



# UNIVERSITE DE KINSHASA

FACULTE POLYTECHNIQUE

TP D'ALGORITHME ET PROGRAMATION

Réaliser par :

- ASIMWE NDRUUDJO GLOIRE 2GC
- IMANI UKURANGO CHRISTIAN 2GC
- HESHIMA MASIMIKE DAVID 2GM

Année Académique 2021 – 2022

### **Question 1**

Qu'est-ce qu'une structure de données ?

**Réponse :** Une structure de données est un format spécial destiné à organiser, traiter, extraire et stocker des données

### **Question 2**

Enumérez les structures de données que nous avons abordé dans ce cours ?

**Réponse :**

Structure de donnée bas niveau :

- les représentations séquentielles
- les représentations chaînées

Structure de données haut niveau :

- les piles (stacks)
- les ensembles (sets)
- les listes
- les files d'attente (queues)
- les arbres (trees)

### **Question 3**

Quelle est la procédure de base de conception d'une structure de donnée ?

**Réponse :** Pour concevoir une base de données efficace et utile, vous devez suivre le bon processus, qui comprend les phases suivantes : Analyse des besoins, c'est-à-dire l'identification de l'objet de votre base de données. Organisation des données en tables. Spécification des clés primaires et analyse des relations

### **Question 4**

Qu'est-ce qu'une interface ?

**Réponse :** Une interface est un dispositif permettant la communication entre utilisateur et un système informatique. Dans domaine restreint, une interface de programmation (aussi nommée API pour Application Programming Interface) est un ensemble standardisé de méthodes, de classes, de fonctions et de constantes. Cela sert d'intermédiaire, par lequel un logiciel offre des prestations à d'autres logiciels

### **Question 5**

Qu'est-ce qu'un tableau ?

**Réponse :** Un tableau est **une liste ordonnée de n valeurs du même type**. On appelle n la taille du tableau, et les valeurs qu'ils contient sont ses éléments. Chaque élément est repéré dans le tableau par son indice, un nombre entier compris entre 0 et n -1 (inclus)

### **Question 6**

Qu'est-ce qu'un tableau de références ?

**Réponse :** Un tableau de référence est une structure de données permettant le stockage ou d'éléments de mêmes types, ces éléments peuvent être : des chaînes de caractères, des nombres, ... Ils sont disposés dans des cases numérotées (indices ou clés) ce qui permet d'y accéder en utilisant leur position dans le tableau. Il permet simplifier la manipulation des données et faciliter leurs traitements par les programmes informatiques.

### **Question 7**

Qu'est-ce qu'un tableau compact ?

**Réponse :** C'est un tableau qui stocke les bits qui représentent les données primaires (caractères, dans le cas des chaînes).

### **Question 8**

Qu'est qu'un tableau dynamique ?

**Réponse :** C'est un tableau dans lequel on peut modifier les variables.

Plus en détails :

La classe list de Python présente une abstraction plus intéressante. Bien qu'une liste ait une longueur particulière lorsqu'elle est construite, la classe nous sert à ajouter des éléments à la liste, sans limite apparente de la capacité globale de la liste. Pour fournir cette abstraction, Python s'appuie sur un tour de passe-passe algorithmique connu sous le nom de : tableau dynamique.

### **Question 9**

Comparer les tableaux compacts, les tableaux de références et les tableaux dynamiques. Utilisez une feuille Excel ou un tableau MS Word pour clarifier la comparaison.

**Réponse :**

Tableaux compacts	Tableaux de référence	Tableaux dynamiques
-Tableaux statiques, leur taille est définie à la compilation et ne peut être modifiée pendant l'exécution du programme.	- utiliser pour stocker des collections d'objet de taille variable, car les éléments peuvent être ajoutés ou supprimés en modifiant simplement les références vers les objets.	-peuvent être dimensionnés pendant l'exécution du programme.-utiles pour stocker des collections d'objet de taille variable lorsque la taille maximale n'est pas connue d'avance.
- chaque éléments est accessible en utilisant son index, qui est un entier positif.	- ajoutez ou supprimez en modifiant simplement les références vers les objets.	- généralement implémentés en utilisant des pointeurs ou des tableaux de référence.

Tableaux compacts	Tableaux de référence	Tableaux dynamiques
les tableaux compacts sont utilisés pour stocker des données de taille fixe	les tableaux de référence sont utilisés pour stocker des collections d'objet de taille variable en utilisant des adresses mémoire	les tableaux dynamiques sont utilisés pour stocker des collections d'objet de taille variable avec une taille maximale inconnue à l'avance et peuvent être dimensionnés pendant l'exécution du programme

### **Question 10**

Définir une pile (stack en anglais)

**Réponse** Une pile est un ensemble d'objets qui sont insérés et supprimés selon le principe du dernier entre, premier sorti (LIFO ; Last In, First Out). Un utilisateur peut insérer des objets dans une pile à n'importe quel moment, mais ne peut accéder ou supprimer que l'objet inséré en dernier qui reste au sommet de la pile.

### **Question 11**

Définir une file d'attente (queue en anglais)

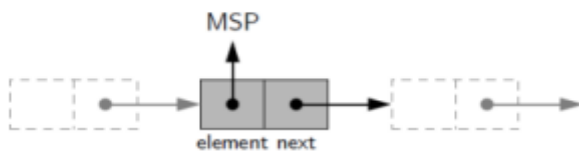
**Réponse :** Une autre structure de données fondamentale est la file d'attente. C'est un proche "cousin" de la pile, car une file d'attente est une collection d'objets qui sont insérés et supprimés selon le principe du premier entré, premier sorti (FIFO : First In, First Out). C'est-à-dire que des éléments peuvent être insérés à tout moment, mais seul l'élément qui a été dans la file d'attente le plus longtemps peut être ensuite supprimé. Nous disons

généralement que les éléments entrent dans une file d'attente par l'arrière et sont retirés par le devant.

### **Question 12**

Définir une liste chaînée simple, une liste chaînée circulaire, une liste doublement chaînée

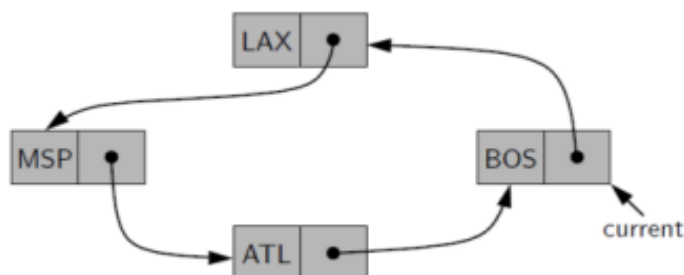
**Réponse :** liste chaînée simple : Une liste chaînée simple, dans sa forme la plus élémentaire, est un ensemble de nœuds qui, collectivement, forment une suite linéaire. Chaque nœud stocke une référence à un objet qui est un élément de la séquence, ainsi qu'une référence au nœud suivant de la liste



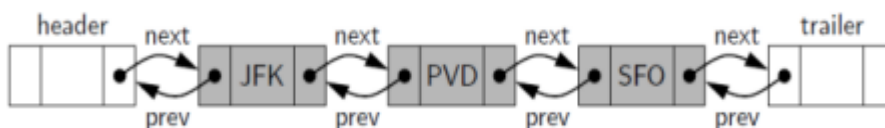
Exemple d'une instance nœud faisant partie d'une liste chaînée simple

Liste chaînée circulaire : Dans le cas des listes chaînées, nous pouvons faire en sorte que la queue de la liste utilise sa prochaine référence pour pointer vers la tête de la liste, comme le montre la figure ci basse Nous appelons une telle structure une liste chaînée circulaire.

Une liste chaînée circulaire fournit un modèle plus général qu'une liste chaînée simple. Il s'agit d'une liste dans laquelle les données sont chaînées de manière circulaire.



Liste doublement chaînée : Dans une liste chaînée simple, chaque nœud conserve une référence vers le nœud qui vient immédiatement après lui.



Une liste doublement chaînée contenant la séquence (JFK, PVD, SFO) et utilisant une sentinelle d'en tête (header) et une sentinelle de queue (trailer).

**Question 13**

Soit ci-dessous l'interface d'une pile (stack) S. Remplacez les points d'interrogation par le contenu approprié dans le tableau ci-dessous

opération	valeur retournée	contenu de la pile (stack)
s.push(5)	-	[5]
len(S)	1	????????????
S.push(10)	-	[5,10]
????????	-	[5,10,12]
S.pop()	12	[5,10]
S.is empty()	False	[5,10]
????????	10	[5]
S.pop()	5	[]
len(S)	0	[]
S.pop()	error	[]
S.push(2)	-	????????
S.push(8)	-	[2,8]
S.push(98)	-	[2,8,98]
????????	98	[2,8]

**Réponse :**

opération	valeur retournée	contenu de la pile (stack)
s.push(5)	-	[5]
len(S)	1	[5]
S.push(10)	-	[5,10]
S.push(12)	-	[5,10,12]
S.pop()	12	[5,10]
S.is empty()	False	[5,10]
S.pop()	10	[5]
S.pop()	5	[]
len(S)	0	[]
S.pop()	error	[]
S.push(2)	-	[2]
S.push(8)	-	[2,8]
S.push(98)	-	[2,8,98]
S.pop()	98	[2,8]

**Question 14**

Ci-dessous l'interface d'une file d'attente F :

opération	valeur retournée	contenu de la pile (stack)
f.enqueue(9)	-	[9]
F.dequeue()	9	?????
len(F)	0	[]
???????	-	[10]
F.first()	10	[10]
F.is empty()	False	[10]
???????	-	[10,5]
S.dequeue()	10	[5]
len(S)	1	[5]
F.enqueue(15)	-	[5,15]
F.first()	5	???????
F.dequeue()	5	[15]
F.is empty()	False	[15]
???????	98	[15,98]

**Réponse :**

opération	valeur retournée	contenu de la pile (stack)
f.enqueue(9)	-	[9]
F.dequeue()	9	[]
len(F)	0	[]
F.enqueue(10)	-	[10]
F.first()	10	[10]
F.is empty()	False	[10]
F.enqueue (5)	-	[10,5]
S.dequeue()	10	[5]
len(S)	1	[5]
F.enqueue(15)	-	[5,15]
F.first()	5	[5,15]
F.dequeue()	5	[15]
F.is empty()	False	[15]
F.enqueue(98)	98	[15,98]