à matrice identité. Notez que la matrice courante doit être carrée.

6) Ajoutez la surcharge de l'opérateur `*` pour faire le produit de deux matrices $A(M \times P)$ et $B(P \times N)$ et qui renvoie la matrice $C(M \times N) = A(M \times P) \times B(P \times N)$.

7) Votre programme devra être en capacité d'exécuter la fonction `main` suivante :

```
int main() {
  const int M=3, N=4, P=4;
  Matrice<M,N> m1;

  for (int i=0; i<M; i++)
    for (int j=0; j<N; j++)
      m1.setValeur(i, j, i+j);

  std::cout << "----------- m1 -----------" << std::endl;
  std::cout << m1;

  MatriceCreuse<N,P> m2;
  m2.identite();
  m2.setValeur(0, P-1, 5);
```



```
  std::cout << "----------- m2 -----------" << std::endl;
  std::cout << m2 << std::endl;
  std::cout << "taux remplissage : " << m2.tauxRemplissage() << std::endl;

  MatriceAbstraite<M,P> *m3 = m1*m2;
  std::cout << "----------- m3 = m1*m2----" << std::endl;
  std::cout << *m3;

  Matrice<N,P> m4 = m2;
  std::cout << "----------- m4 -----------" << std::endl;
  std::cout << m4;

  Matrice<M,N> m5;
  m5 = m1;
  m1.setValeur(0, 0, 9);
  std::cout << "----------- m1 -----------" << std::endl;
  std::cout << m1;
  std::cout << "----------- m5 -----------" << std::endl;
  std::cout << m5;

  MatriceCreuse<M,N> m6;

  for (int i=0; i<M; i++)
    for (int j=0; j<N; j++)
      m6.setValeur(i, j, i+j);

  std::cout << "----------- m6 -----------" << std::endl;
  std::cout << m6 << std::endl;
  std::cout << "taux remplissage : " << m6.tauxRemplissage() << std::endl;

  if (m6.tauxRemplissage()>0.5) {
    Matrice<M,N> m7;
    m7 = m6;
    std::cout << "----------- m7 -----------" << std::endl;
    std::cout << m7 << std::endl;
  }
  //
  try {
    std::cout << m6.getValeur(10,10) << std::endl;
  }
  catch (IndexException &e) {
    std::cout << e.what() <<  std::endl;
  }
  // erreur m5 n'est pas une matrice carrée
  std::cout << "------- erreur -----------" << std::endl;
  m5.identite();
  //
  return EXIT_SUCCESS;
}
```

qui écrit sur la sortie standard :

```
----------- m1 -----------
```

```
0 1 2 3
1 2 3 4
2 3 4 5
----------- m2 -----------
1 0 0 5
0 1 0 0
0 0 1 0
0 0 0 1

taux remplissage : 0.3125
----------- m3 = m1*m2----
0 1 2 3
1 2 3 9
2 3 4 15
----------- m4 -----------
1 0 0 5
0 1 0 0
0 0 1 0
0 0 0 1
----------- m1 -----------
9 1 2 3
1 2 3 4
2 3 4 5
----------- m5 -----------
0 1 2 3
1 2 3 4
2 3 4 5
----------- m6 -----------
0 1 2 3
1 2 3 4
2 3 4 5

taux remplissage : 0.916667
----------- m7 -----------
0 1 2 3
1 2 3 4
2 3 4 5

(10,10) : Index out of bounds exception
------- erreur -----------
test: MatriceAbstraite.hpp:54: void MatriceAbstraite<M, N>::identite()
[with int M = 3; int N = 4]: Assertion 'M==N' failed.
zsh: abort (core dumped)  ./test
```

Rendu

Projet à rendre au plus tard le **jeudi 11 janvier 2018**.