

---

# Un jour au ski

Agathe Pascal et Nastassia Bonetti



---

# Sommaire

- I. Objectif du projet
- II. Recherche de solution
- III. Comparaison
- IV. Vision du projet



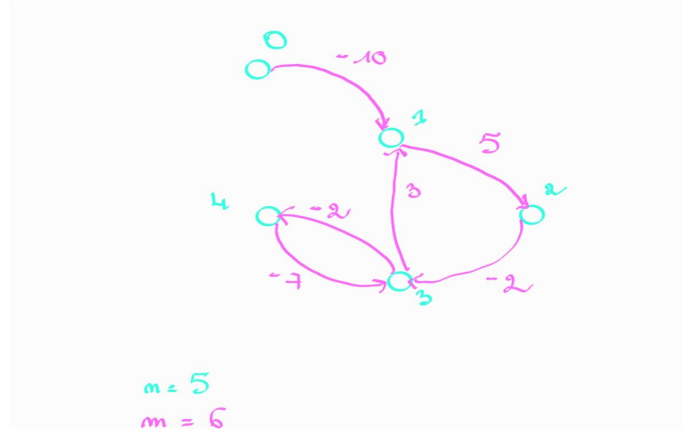
# I.objectif du projet

Rechercher la quantité maximale de plaisir que l'on peut prendre.

Si jamais cette quantité est infinie on affichera : "SKY IS THE LIMIT"

Hypothèse: sommet de départ = 0

Graphe du problème



## II. recherche de solution

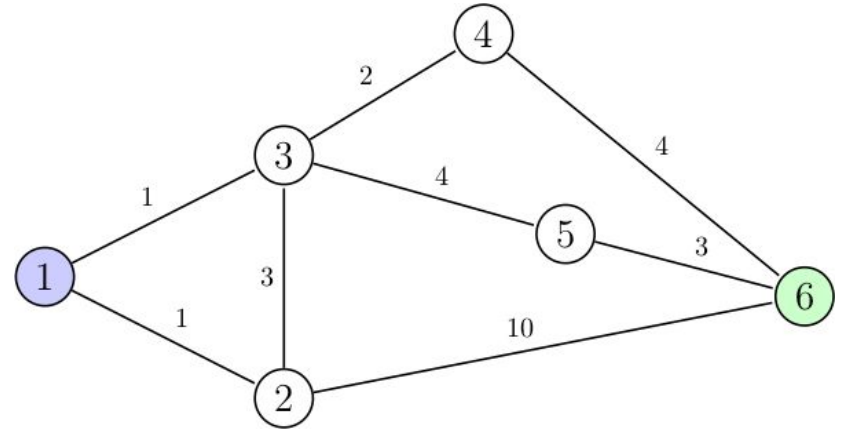
1. Lire les données

→ fichier.txt

2. Utiliser les données

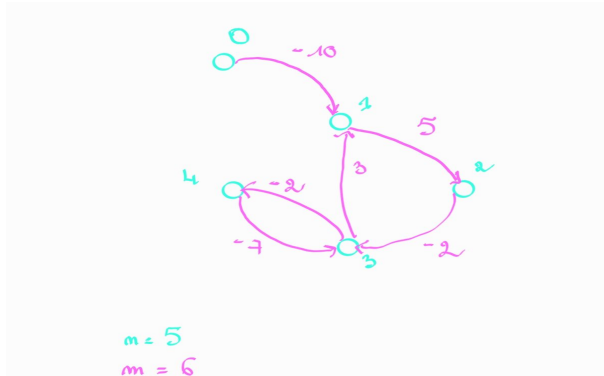
-recherche de cycles

-parcours du graphe et calcul du plaisir maximal



# 1. Lecture des données

- plusieurs tentatives: 1 échec et 1 réussite
- représentation en matrice adjacente
- chaque valeur=plaisir associé entre deux sommets



Exemple

0	-10	0	0	0
0	0	5	0	0
0	0	0	-2	0
0	3	0	0	-2
0	0	0	-7	0



## 2. Recherches de cycles

### Version1

- détection de cycle grâce à un parcours en profondeur de la matrice
- 
- tentative de récupérer tous les cycles pour ensuite calculer le plaisir associé ( trop complexe)
- 
- récupérer tous les sommets du cycle
- 
- calcul du plaisir pour chaque cycle trouvé

### Version 3

- une fonction qui détecte directement la présence de cycle positif dans le graphe
- calcul du plaisir associé directement lors du parcours
- utilisation d'un compteur qui indiquera si on a un cycle



## 3.parcours du graphe/calcul plaisir

### Version1

- parcours du graphe en profondeur grâce à la récursivité
- tableau de structure des chemin {chemin ,taille}
- calculer pour chaque chemin le plaisir
- puis récupération du maximum
- liste avec tous les plaisirs
- récupère la taille et la liste dans une structure {liste, taille de la liste}

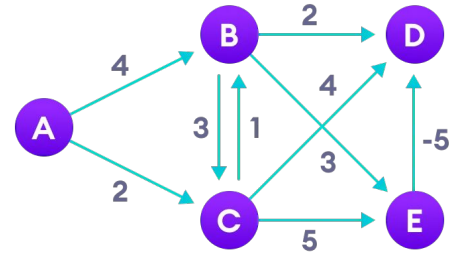
### Version3

- idée de base de la version 1 sans la complication de gestion des données
- utilisation d'un compteur qui compte le plaisir des chemins
- on compare à chaque fois si le plaisir calculé est supérieur ou non

## II.recherche de solution

### Solution avec Bellman Ford

1. Initialisation
2. Relaxation des arêtes
3. Détection des cycles
4. Récupération du plus long chemin





# Solution Bellman Ford

étape 1: on choisit l'origine 0 et on initialise les plaisirs à l'infini.

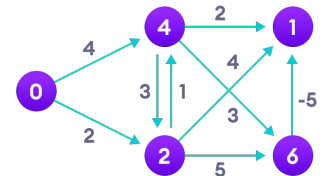
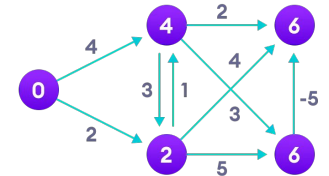
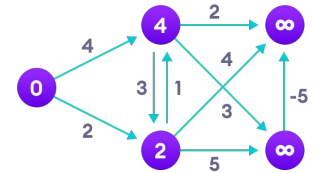
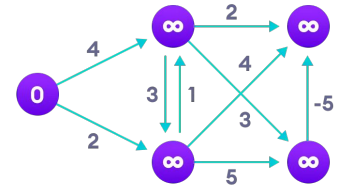
étape 2: on visit chaque sommet et on met le plaisir associé au chemin effectué

étape 3: on fait cette étape V fois

étape 4: on change les plaisirs sur chaque sommet

étape 5: on obtient le tableau des plaisirs et on regarde s'il y a un cycle

	B	C	D	E
0	$\infty$	$\infty$	$\infty$	$\infty$
0	4	2	$\infty$	$\infty$
0	3	2	6	6
0	3	2	1	6
0	3	2	1	6





## III. Comparaison

### Nos programmes

- test sur tous les datas y compris ceux à 1000 pistes OK
- 3ms et 1.9 millions de kbits pour 1000 pistes mais fastidieux à coder

### Bellman Ford

- fonctionne sur tous les data y compris à 1000 pistes
- 282ms et 1.9 millions de kbits pour 1000 pistes et plus facile de coder



## IV. Vision du projet

### Points utiles:

- utilisation du git
  - différentes branches
  - retrouver les anciens commit
- décomposer les programmes en plusieurs sous fonctions
- dans chaque fonction, la tester avec un petit main
- aller au plus simple

### Nouvelles connaissances

- github
- travail sur les programmes récurifs
- fonction ne modifie pas les données en entrée