

# INSTITUT SUPÉRIEUR D'ENSEIGNEMENT PROFESSIONNEL DE DIAMNIADIO FILIÈRE TIC

RESOLUTION D'ENONCE ALGORITMIQUE

2021-2022

Présenté par: Abdoulaye MBAYE

+221772389823

[ambaye@isepdiamniadio.edu.sn](mailto:ambaye@isepdiamniadio.edu.sn)

# PLAN

- **Qu'est-ce qu'un algorithme dans la résolution de problèmes?**
- **Stratégies de résolution de problèmes et obstacles**
- **Qu'est-ce qu'un heuristique et comment ça marche?**
- **Vue d'ensemble du processus mental de résolution de problèmes**



# QU'EST-CE QU'UN ALGORITHME DANS LA RÉOLUTION DE PROBLÈMES?

- Lorsque vous résolvez un problème , choisir la bonne approche est souvent la clé pour arriver à la meilleure solution.
- En psychologie, l'une de ces approches de résolution de problèmes est connue sous le nom d' **algorithme** .
- Pourquoi utiliser un algorithme?
  - L'avantage d'utiliser un algorithme pour résoudre un problème ou prendre une décision est de donner la meilleure réponse possible à chaque fois.
  - Ceci est utile dans les situations où la précision est critique ou lorsque des problèmes similaires doivent être fréquemment résolus.

# Qu'est-ce qu'un algorithme dans la résolution de problèmes?

- L'inconvénient de l'utilisation d'un algorithme pour résoudre le problème est que ce processus a tendance à prendre beaucoup de temps.
- Donc, si vous faites face à une situation où une décision doit être prise très rapidement, vous feriez mieux d'utiliser une autre stratégie de résolution de problèmes .





# Qu'est-ce qu'un algorithme dans la résolution de problèmes?

- **EXEMPLE:**

- Par exemple, un médecin qui prend une décision sur la façon de traiter un patient pourrait utiliser une approche algorithmique, mais cela prendrait beaucoup de temps et un traitement devrait être mis en œuvre rapidement. Dans ce cas, le médecin s'appuierait plutôt sur son expertise et ses expériences passées pour choisir très rapidement ce qu'il pense être la bonne approche de traitement.

# STRATÉGIES DE RÉOLUTION DE PROBLÈMES ET OBSTACLES





# Stratégies de résolution de problèmes et obstacles

- De l'organisation de votre collection de films à la décision d'acheter une maison, la résolution de problèmes constitue une grande partie de la vie quotidienne.
- Les problèmes peuvent aller de petit (résolution d'une seule équation mathématique sur votre devoir) à très grand (planification de votre future carrière).
- En psychologie cognitive , le terme « *résolution de problèmes* » fait référence au processus mental que les gens traversent pour découvrir, analyser et résoudre des problèmes.

# Stratégies de résolution de problèmes et obstacles

- Cela implique toutes les étapes du processus du problème, y compris:
  - la découverte du problème,
  - la décision de s'attaquer au problème,
  - la compréhension du problème,
  - la recherche des options disponibles et
  - la prise de mesures pour atteindre vos objectifs.
- Avant que la résolution de problèmes puisse se produire, il est important de comprendre d'abord la nature exacte du problème lui-même.
- Si votre compréhension du problème est erronée, vos tentatives pour la résoudre seront également incorrectes ou erronées.



# Stratégies de résolution de problèmes et obstacles

- Il y a un certain nombre de processus mentaux au travail pendant la résolution de problèmes. Ceux-ci inclus:
  - Perceptivement reconnaître un problème
  - Représenter le problème en mémoire
  - Considérant l'information pertinente qui s'applique au problème actuel
  - Identifier différents aspects du problème
  - Étiquetage et description du problème

# Stratégies de résolution de problèmes et obstacles

- Problèmes et obstacles dans la résolution de problèmes
  - La résolution de problèmes n'est pas un processus sans faille. Il existe un certain nombre d'obstacles qui peuvent interférer avec notre capacité à résoudre un problème rapidement et efficacement.
    - **Fixité fonctionnelle** : Ce terme fait référence à la tendance à ne voir les problèmes que de manière habituelle.
    - **Informations non pertinentes ou trompeuses**: Lorsque vous tentez de résoudre un problème, il est important de faire la distinction entre les informations pertinentes pour le problème et les données non pertinentes pouvant conduire à des solutions erronées.
    - **Hypothèses**: Lorsqu'ils traitent un problème, les gens font souvent des hypothèses sur les contraintes et les obstacles qui empêchent certaines solutions.
    - **Mental Set**: Un autre obstacle commun à la résolution de problèmes est connu comme un ensemble mental, qui est la tendance que les gens ont à utiliser uniquement des solutions qui ont fonctionné dans le passé plutôt que de chercher des idées alternatives.



# QU'EST-CE QU'UN HEURISTIQUE ET COMMENT ÇA MARCHE?

- Une heuristique est un raccourci mental qui permet aux gens de résoudre des problèmes et de porter des jugements rapidement et efficacement.
- Ces stratégies empiriques raccourcissent le temps de prise de décision et permettent aux gens de fonctionner sans s'arrêter constamment pour réfléchir à leur prochaine action.
- Heuristiques sont utiles dans de nombreuses situations, mais ils peuvent également conduire à des biais cognitifs .

# QU'EST-CE QU'UN HEURISTIQUE ET COMMENT ÇA MARCHE?

- Pourquoi utilisons-nous l'heuristique?
  - L'heuristique joue un rôle important dans la résolution de problèmes et la prise de décision .
  - Lorsque nous essayons de résoudre un problème ou de prendre une décision, nous nous tournons souvent vers ces raccourcis mentaux lorsque nous avons besoin d'une solution rapide.
  - Le monde est plein d'informations, mais nos cerveaux sont seulement capables de traiter un certain montant. Si vous essayiez d'analyser chaque aspect de chaque situation ou décision, vous n'obtiendriez jamais rien!



# QU'EST-CE QU'UN HEURISTIQUE ET COMMENT ÇA MARCHE?

- Afin de faire face à l'énorme quantité d'informations que nous rencontrons et d'accélérer le processus de prise de décision, le cerveau s'appuie sur ces stratégies mentales pour simplifier les choses afin que nous n'ayons pas à passer du temps à analyser chaque détail.
- Vous prenez probablement des centaines, voire des milliers de décisions chaque jour.
  - Que devriez-vous avoir pour le petit-déjeuner?
  - Que devriez-vous porter aujourd'hui?
  - Devriez-vous conduire ou prendre le bus?
  - Devriez-vous aller boire un verre plus tard avec vos collègues?

# QU'EST-CE QU'UN HEURISTIQUE ET COMMENT ÇA MARCHE?

- Les heuristiques vous permettent de prendre de telles décisions avec une relative facilité sans beaucoup d'agonie.
- Par exemple, lorsque vous essayez de décider si vous devez conduire ou prendre l'autobus pour vous rendre au travail, vous pourriez soudainement vous rappeler qu'il y a des travaux routiers le long de la route d'autobus standard.



# QU'EST-CE QU'UN HEURISTIQUE ET COMMENT ÇA MARCHE?

- L'heuristique aide à rendre la vie plus facile et nous permet de prendre des décisions rapides qui sont généralement assez précises.

# VUE D'ENSEMBLE DU PROCESSUS MENTAL DE RÉOLUTION DE PROBLÈMES





# VUE D'ENSEMBLE DU PROCESSUS MENTAL DE RÉOLUTION DE PROBLÈMES

- La résolution de problèmes est un processus mental qui implique la découverte, l'analyse et la résolution de problèmes.
- Le but ultime de la résolution de problèmes est de surmonter les obstacles et de trouver la solution qui résout le mieux le problème.
- La meilleure stratégie pour résoudre un problème dépend largement de la situation unique.
- Dans certains cas, les gens ont intérêt à apprendre tout ce qu'ils peuvent sur le problème et à utiliser ensuite des connaissances factuelles pour trouver une solution.
- Dans d'autres cas, la créativité et la perspicacité sont les meilleures options.

# VUE D'ENSEMBLE DU PROCESSUS MENTAL DE RÉOLUTION DE PROBLÈMES

- **Les étapes de la résolution de problèmes:**

- Afin de résoudre correctement un problème, il est important de suivre une série d'étapes.

1. **Identifier le problème:** Bien que cela puisse sembler une étape évidente, l'identification du problème n'est pas toujours aussi simple que cela puisse paraître.
2. **Définition du problème:** Une fois le problème identifié, il est important de définir complètement le problème afin qu'il puisse être résolu.

- **Former une stratégie:** L'étape suivante consiste à élaborer une stratégie pour résoudre le problème. L'approche utilisée varie en fonction de la situation et des préférences uniques de l'individu.

1. **Organisation de l'information:** Avant de trouver une solution, nous devons d'abord organiser les informations disponibles.

1. Que savons-nous du problème?

2. Que ne savons-nous *pas* ?

Plus nous aurons d'informations disponibles, mieux nous serons en mesure de trouver une solution précise.



# VUE D'ENSEMBLE DU PROCESSUS MENTAL DE RÉOLUTION DE PROBLÈMES

1. **Allouer des ressources:** Bien sûr, nous n'avons pas toujours de l'argent, du temps et d'autres ressources illimitées pour résoudre un problème. Avant de commencer à résoudre un problème, vous devez déterminer sa priorité.
2. **Suivi des progrès:** Les résolveurs de problèmes efficaces ont tendance à surveiller leurs progrès lorsqu'ils travaillent à trouver une solution. S'ils ne font pas de bons progrès pour atteindre leur objectif, ils réévalueront leur approche ou chercheront de nouvelles stratégies .
3. **Évaluation des résultats:** Une fois qu'une solution a été trouvée, il est important d'évaluer les résultats pour déterminer si c'est la meilleure solution possible au problème.

# EXEMPLE

## 1.1.1 Exemple 1 : permutation de deux voitures

**Tâche :** *Permuter deux voitures sur un parking de trois places numérotées de  $P_1$  à  $P_3$  et ceci sans gêner la circulation. La voiture B, est sur l'emplacement  $P_2$ , la voiture R, est sur l'emplacement  $P_3$ .*

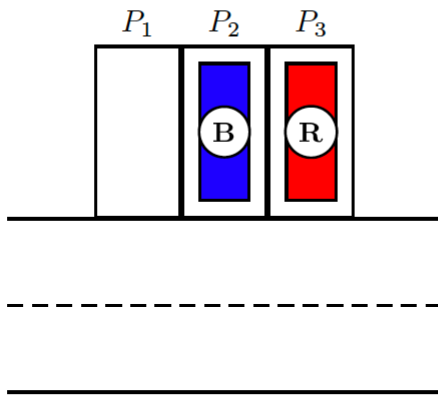


Figure 1.1: Avant permutation

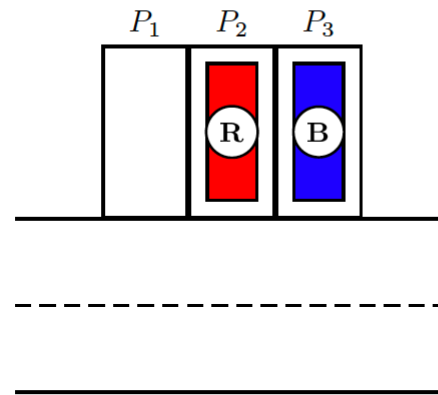


Figure 1.2: Après permutation

Voici, l'«algorithme» basique permettant de réaliser cette tâche :

- 1: Déplacer la voiture R de l'emplacement  $P_3$  à  $P_1$ .
- 2: Déplacer la voiture B de l'emplacement  $P_2$  à  $P_3$ .
- 3: Déplacer la voiture R de l'emplacement  $P_1$  à  $P_2$ .



# EXEMPLE

Malheureusement, cet algorithme souffre de nombreuses lacunes et ambiguïtés :

- Au départ, l'emplacement  $P_1$  est-il libre?
- Que veut-dire l'action **déplacer**? Si les voitures sont non-roulantes, prendre une grue? sinon a-t'on les clés des deux véhicules?
- Dans l'énoncé, il est dit «sans gêner la circulation»! Donc il y a de la circulation et à tout moment un véhicule extérieur au problème peut venir occuper un emplacement libre!
- ...

En fait, les lacunes et ambiguïtés proviennent pour la plupart d'une mauvaise description de la tâche à réaliser : le problème est mal posé.

Cependant nous pouvons tout de même écrire un algorithme mais en restreignant son champ d'application : nous allons faire des **hypothèses** sur les données du problème.

Tout d'abord il faut préciser les **données** du problème :

## **Données :**

- | Parking de trois emplacements, numérotés de  $P_1$  à  $P_3$ .
- | Deux voitures, l'une notée B et l'autre R.

Ensuite, nous précisons les **hypothèses** sur les données

## **Hypothèses sur les données :**

- La voiture B est sur l'emplacement  $P_2$ .
- La voiture R est sur l'emplacement  $P_3$ .
- Les deux voitures sont «roulantes».

# EXEMPLE

Ensuite, nous donnons des hypothèses plus générales

## Hypothèses générales :

1. Une seule personne réalise la tâche.
2. Celle-ci a son permis de conduire et les clés des deux voitures.
3. Lors du déplacement d'un véhicule d'un emplacement à un autre aucune voiture extérieure ne vient sur le parking.
4. Une voiture extérieure ne reste qu'un temps fini sur un emplacement

Nous considérons que la phase de déplacement d'un véhicule d'un emplacement à un autre (libre) ne pose aucun problème au titulaire d'un permis! Et enfin, nous précisons le résultat voulu :

## Résultats :

1. La voiture B est sur l'emplacement  $P_3$ .
2. La voiture R est sur l'emplacement  $P_2$ .

Nous obtenons alors l'algorithme suivant :

- 1: Aller au volant du véhicule R (emplacement  $P_3$ ).
- 2: **Tantque** emplacement  $P_1$  est occupé **faire**
- 3:     Attendre.
- 4: **Fin**
- 5: Déplacer le véhicule R de l'emplacement  $P_3$  à  $P_1$
- 6: Aller au volant du véhicule B (emplacement  $P_2$ ).
- 7: **Tantque** emplacement  $P_3$  est occupé **faire**
- 8:     Attendre.
- 9: **Fin**
- 10: Déplacer le véhicule B de l'emplacement  $P_2$  à  $P_3$
- 11: Aller au volant du véhicule R (emplacement  $P_1$ ).
- 12: **Tantque** emplacement  $P_2$  est occupé **faire**
- 13:     Attendre.
- 14: **Fin**
- 15: Déplacer le véhicule R de l'emplacement  $P_1$  à  $P_2$



# EXEMPLE

## 1.1.2 Exemple 2 : résolution d'une équation

**Tâche :** Résoudre l'équation  $ax = b$ .

Voici, l'«algorithme» basique permettant de réaliser cette tâche :

```
1: Si  $a$  différent de 0 alors
2:   La solution est  $x = b/a$ .
3: Sinon
4:   Il n'y a pas de solution
5: Fin
```

Une nouvelle fois, l'énoncé de la tâche à effectuer est trop imprécise! En effet, rien ne nous permet d'affirmer que  $x$  est l'inconnue ou que  $a$  n'est pas une matrice. Il faut alors palier au manque d'informations en ajoutant des hypothèses pour **clarifier le problème** :

Soient  $a \in \mathbb{R}^*$  et  $b \in \mathbb{R}$  donnés. Le problème est

trouver  $x \in \mathbb{R}$  tel que  
 $ax = b$ .

Ce problème est bien posé : sa résolution ne souffre d'aucune ambiguïté.

---

**Algorithme 1.1** Résolution de l'équation du premier degré  $ax = b$ .

---

**Données :**  $a$  : un réel non nul,  
 $b$  : un réel.

**Résultat :**  $x$  : un réel.

1:  $x \leftarrow b/a$

---

### 1.1.3 caractéristiques

Voici en résumé les caractéristiques d'un *bon* algorithme :

- Il ne souffre d'aucune ambiguïté  $\Rightarrow$  très clair.
- Combinaison d'opérations (actions) élémentaires.
- Pour toutes les données d'entrée, l'algorithme doit fournir un résultat en un nombre fini d'opérations.
- Il est adapté au public auquel il est destiné.

### 1.1.4 Première approche méthodologique

Nous présentons ci-dessous quelques éléments méthodologiques pour la réalisation d'un algorithme :

**Etape 1 :** Définir clairement le problème.

**Etape 2 :** Rechercher une méthode de résolution (formules, ...)

**Etape 3 :** Ecrire l'algorithme (par raffinement successif pour des algorithmes *compliqués*).



## 1.4 Méthodologie

### 1.4.1 Description du problème

- Spécification d'un ensemble de données  
Origine : énoncé, hypothèses, sources externes, ...
- Spécification d'un ensemble de buts à atteindre  
Origine : résultats, opérations à effectuer, ...
- Spécification des contraintes

### 1.4.2 Recherche d'une méthode de résolution

- Clarifier l'énoncé.
- Simplifier le problème.
- Ne pas chercher à le traiter directement dans sa globalité.
- S'assurer que le problème est soluble (sinon problème d'indécidabilité!)
- Recherche d'une stratégie de construction de l'algorithme
- Décomposer le problème en sous problèmes partiels plus simples : raffinement.
- Effectuer des raffinements successifs.
- Le niveau de raffinement le plus élémentaire est celui des instructions.

- Que doit-il faire ? (i.e. Quel problème est-il censé résoudre?)
- Quelles sont les données nécessaires à la résolution de ce problème?
- Comment résoudre ce problème «à la main» (sur papier)?

Si l'on ne sait pas répondre à l'une de ces questions, l'écriture de l'algorithme est fortement comprise.

### Exemple : algorithme pour une somme



#### Exercice 1.4.1

Ecrire un algorithme permettant de calculer

$$S(x) = \sum_{k=1}^n k \sin(2kx)$$

**Correction** L'énoncé de cet exercice est imprécis. On choisit alors  $x \in \mathbb{R}$  et  $n \in \mathbb{N}$  pour rendre possible le calcul. Le problème est donc de calculer

$$\sum_{k=1}^n k \sin(2kx).$$

Toutefois, on aurait pu choisir  $x \in \mathbb{C}$  ou encore un tout autre problème :

$$\text{Trouver } x \in \mathbb{R} \text{ tel que } S(x) = \sum_{k=1}^n k \sin(2kx)$$



où  $n \in \mathbb{N}$  et  $S$ , fonction de  $\mathbb{R}$  à valeurs réelles, sont les données!

---

**Algorithme 1.7** Calcul de  $S = \sum_{k=1}^n k \sin(2kx)$

---

**Données :**  $x$  : nombre réel,  
 $n$  : nombre entier.

**Résultat :**  $S$  : un réel.

1:  $S \leftarrow 0$   
2: **Pour**  $k \leftarrow 1$  à  $n$  **faire**  
3:    $S \leftarrow S + k * \sin(2 * k * x)$   
4: **Fin**

---



**Exemple : algorithme pour un produit**



#### **Exercice 1.4.2**

Ecrire un algorithme permettant de calculer

$$P(z) = \prod_{n=1}^k \sin(2kz/n)^k$$

**Correction** L'énoncé de cet exercice est imprécis. On choisit alors  $z \in \mathbb{R}$  et  $k \in \mathbb{N}$  pour rendre possible le calcul.

**Algorithme 1.8** Calcul de  $P = \prod_{n=1}^k \sin(2kz/n)^k$

**Données :**  $z$  : nombre réel,  
 $k$  : nombre entier.

**Résultat :**  $P$  : un réel.

- 1:  $P \leftarrow 1$
- 2: **Pour**  $n \leftarrow 1$  à  $k$  **faire**
- 3:    $P \leftarrow P * \sin(2 * k * z/n)^k$
- 4: **Fin**




## 1.5 Principes de «bonne» programmation pour attaquer de «gros» problèmes

Tous les exemples vus sont assez courts. Cependant, il peut arriver que l'on ait des programmes plus longs à écrire (milliers de lignes, voir des dizaines de milliers de lignes). Dans l'industrie, il arrive que des équipes produisent des codes de millions de lignes, dont certains mettent en jeu des vies humaines (contrôler un avion de ligne, une centrale nucléaire, ...). Le problème est évidemment d'écrire des programmes sûrs. Or, *un programme à 100% sûr, cela n'existe pas!* Cependant, plus un programme est simple, moins le risque d'erreur est grand : c'est sur cette remarque de bon sens que se basent les «bonnes» méthodes. Ainsi :

*Tout problème compliqué doit être découpé en sous-problèmes plus simples*

Il s'agit, lorsqu'on a un problème  $P$  à résoudre, de l'analyser et de le décomposer en un ensemble de problèmes  $P_1, P_2, P_3, \dots$  plus simples. Puis,  $P_1$ , est lui-même analysé et décomposé en  $P_{11}, P_{12}, \dots$ , et  $P_2$  en  $P_{21}, P_{22}$ , etc. On poursuit cette analyse jusqu'à ce qu'on n'ait plus que des problèmes élémentaires à résoudre. Chacun de ces problèmes élémentaires est donc traité séparément dans un module, c'est à dire un morceau de programme relativement indépendant du reste. Chaque module sera *testé et validé* séparément dans la mesure du possible et naturellement *largement documenté*. Enfin ces modules élémentaires sont assemblés en modules de plus en plus complexes, jusqu'à remonter au problème initiale. A chaque niveau, il sera important de bien réaliser les phases de test, validation et documentation des modules.



Par la suite, on s'évertue à écrire des algorithmes!

**Ceux-ci ne seront pas optimisés<sup>a</sup>!**

---

<sup>a</sup>améliorés pour minimiser le nombre d'opérations élémentaires, l'occupation mémoire, ..