Algorithmique : le crêpier psychorigide

Référence au programme :

- Tri par sélection
- Notion de coût

Pré - requis :

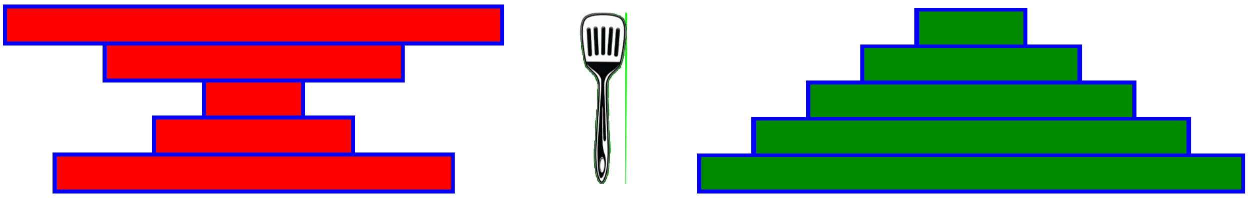
- Parcours séquentiel d’un tableau
- Tri par sélection : terminaison, coût de cet algorithme.

Présentation :

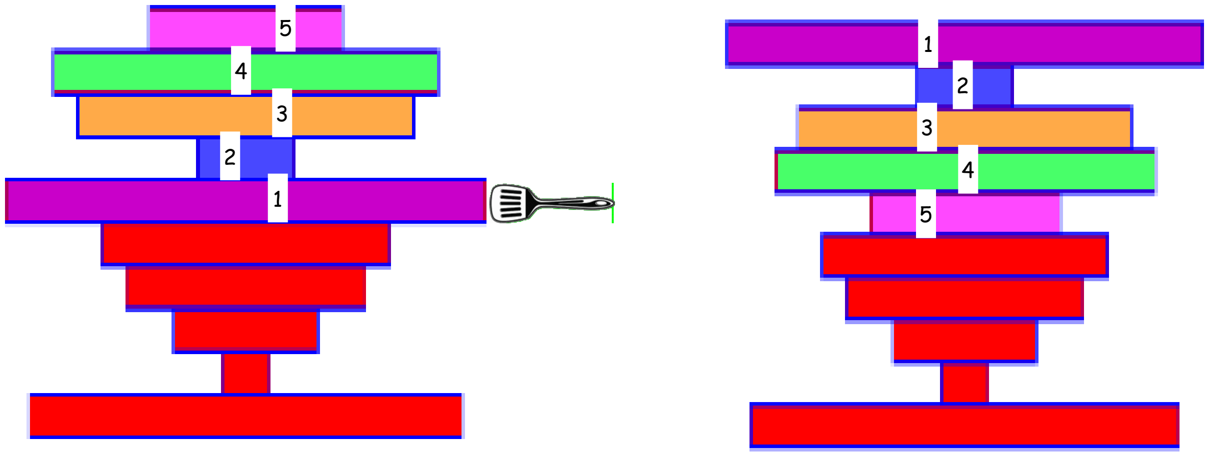
Les ordinateurs sont souvent utilisés pour classer des listes selon un certain ordre, par exemple des noms par ordre alphabétique, des rendez-vous ou des messages par date ou d’autres éléments qu’on peut classer par ordre numérique. Les mathématiciens appellent cela ranger, tandis que les informaticiens appellent cela trier,le terme que nous utiliserons. Le tri des listes nous permet d'en retrouver rapidement les éléments, et il facilite par exemple la recherche des valeurs extrêmes. Si vous n’utilisez pas une bonne méthode, trier une longue liste peut prendre beaucoup de temps, même avec un ordinateur rapide.

1) Activité découverte du tri de crêpes

<u>Objectif du jeu :</u> trier un tas de 5 crêpes dans diamètre croissant



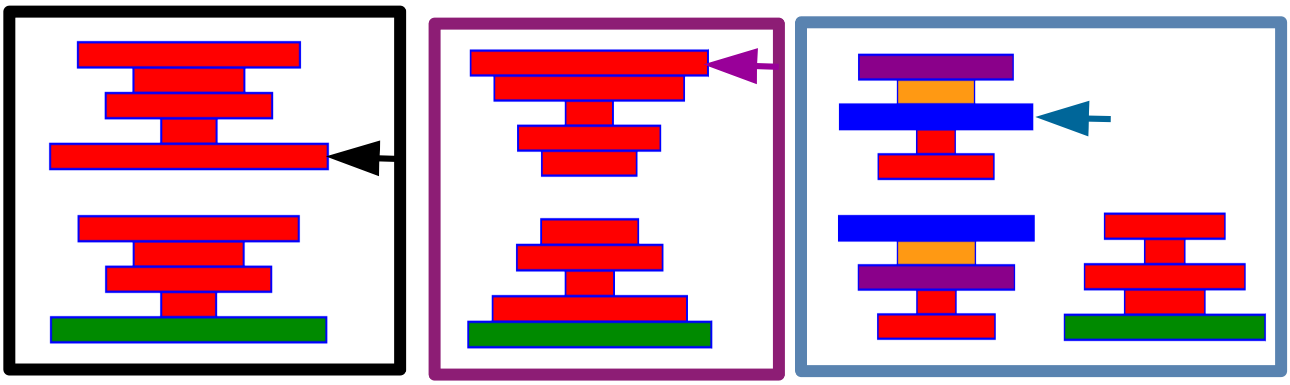
On ne peut que retourner un tas de crêpe (de une à toutes)



<u>Objectif de l'activité :</u>
Décrire en langage naturel comment faire pour résoudre ce problème de tri (notamment où placer la spatule). Puis discussion sur les résultats.

<u>Restitution de l'activité :</u>

On recherche la crêpe **la plus grande** parmi les crêpes déplaçables.
    Si elle est à sa place alors elle devient non déplaçable.
    Sinon si elle est sur le dessus du tas alors
        on retourne les crêpes déplaçables.
        La crêpe déplaçable la plus basse devient non déplaçable.
    Sinon on retourne le tas entre cette crêpe et le sommet.
        On retourne les crêpes déplaçables.
        La crêpe déplaçable la plus basse devient non déplaçable.



A quoi ce tri peut-il nous faire penser ? (tri par sélection)

2) Application de l’algorithme

Sur une suite de nombres :

Appliquer les instructions successives suivantes pour modifier la liste de nombres (qui représentent la taille des crêpes) :

- je cherche le plus grand (en bleu) ;
- je le mets en haut, (comme pour le retournement des crêpes) ;
- je le mets en bas (retournement de toutes les crêpes déplaçables), il devient vert ;
- je reprends au premier point (avec les crêpes déplaçables – celles qui ne sont pas en vert).

5	1	1	**4**	...					
2	3	3	3	...					
4	4	**4**	1	...					
3	2	2	2	**4**					
1	**5**	**5**	**5**	**5**					

Compter le nombre de retournements de tas de crêpes dans l’exemple ci-dessus. (6 ou 7)

L’algorithme s’arrête-t-il à la 5ème colonne ?

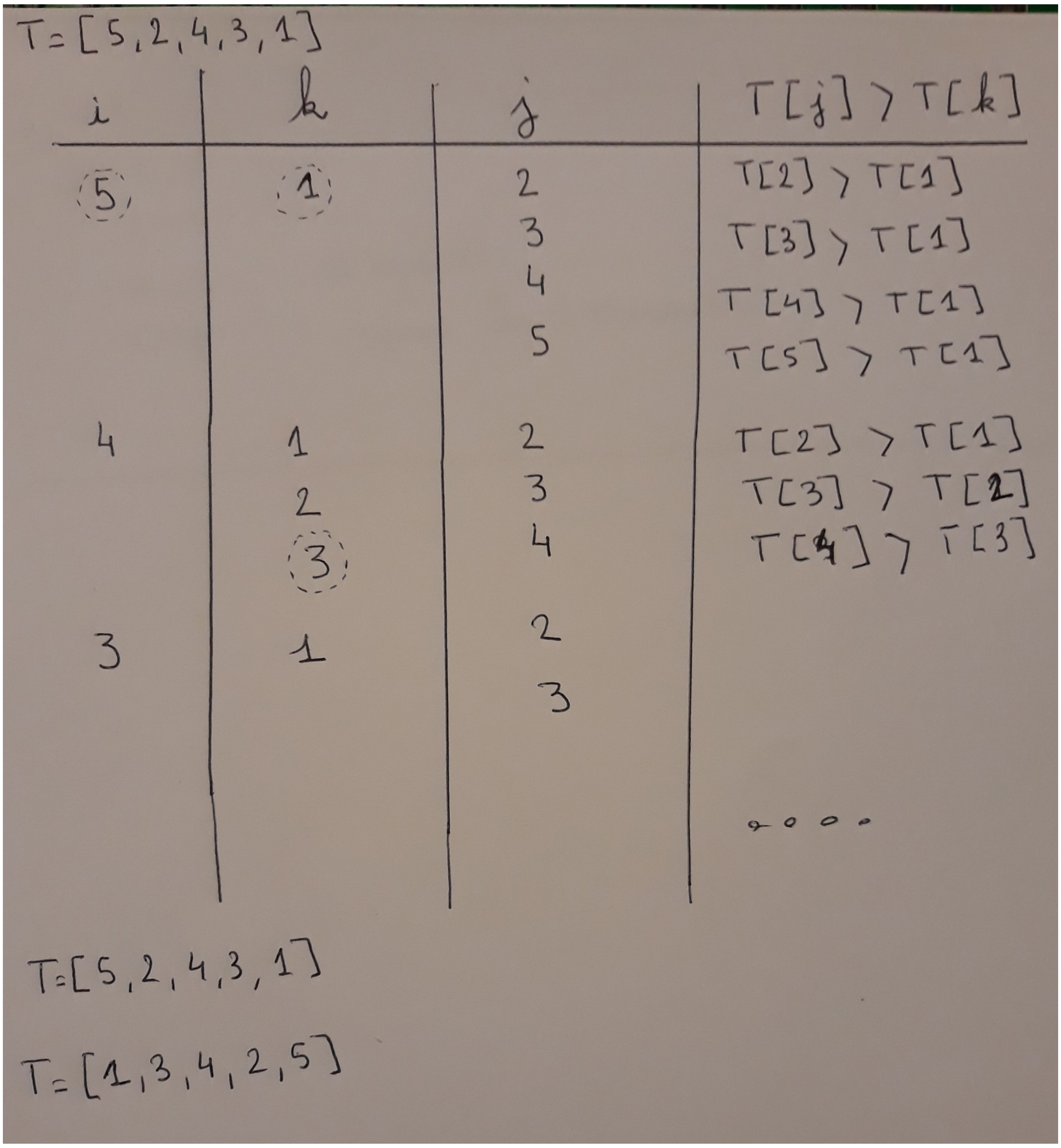
Passage au cas général : conjecturer le nombre maximal de retournements avec n crêpes si on suit les étapes successives de cet algorithme.

$2n$ retournements (on peut l’affiner en prenant en compte le sommet du tas).

Algorithme en pseudo-code :

Groupe algo IREM Grenoble

```
Algorithme Tri de crêpes
T un tableau d'entiers naturels
Début
    Pour i allant de len(T) a 2 Faire
        k = 1
        Pour j allant de 2 a i Faire
            Si T[j] > T[k] Alors
                k = j
            FinSi
        FinPour
        Retourne(T,k)
        Retourne(T,i)
    FinPour
Fin
```



T=[5,2,4,3,1]

i	k	j	T[j] > T[k]
5	1	2	T[2] > T[1]
		3	T[3] > T[1]
		4	T[4] > T[1]
		5	T[5] > T[1]
4	1	2	T[2] > T[1]
	2	3	T[3] > T[2]
	3	4	T[4] > T[3]
3	1	2	
		3	
			....

T=[5,2,4,3,1]

T=[1,3,4,2,5]

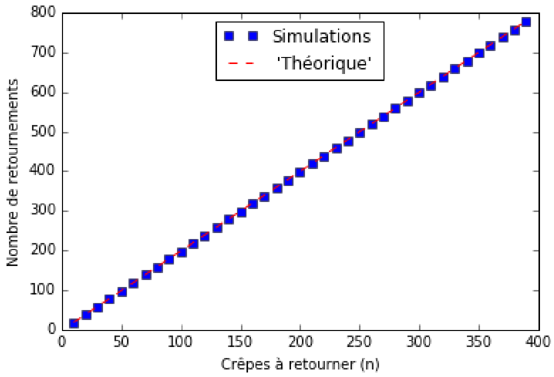
Phase d’expérimentation sur machine :

Ouvrir le programme python, traduction de l’algorithme précédent.

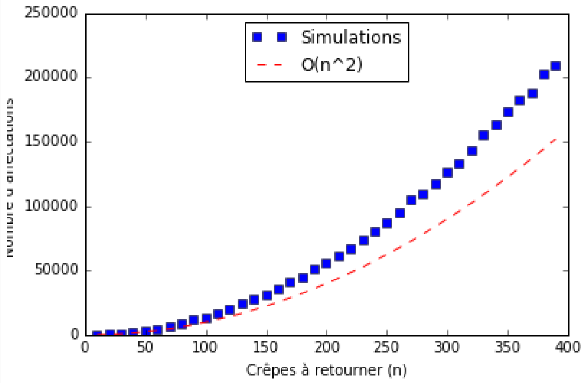
1. Le compléter pour qu’il compte le nombre de retournements effectués (comparer cette valeur avec $2n$).
2. Comment pourrait-on demander au programme de compter le nombre d’affectations ?
3. Modifier le nombre de crêpes et relancer le programme.

Synthèse : présentation de l’évolution du nombre de retournements et d’affectations en fonction du nombre *n* de crêpes.

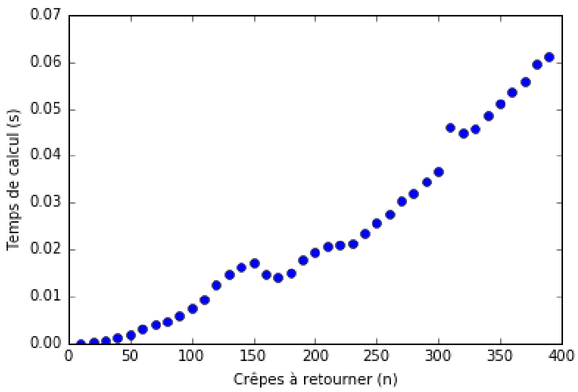
Calcul effectif du nombre de retournements



Nombre d’affectations dans le programme



Qu’en est-il du temps de calcul dans le programme ?

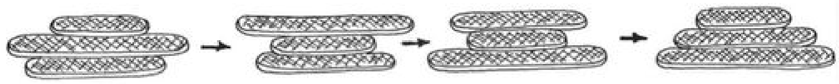


Quel lien peut-on faire avec le tri par sélection ?

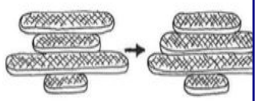
3) Algorithme optimal ?

Quelques cas particuliers :

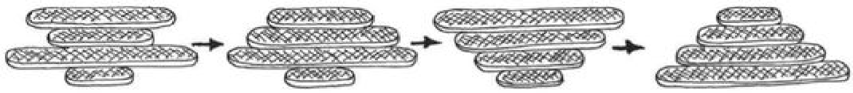
1. Combien faut-il de retournements pour un tas de 3 crêpes, dans le pire des cas (valable dans les 6 configurations possibles) ? Réponse : 3



2. Avec 4 crêpes, dans le pire des cas, 4 retournements seront nécessaires.
   Sur l'exemple ci-dessous, 3 retournements suffisent.



Solution :



3. Configuration avec 5 crêpes (partir des crêpes bien placées, effectuer 3 retournements et demander à l'élève d'appliquer l'algorithme - qui donnera plus de 3 retournements).
   Conclusion : l'algorithme proposé n'est pas optimal.

Quelques éléments historiques du problème :

Bill Gates (créateur de Micro-Soft) et Christos Papadimitriou en 1979, étudiants à Harvard, démontrent dans un article publié dans la revue *Discrete Mathematics*, que pour N crêpes disposées aléatoirement au sein d'une pile il faut, dans le pire des cas, au minimum $\frac{17}{16}N=1{,}0625\,N$ et au maximum $\frac{5N+5}{3}$ opérations.

En 1997, puis en 2008, des chercheurs ont tout de même réussi à démontrer que ce nombre mystère était compris entre $1{,}07 \times N$ et $\frac{18}{11}N \approx 1{,}0636\,N$.

Remarque : Le nombre de configurations de piles de crêpes possibles est $N!$
L'arbre de recherche grandit rapidement. *Remarque: Pour N = 17, on a* $17! \approx 3{,}5 \times 10^{14}$.
Le nombre de retournements optimal (noté $f(N)$) de N crêpes est déterminé actuellement jusqu'à N = 19.

N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$f(N)$	0	1	3	4	5	7	8	9	10	11	13	14	15	16	17	18	19	21	22	**?**

Ces nombres sont trouvés en créant un graphe illustrant la situation. Un algorithme de plus court chemin est ensuite appliqué au graphe.

La taille du graphe limite actuellement la recherche de f(N) à N = 20.

Recherche d'un algorithme plus efficace ?

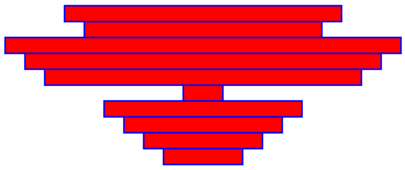
On propose un autre algorithme consistant à placer au meilleur endroit possible la crêpe du haut.

- on la place tout en bas (donc on retourne toutes les crêpes non triées) si c'est la plus grande ;
- on la place au-dessus de la crêpe juste un peu plus grande si elle ne la touche pas ;
- on la place au-dessus de la crêpe juste un peu plus petite si elle ne la touche pas.

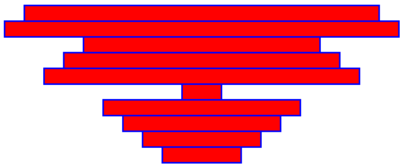
À chaque coup, on se rapproche de la solution en augmentant le nombre de crêpes placées juste à côté de la crêpe à côté de laquelle elle devront se placer.

Ceci nous permet-t-il d'aboutir ? Pas toujours… Voici un exemple (un début de tri ayant déjà été opéré).

Suivant l'algorithme proposé, on doit retourner les 4 premières crêpes :



On obtient alors ceci :



Mais en suivant l'algorithme, on doit de nouveau permuter les 4 premières crêpes ce qui nous ramène à la position précédente et l'algorithme boucle indéfiniment.

Il faut donc prendre des précautions pour ne pas casser ce qu'on réussi à mettre en place à chaque étape : l'algorithme doit permettre de placer une suite de crêpes de taille croissante ou décroissante, il ne faut donc revenir en arrière, ce qui ne peut conduire qu'à empirer la situation (voire la rendre impossible à solutionner).

Voir programme python…

Pour aller plus loin :

Il existe une variante du problème des crêpes : celui des crêpes brûlées. Dans ce problème, l'une des faces de chacune des crêpes est brûlée. Dans l'empilement final, aucune de ces faces ne doit apparaître. La question reste la même.

Quel est le nombre $g(n)$ de retournements, dans le pire des cas à effectuer pour ranger notre pile de crêpes ?

Cas des crêpes brûlées $\frac{3}{2}n \leqslant g(n) \leqslant 2n-2$.

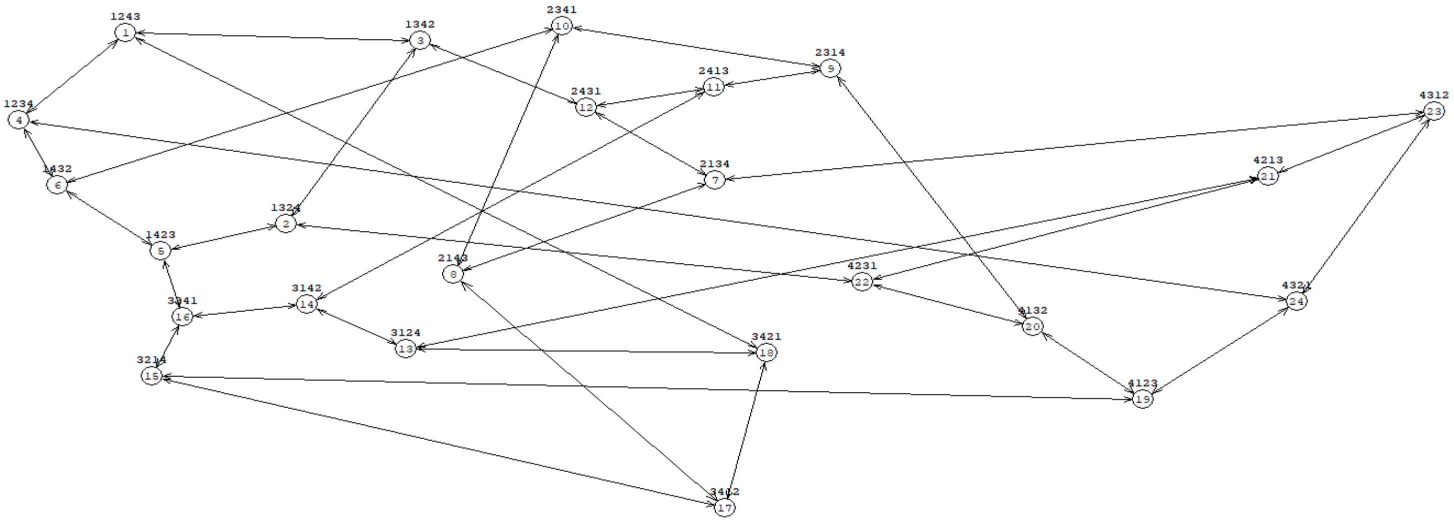
N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$g(N)$	1	4	6	8	10	12	14	15	17	18	19	21	22	23	24	26	28	29	**?**	**?**

Références :

Vidéo Marie Duflot-Kremer : https://www.youtube.com/watch?v=tI6uTAlX-_w&feature=youtu.be
http://www.irem.univ-bpclermont.fr/IMG/pdf/CrepierVersion2017-05-10.pdf
Concours Castor 2014 : https://concours.castor-informatique.fr/index.php
Lien vers une vidéo de l'algorithme présenté :
https://www.youtube.com/watch?v=kk-_DDgoXfk&feature=youtu.be
https://interstices.info/genese-dun-algorithme/

4) <u>Pour aller plus loin</u>

Le problème peut être représenté sous la forme d’un graphe.
Chaque nœud représentant une position du tas de crêpes et les arêtes le passage possible d’une position à une autre, voilà ce que cela donne pour 4 crêpes :



Graphe réalisé à l’aide de Grin4.0.

Résultat obtenu dans le recherche des plus courts chemins :

All Shortest Path from Source
Time : 00:15:45
Date : 27/06/2019
NetWork : NoName
Type : dirNet
Number of Points : 24
Number of Edges : 72

Source = 24
The Lengths of the Shortest Paths from 24 to other Points :

(1) 2	(2) 4	(3) 3
(4) 1	(5) 3	(6) 2
(7) 2	(8) 3	(9) 3
(10) 3	(11) 4	(12) 3
(13) 3	(14) 4	(15) 2
(16) 3	(17) 3	(18) 3
(19) 1	(20) 2	(21) 2
(22) 3	(23) 1	

Procedure complete.