

Graphes

Représentation carte Course d'Orientation

Christophe Viroulaud

Terminale - NSI

Algo 13

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

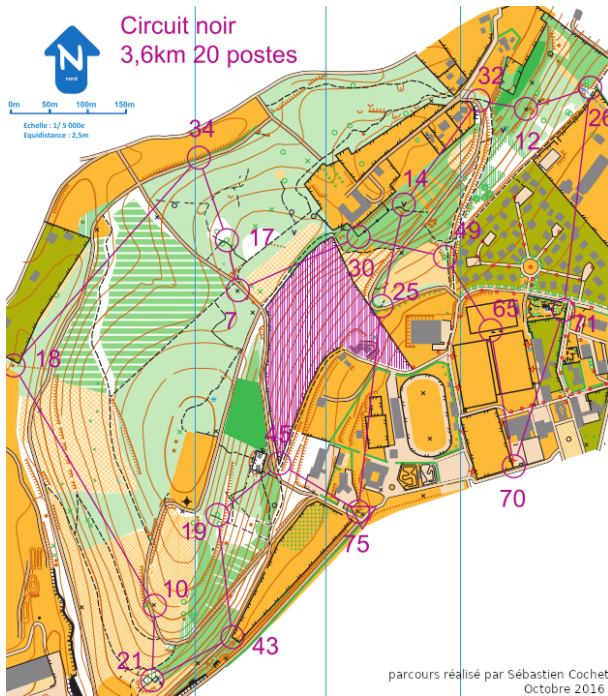
Passage d'une structure à
l'autre

Représentation de la carte
de CO



Circuit noir 3,6km 20 postes

Echelle : 1/5 000e
Équidistance : 2,5m



parcours réalisé par Sébastien Cochet
Octobre 2016

Graphes Représentation carte Course d'Orientation

Notion de graphe

Vocabulaire

Propriétés

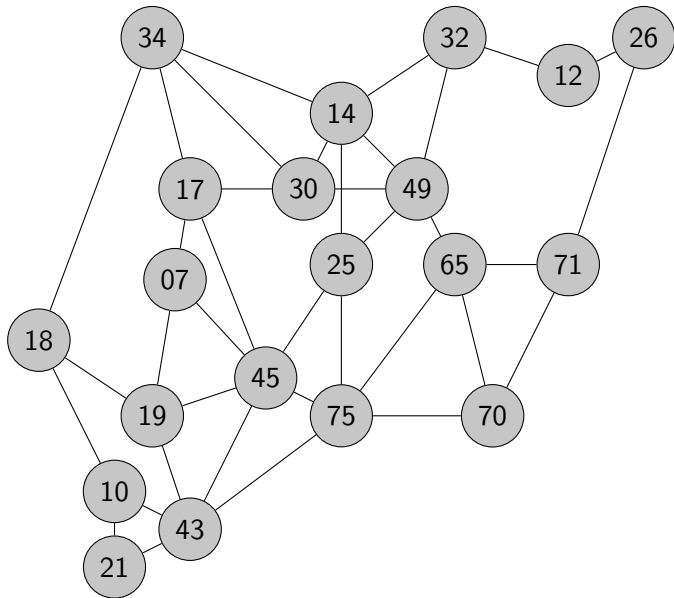
Représentations en mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à l'autre

Représentation de la carte de CO



Notion de graphe

Vocabulaire

Propriétés

Représentations en mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à l'autre

Représentation de la carte de CO

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

Comment représenter un graphe en mémoire ?

1. Notion de graphe

1.1 Vocabulaire

1.2 Propriétés

2. Représentations en mémoire

Notion de graphe

Vocabulaire

Propriétés

Représentations en mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à l'autre

Représentation de la carte de CO

Notion de graphe

Vocabulaire

Propriétés

Représentations en mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à l'autre

Représentation de la carte de CO

Un graphe est défini par :

- ▶ ses **sommets** (ou **nœuds**),
- ▶ ses **arêtes** (ou **arcs**) qui relient deux sommets.

À retenir

- ▶ L'**ordre** du graphe est le nombre de ses sommets.
- ▶ Un graphe est **non orienté** quand ses arêtes peuvent être parcourues dans les deux sens.

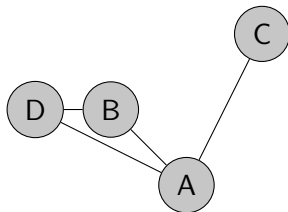


FIGURE 1 – Graphe non orienté d'ordre 4

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

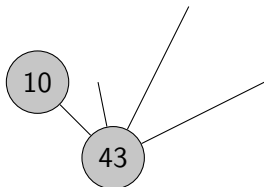
Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

À retenir

- ▶ Deux sommets reliés par une arête sont **adjacents**.



Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

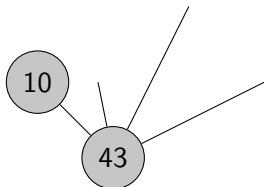
Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

À retenir

- ▶ Deux sommets reliés par une arête sont **adjacents**.
- ▶ Le **degré d'un sommet** est le nombre d'arêtes de ce sommet.



Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

À retenir

Un graphe est **complet** si tous les sommets sont adjacents à tous les autres.

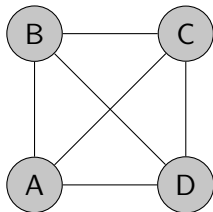


FIGURE 2 – Graphe complet d'ordre 4

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

1. Notion de graphe

1.1 Vocabulaire

1.2 Propriétés

2. Représentations en mémoire

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

À retenir

La somme des degrés d'un graphe est pair.

$$\sum_{s \in S} \text{deg}(s) = 2.A$$

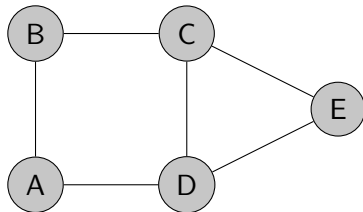


FIGURE 3 – Chaque arête est comptée deux fois.

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

À retenir

Un arbre est un graphe qui ne possède pas de cycle.

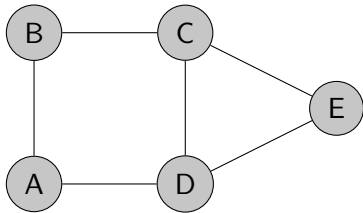


FIGURE 4 – Graphe avec au moins un cycle.

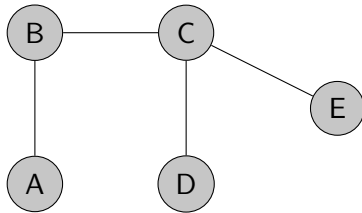


FIGURE 5 – Arbre

Notion de graphe

Vocabulaire

Propriétés

Représentations en mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à l'autre

Représentation de la carte de CO

1. Notion de graphe

2. Représentations en mémoire

2.1 Matrice d'adjacence

2.2 Dictionnaire d'adjacence

2.3 Passage d'une structure à l'autre

2.4 Représentation de la carte de CO

Notion de graphe

Vocabulaire

Propriétés

Représentations en mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

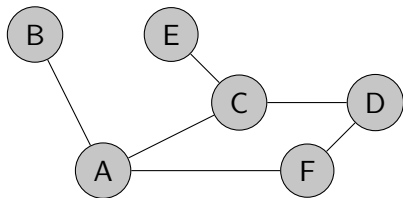
Passage d'une structure à l'autre

Représentation de la carte de CO

Représentations en mémoire - matrice d'adjacence

À retenir

La **matrice d'adjacence** est la représentation mathématique dont le terme a_{ij} vaut 1 si les sommets i et j sont reliés par une arête et 0 sinon.



$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autreReprésentation de la carte
de CO

	A	B	C	D	E	F
A	0	1	1	0	0	1
B	1	0	0	0	0	0
C	1	0	0	1	1	0
D	0	0	1	0	0	1
E	0	0	1	0	0	0
F	1	0	0	1	0	0

Remarque

Dans un graphe non orienté la matrice est symétrique.

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

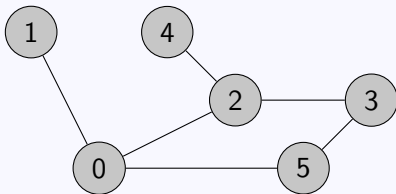
Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

Activité 1 :

1. Déterminer une structure de données permettant de représenter en mémoire la matrice d'adjacence représentative d'un graphe dont les sommets sont des entiers commençant à 0.
2. Construire la matrice d'adjacence du graphe suivant :



3. Écrire la fonction `est_symetrique(mat: list)` → `bool` qui renvoie `True` si la matrice est symétrique.

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

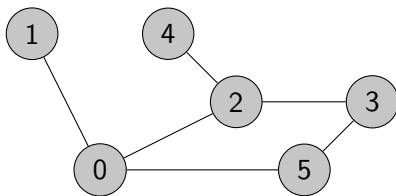
Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

Correction



```
1 # Le noeud A est représenté par la ligne 0
2 mat = [ [0, 1, 1, 0, 0, 1],
3         [1, 0, 0, 0, 0, 0],
4         [1, 0, 0, 1, 1, 0],
5         [0, 0, 1, 0, 0, 1],
6         [0, 0, 1, 0, 0, 0],
7         [1, 0, 0, 1, 0, 0] ]
```

Code 1 – Tableau de tableau

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

Observation

Cette représentation peut être gourmande en mémoire : si le nombre d'arêtes est faible, la structure contient peu d'informations. La matrice est **creuse**.

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

```
1 def est_symetrique(mat: list) -> bool:
2     for i in range(len(mat)):
3         for j in range(len(mat)):
4             if mat[i][j] != mat[j][i]:
5                 return False
6     return True
```

```
1 def est_symetrique(mat: list) -> bool:
2     for i in range(len(mat)):
3         # on limite le parcours au triangle haut
4         for j in range(i, len(mat)):
5             if mat[i][j] != mat[j][i]:
6                 return False
7     return True
```

Code 2 – version optimisée : inutile de parcourir toute la matrice

1. Notion de graphe

2. Représentations en mémoire

2.1 Matrice d'adjacence

2.2 Dictionnaire d'adjacence

2.3 Passage d'une structure à l'autre

2.4 Représentation de la carte de CO

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

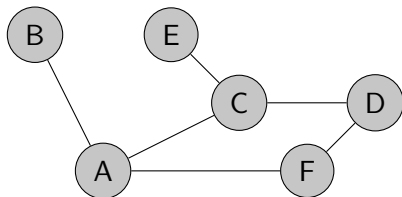
Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

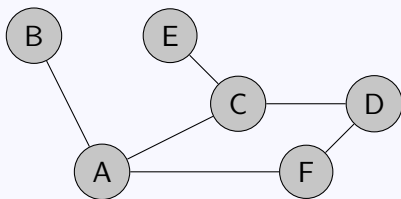
À retenir

Un dictionnaire d'adjacence liste les sommets adjacents à chaque sommet.



- ▶ A : B, C, F
- ▶ B : A
- ▶ C : A, D, E
- ▶ D : C, F
- ▶ E : C
- ▶ F : A, D

Activité 2 : Construire le dictionnaire d'adjacence en Python du graphe suivant :



Notion de graphe

Vocabulaire

Propriétés

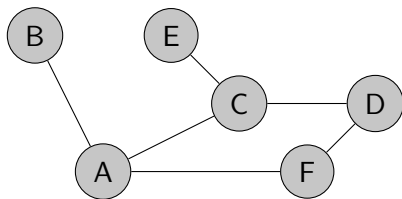
Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO



```
1 dico = {"A": ["B", "C", "F"],  
2         "B": ["A"],  
3         "C": ["A", "D", "E"],  
4         "D": ["C", "F"],  
5         "E": ["C"],  
6         "F": ["A", "D"]}
```

Code 3 – Dictionnaire de tableau

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

1. Notion de graphe

2. Représentations en mémoire

2.1 Matrice d'adjacence

2.2 Dictionnaire d'adjacence

2.3 Passage d'une structure à l'autre

2.4 Représentation de la carte de CO

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

**Passage d'une structure à
l'autre**

Représentation de la carte
de CO

Activité 3 : Écrire la fonction `mat_to_dic(mat: list) → dict` qui construit le dictionnaire d'adjacence à partir de la matrice d'adjacence.

Indication : Les nœuds sont nommés en suivant l'ordre alphabétique majuscule. La première ligne de la matrice représente les adjacences de **A**. La fonction native `chr(n: int) → str` renvoie le caractère correspondant au point de code UTF-8 **n**.

```
1 def mat_to_dic(mat: list) -> dict:
2     dico = {}
3     for i in range(len(mat)):
4         # nom du noeud
5         noeud = chr(65+i)
6         dico[noeud] = []
7
8         for j in range(len(mat[i])):
9             if mat[i][j] == 1:
10                # noeud adjacent
11                adj = chr(65+j)
12                dico[noeud].append(adj)
13     return dico
```

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

Activité 4 : Écrire la fonction `dic_to_mat(dic: dict) → list` qui construit la matrice d'adjacence à partir de la matrice d'adjacence.

Indication : La fonction native `ord(c: str) → int` renvoie le point de code UTF-8 correspondant au caractère `c`.

```
1 def dic_to_mat(dic: dict) -> list:
2     # taille de la matrice connue
3     mat = [ [0 for _ in range(len(dic))]
4             for _ in range(len(dic)) ]
5     for noeud, adjacents in dic.items():
6         # indice de la ligne
7         ind_noeud = ord(noeud)-65
8
9         for adj in adjacents:
10            # indice de la colonne
11            ind_adj = ord(adj)-65
12            mat[ind_noeud][ind_adj] = 1
13     return mat
```

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

1. Notion de graphe

2. Représentations en mémoire

2.1 Matrice d'adjacence

2.2 Dictionnaire d'adjacence

2.3 Passage d'une structure à l'autre

2.4 Représentation de la carte de CO

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

Activité 5 : L'organisme qui maintient les cartes à jour stocke les informations dans un fichier `json`.

1. Télécharger le dossier compressé `representation-co.zip` et extraire le fichier `parcours_noir.json`
2. Ouvrir le fichier et observer la structure des données.
3. Créer le fichier `parcours_noir.py`
4. Importer les données `json`.
5. Créer le dictionnaire d'adjacence associé à la carte de CO de la forme

```
1 {7: [17, 19, 45], 10: [18, 21, 43],  
   12: [26, 32], ...}
```

Notion de graphe

Vocabulaire

Propriétés

Représentations en
mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à
l'autre

Représentation de la carte
de CO

Notion de graphe

Vocabulaire

Propriétés

Représentations en mémoire

Matrice d'adjacence

Dictionnaire d'adjacence

Passage d'une structure à l'autre

Représentation de la carte de CO

```
1 import json
2
3 f = open("parcours_noir.json")
4 donnees = json.load(f) # tableau de dictionnaires
5 dico_adj = {}
6 for info in donnees:
7     sommet = info["sommet"]
8     adjacents = info["adjacents"]
9     dico_adj[sommet] = adjacents
10 f.close()
```