

## I Autour de la dichotomie (E3A 2019)

Voici une tentative de code d'une fonction C pour tester par dichotomie si un entier  $e$  se trouve dans un tableau d'entiers  $T$  :

---

```

bool dichotomie(int e, int* T, int n) {
    // Pré-condition : T est un tableau de n entiers trié dans l'ordre croissant
    int g = 0, d = n - 1;
    while (d - g > 0) {
        int m = (g + d) / 2;
        if (T[m] >= e) {
            d = m;
        } else {
            g = m;
        }
    }
    return T[g] == e;
}

```

---

1. Pour quelles raisons ne remplace-t-on pas la précondition par un appel à une fonction qui trierait  $T$  dans l'ordre croissant?
2. Montrer que la fonction `dichotomie` ne termine pas forcément.
3. Indiquer sans justification la ou les corrections à apporter pour que la fonction `dichotomie` termine, tout en restant correcte.
4. Justifier que la fonction corrigée termine et est correcte.

Une première extension consiste à réduire le problème non pas en 2 mais en 3 : c'est le principe de trichotomie.

5. Écrire une fonction `bool trichotomie(int e, int* T, int n)` qui renvoie `true` si l'élément  $e$  se trouve dans  $T$  et `false` sinon.
6. Estimer la complexité de `trichotomie(e, T, n)` en fonction de  $n$ . Comparer avec la méthode par dichotomie.

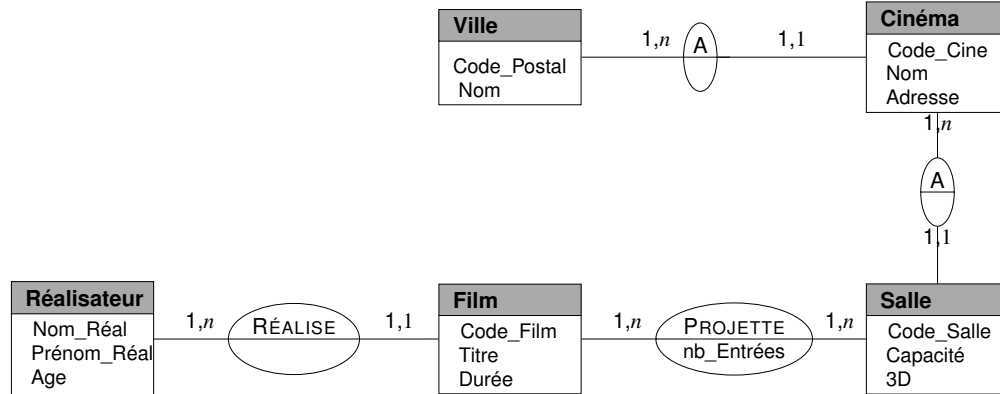
Une seconde extension consiste à adapter le principe de dichotomie au cas d'une matrice d'entiers dont les éléments sont triés en colonne de haut en bas et de gauche à droite. L'illustration ci-dessous indique l'ordre des éléments d'une matrice à 4 lignes et 6 colonnes :

$$\begin{pmatrix} 0 & 4 & 8 & 12 & 16 & 20 \\ 1 & 5 & 9 & 13 & 17 & 21 \\ 2 & 6 & 10 & 14 & 18 & 22 \\ 3 & 7 & 11 & 15 & 19 & 23 \end{pmatrix}$$

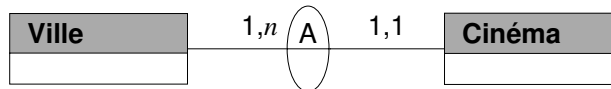
7. Expliquer comment déterminer si un entier  $e$  se trouve dans une telle matrice de taille  $n \times p$ , en complexité logarithmique en  $\max(n, p)$ .

## Partie II - Bases de données

On considère la base de données décrite par le modèle entité-association suivant :



Dans cette représentation, on lie des entités par des associations avec des cardinalités. Ainsi par exemple



peut se lire comme “Une ville a 1 à  $n$  cinéma(s)” et “un cinéma est présent dans 1 et une seule ville”. Parfois, l’association possède des propriétés (le nombre d’entrées d’un film projeté dans une salle par exemple).

On suppose que le champ 3D de l’entité Salle est un entier, valant 0 si la salle n’est pas en 3D, et 1 sinon. On suppose de plus que la durée des films est en heures. Enfin, on affirme qu’une ville peut être uniquement déterminée par son code postal et son nom.

**Q6.** Donner le schéma relationnel correspondant. Préciser les clés primaires et étrangères des relations. Les clés primaires peuvent être associées à plusieurs attributs.

**Q7.** Écrire les requêtes suivantes en langage SQL :

- (i). Donner le titre des films durant moins de 2h.
- (ii). Donner le nom et le prénom du réalisateur ayant réalisé le film “Matrix”.
- (iii). Compter le nombre de cinémas à Nantes.
- (iv). Donner l’adresse des cinémas contenant au moins une salle 3D.
- (v). Donner le code des films projetés dans toutes les salles.
- (vi). Donner la liste des titres des films projetés dans le cinéma “Le Rio”.