



**HAL**  
open science

## Apprendre à penser les algorithmes

Christian Blanvillain

► **To cite this version:**

Christian Blanvillain. Apprendre à penser les algorithmes. Atelier “ Apprendre la Pensée Informatique de la Maternelle à l’Université ”, dans le cadre de la conférence Environnements Informatiques pour l’Apprentissage Humain (EIAH), 2021, Fribourg, Suisse. pp.1-12. hal-03241685

**HAL Id: hal-03241685**

**<https://hal.archives-ouvertes.fr/hal-03241685>**

Submitted on 28 May 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Apprendre à penser les algorithmes

Christian Blanvillain

HEP Lausanne, Suisse & Université de Patras, Grèce

christian.blanvillain@hepl.ch

## RÉSUMÉ

---

Nous présentons un algorithme de résolution de problèmes algorithmiques. Cet algorithme est utile aux enseignants qui souhaitent faire apprendre aux élèves comment trouver des solutions à des problèmes algorithmiques. Nous proposons un schéma des processus cognitifs mobilisés dans l'acte de conception d'algorithmes. C'est un outil d'analyse pragmatique susceptible d'aider les enseignants à identifier les éventuelles lacunes cognitives des élèves en difficulté. Il est accompagné d'une liste de stratégies utiles aux élèves. Nous concluons avec une réflexion sur les facteurs pouvant influencer l'aptitude à concevoir des algorithmes.

## ABSTRACT

---

### **How to think algorithms.**

We present an algorithm for solving algorithmic problems. This algorithm is useful for teachers who wish to teach students how to find solutions to algorithmic problems. We propose a schema of the cognitive processes mobilized in the act of designing algorithms. It is a pragmatic analysis tool that can help teachers to identify possible cognitive deficiencies of students in difficulty. It is accompanied by a list of useful strategies for students. We conclude with a discussion of factors that can influence the ability to design algorithms.

---

**MOTS-CLÉS :** Intelligence algorithmique, Fonctions cognitives, Stratégies cognitives, Noésologie.

**KEYWORDS:** Algorithmic intelligence, Cognitive functions, Cognitive strategies, Noesiology.

---

## 1 Introduction

Lorsque l'on souhaite enseigner l'informatique aux jeunes élèves, plusieurs questions se posent : peut-on se contenter de développer leur pensée informatique ou bien faut-il leur apprendre à programmer ? Finalement, l'essentiel n'est-il pas de leur apprendre à penser les algorithmes ? Développer la pensée informatique sans leur apprendre à penser les algorithmes, c'est passer à côté d'une opportunité extraordinaire de leur enseigner des stratégies cognitives pour la résolution de problèmes en cultivant un état d'esprit rigoureux et méthodique, qui leur serviront bien au-delà des exercices proposés et tout particulièrement dans les situations scolaires ou extrascolaires où une approche intuitive ne suffit plus pour venir à bout des obstacles rencontrés. Leur apprendre à développer leur intelligence algorithmique dès leur plus jeune âge, c'est leur apprendre à développer leur intelligence créative, pratique et logico-mathématique (en référence à la conception triarchique de l'intelligence de Robert J. Sternberg (1985)). Les transferts possibles dans les autres disciplines scolaires et dans la vie quotidienne sont nombreux. Nous les préparerons ainsi à mieux appréhender le monde numérique dans lequel ils évoluent quotidiennement.

## 2 Algorithme de résolution de problèmes algorithmiques

Dans ce chapitre, nous décrivons un algorithme présentant comment l'élève pourrait s'y prendre pour résoudre des problèmes algorithmiques. Cette théorie est issue d'une analyse introspective de notre propre processus de résolution de problème algorithmique, ainsi que de nombreuses lectures :

- Jean-Yves Fournier, *À l'école de l'intelligence* (1999, 2e partie) ;
- Pierre Vianin, *L'aide stratégique aux élèves en difficulté scolaire* (2009, p. 72 à 116) ;
- Todd Lubart, Christophe Mouchiroud, Sylvie Tordjman, Franck Zenasni, *Psychologie de la créativité* (2015, p. 117 et 122) ;
- Patrick Lemaire et André Didierjean, *Introduction à la psychologie cognitive* (2018, chap. 8).

### 2.1 Lecture et compréhension de l'énoncé

Pour résoudre un problème algorithmique, on commence par lire son énoncé. En première lecture, on peut facilement saturer notre mémoire de travail et ne pas comprendre le texte, ce n'est pas grave. Il ne faut pas paniquer et garder confiance en soi. Reprendre la lecture en essayant d'éliminer l'inutile et conserver les informations importantes pour identifier ce qui est demandé de faire ou de résoudre. Trier les informations fournies dans les consignes en cherchant les données de départ et les choses à produire [**Récolte**]. Cette étape est parfois nécessaire pour pouvoir être plus sélectif sur les informations pertinentes pour se représenter le travail à réaliser et les distinguer des informations qui le sont moins, qui posent juste le contexte. L'objectif est de ne conserver en mémoire de travail que les informations nécessaires pour que l'on puisse se créer une représentation mentale du problème à résoudre [**Assemblage**] en écartant les informations parasites. Il faut réussir à se représenter le problème à résoudre dans son ensemble : ce que l'on sait, ce que l'on cherche. Il s'agit de faire une synthèse de ce qui est demandé afin de pouvoir émettre une hypothèse heuristique sur la stratégie à utiliser et sur la démarche qui conduira à la solution [**Projection**]. Stratégie qu'il va falloir mentalement tester en une approche exploratoire rapide sur des données simples, pour voir si elle est efficiente ou pas. L'exploitation de la représentation mentale du problème à résoudre que l'on s'est créée en lisant l'énoncé va permettre d'explorer plusieurs possibilités en éliminant rapidement les mauvaises idées, juste en testant les situations triviales.

Parmi les hypothèses faites sur les différentes stratégies de résolution possibles, conserver celle qui nous semble être la plus simple à mettre en œuvre, même si ce n'est pas forcément la plus élégante. Attention cependant à ne pas se précipiter sur la première idée qui vient à l'esprit ni se lancer dans l'élaboration d'une réponse sans avoir pris le temps de vérifier si l'idée semble valable ou pas. Tester cette hypothèse : appliquer le plan d'action prévu sur les données mémorisées, toujours en commençant par les situations les plus simples, car on cherche encore à valider notre hypothèse heuristique de résolution. Si notre hypothèse est correcte, on a un début de solution. Continuer alors d'analyser les autres données pour résoudre la suite du problème. Si notre hypothèse bloque sur certaines données et ne permet plus d'obtenir la réponse à la question posée, revoir l'hypothèse de départ et envisager d'utiliser une stratégie moins simple, mais plus précise. Un changement d'hypothèse peut aussi avoir lieu en cours de résolution : l'hypothèse initiale peut tout à fait fonctionner pour les cas simples, mais pas pour tous les cas. Les résultats obtenus ne sont pas forcément tous à remettre en cause : il ne faut pas obligatoirement jeter les réflexions faites et les conclusions partielles auxquelles on est arrivé. Une manière de chercher de nouvelles hypothèses est de faire des liens avec ce que l'on connaît en cherchant des similitudes avec d'autres situations déjà

rencontrées. Observer les ressemblances et les différences avec ce que l'on a déjà vécu par le passé peut nous aider à penser à une nouvelle manière de faire.

Lorsque l'on réussit à produire le résultat escompté dans les situations triviales, vérifier que la solution trouvée est consistante avec l'énoncé du problème. La compréhension partielle du problème va nous aider à mieux cerner ce qui est demandé et mieux comprendre ce qu'il faut faire pour pouvoir gérer les situations non triviales. Les réflexions initiales ne sont donc pas perdues. Elles étaient nécessaires pour arriver à ce stade du raisonnement. Elles nous permettent maintenant d'affiner nos réponses, de pouvoir reprendre l'ensemble de notre approche avec une conscience plus globale de ce qu'il faut faire en prenant en considération toutes les données du problème.

Le processus peut sembler coûteux en temps, mais il n'en est rien. Chaque étape permet de se familiariser davantage avec le problème et de mieux appréhender les difficultés, les choses à faire et à ne pas faire, les informations essentielles et les informations moins cruciales. Cette expertise qui se construit en nous va nous permettre de nous approcher, mentalement et de manière itérative vers ce qui est attendu. Ce chemin intellectuel de compréhension est nécessaire pour se construire une représentation mentale fidèle du problème à résoudre. Parfois, le problème est suffisamment simple pour qu'il n'y ait pas besoin de faire de démarche particulière : on comprend tout de suite ce qui est demandé et ce qu'il faut faire. Mais parfois, le véritable problème n'apparaît pas tout de suite et se découvre au fil du processus de lecture et de tentative de résolution. L'établissement d'une stratégie de résolution s'effectue conjointement au processus de compréhension du problème. Si l'on ne réussit pas à trouver une stratégie de résolution satisfaisante, c'est peut-être que l'on n'a pas assez bien exploré les données du problème. Une boucle se crée alors, entre la relecture pour une meilleure compréhension du problème, la recherche d'une nouvelle hypothèse heuristique et la vérification de l'hypothèse retenue. Nous appelons cette boucle le cycle d'élaboration d'une image mentale du problème (figure 1).

## 2.2 Résolution du problème

La phase de résolution commence lorsque l'on met en œuvre la stratégie de résolution imaginée sur l'ensemble des données du problème en envisageant les cas plus complexes. Concrètement, nous allons manipuler (ou plutôt *mentipuler* ?) les informations disponibles (données du problème) pour déplacer avec les yeux les données en faisant attention à ne pas oublier les nouveaux états du système que nous sommes en train de produire au travers de nos manipulations mentales [**Réflexion**].

À ce stade de la réflexion, il est très important de rester concentré sur la tâche : écarter tout événement extérieur perturbateur, pour ne pas altérer notre état mental et la représentation que nous nous faisons du système. Notre mémoire de travail est fragile et éphémère. Si une perturbation que nous ne réussissons pas éviter à lieu (quelqu'un qui nous pose directement une question par exemple), ne pas hésiter à abandonner le raisonnement dans son ensemble et à recommencer depuis le début. Le nouvel état mental du système doit être fiable et dénué de toute erreur pour que nous puissions l'utiliser dans la suite du raisonnement, pour calculer les prochains états du système de manière itérative jusqu'à ce que nous atteignons la fin de notre algorithme de résolution associé à notre stratégie et à nos hypothèses. Ces opérations mentales sont nécessaires et importantes pour que nous puissions nous assurer d'avancer, dans le raisonnement, vers une solution au problème posé. De temps en temps, si l'image mentale devient floue ou imprécise, reprendre rapidement les dernières étapes pour rafraîchir notre mémoire de travail et éliminer les risques d'erreur de raisonnement dus aux imprécisions inhérentes à notre mémoire de travail.

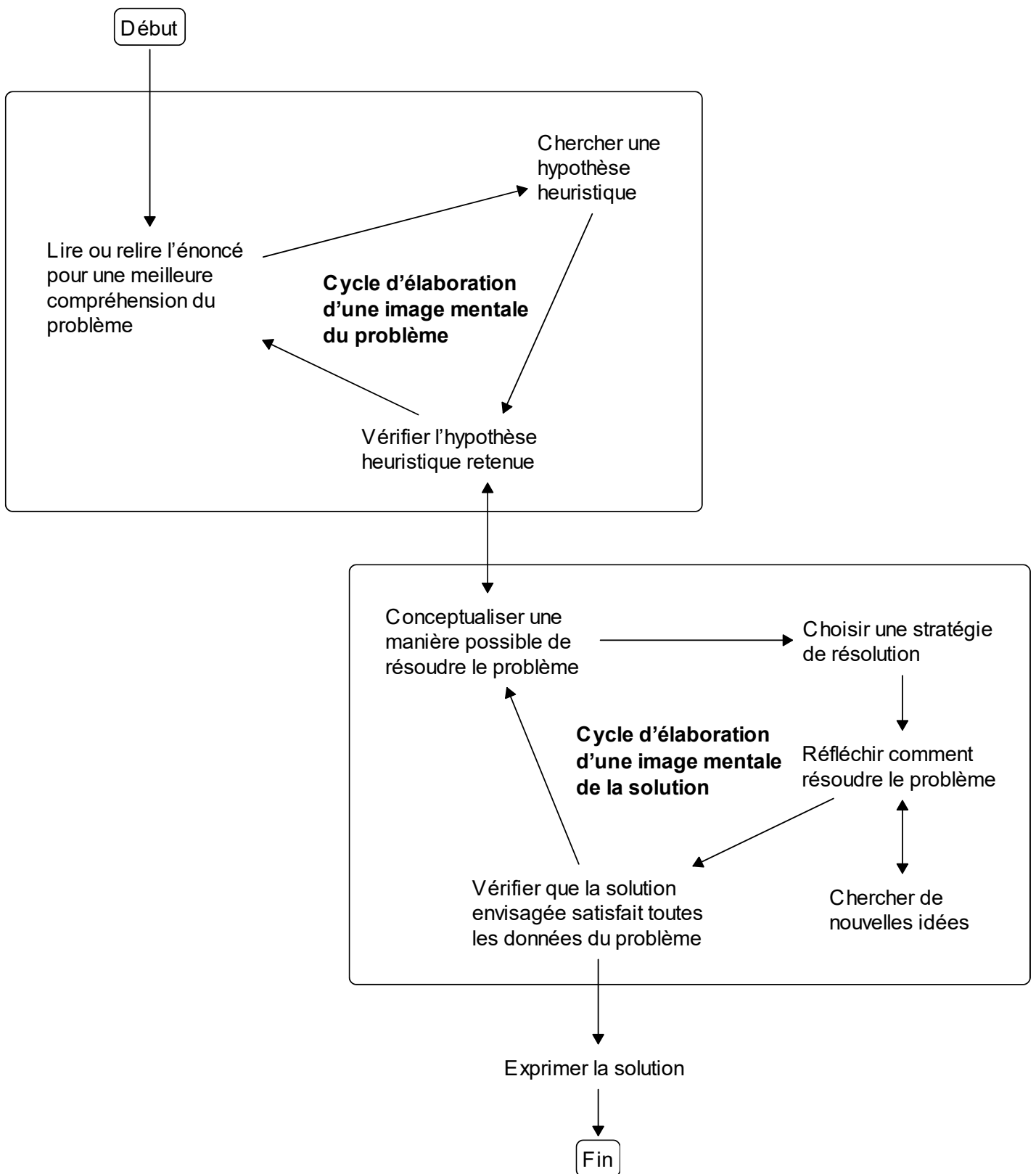


FIGURE 1 – Les cycles d’élaboration des images mentales

Concrètement, réfléchir au code à écrire ou bien aux opérations à décrire pour élaborer un algorithme solution au problème posé, consiste à visualiser dans son esprit les différents états du modèle mental du système en cours de conception, puis de décrire les opérations qui vont faire évoluer cet état pour obtenir un état objectif qui est soit l'objectif final, soit une étape intermédiaire que l'on imagine nous rapprocher de l'objectif final. La représentation des états internes du modèle du système que nous programmons mentalement se fait en utilisant une image de l'algorithme en cours de construction. Cette image permet de visualiser les valeurs et leurs évolutions. Nous pouvons nous représenter les différentes opérations à effectuer comme si c'était nous qui devons physiquement réaliser ces opérations. En nous projetant à la place du système que nous sommes en train de programmer, nous mobilisons notre intelligence pratique : il suffit de faire appel au bon sens et de réfléchir à ce que nous devrions faire et pouvons faire pour obtenir ce que l'on souhaite, puis une fois que ces opérations sont conscientisées, de les décrire en langage naturel.

Souvent, une bonne stratégie de résolution consiste à découper le problème en étapes plus simples à résoudre et se concentrer sur chacune des étapes pour trouver une solution sans se préoccuper pour le moment des autres étapes. Mais lorsque l'on ne sait plus quoi faire et que l'on est bloqué, il faut observer avec un peu de recul l'état global du système dans notre tête, vérifier que nous avons bien exploité toutes les données à disposition, que nous avons bien exploré tout ce qu'il est possible de faire avec ces données, y compris les configurations de données improbables lorsque le problème est corsé. On peut donc faire un inventaire des données déjà utilisées et traitées, des résultats de réflexion intermédiaires obtenus et des données qu'il reste à traiter pour atteindre l'objectif qui nous est demandé. Si l'on ne sait vraiment plus quoi faire après avoir fait cet inventaire, on peut repasser en revue toutes les informations de l'énoncé du problème au lieu de se contenter de parcourir uniquement les données conservées dans notre mémoire de travail. La recherche d'une éventuelle information oubliée dans l'énoncé risque de nuire à notre représentation avancée de l'état du système et parfois, cet exercice nous obligera à reprendre notre raisonnement depuis le début pour intégrer la nouvelle information glanée dans l'énoncé. Cette gymnastique n'est pas aussi coûteuse en temps que ce que l'on pourrait croire, car refaire mentalement le chemin que nous venons juste de parcourir est assez rapide. Notre mémoire de travail va nous aider à aller bien plus vite la deuxième fois.

Si l'on est toujours bloqué dans l'étape de résolution du problème, avant d'abandonner notre hypothèse de travail, de tout jeter et de repartir sur une autre hypothèse plus complexe pour reprendre nos réflexions à partir de zéro, il convient de faire une pause. Relâcher la tension intellectuelle intérieure, calmer son esprit, laisser le cerveau se reposer quelques secondes en ne pensant à rien : une nouvelle idée pourrait alors jaillir spontanément nous permettant de nous débloquent [Idée]. À ce stade du raisonnement, il est hyper important de ne pas se laisser distraire par les événements extérieurs, ou par nos émotions intérieures : stress, doute, peur de ne pas y arriver, peur de se tromper, peur d'être en échec, autant d'émotions qui viennent perturber notre tranquillité et notre raisonnement. Toutes ces émotions doivent être mises en sourdine, pour laisser au cerveau la possibilité d'activer sa pensée créatrice et nous montrer une alternative que nous n'avions pas jusqu'alors envisagée. Mais pour que la magie opère, il faut accepter de ne pas savoir quoi faire, de ne pas avoir de réponse, d'être perdu et rempli de doute, sans sourciller, sans paniquer, sans arrêter de faire l'effort de chercher à faire des associations d'idées pour en générer de nouvelles ou bien de ne penser à rien pour laisser émerger spontanément de nouvelles idées.

Si toujours rien ne se passe, nous pouvons passer en revue des techniques de résolution connues, vues précédemment en nous demandant comment faire autrement, en tentant résoudre le problème en lui soumettant des principes déjà vus, des approches apprises, jusqu'à ce qu'une piste de solution apparaisse et que nous puissions repartir dans une nouvelle hypothèse de résolution, avec une stratégie

adaptée. Nous pouvons également, à ce stade de la réflexion, nous aider de l'intelligence du groupe et demander aux autres élèves de partager leurs idées avec nous. Une idée fait émerger d'autres idées. Parfois, le chemin qu'emprunte la solution peut être assez détourné et c'est une idée à priori anodine qui pourrait nous mettre sur la voie d'une solution possible.

## 2.3 Expression de la solution

Lorsque l'on a terminé le processus de réflexion, que l'on a construit une solution à proposer, il faut vérifier qu'elle est bien juste [**Vérification**]. C'est une vérification plus lente qui s'opère par rapport à la vérification rapide de l'heuristique initiale, car ici on travaille avec une conscience complète de toutes les données du problème. Si la solution envisagée ne satisfait pas toutes les données du problème, chercher une nouvelle manière possible de résoudre le problème, adapter la stratégie de résolution et reprendre nos réflexions. Nous appelons cette boucle le cycle d'élaboration d'une image mentale de la solution (figure 1).

On peut être amené à élaborer des explications plus ou moins complexes pour présenter nos idées et notre résultat [**Expression**] à d'autres personnes. Lorsque l'on exprime notre solution, bien s'assurer que nos idées soient structurées, concises et compréhensibles par nos interlocuteurs. Enfin, lorsque tout est fini, repenser à ce qui a été réalisé. Se remémorer la stratégie qui nous a conduits à la résolution du problème posé et essayer alors d'identifier des situations similaires où cette stratégie serait à même de fonctionner pour favoriser les transferts, ou bien des situations déjà vécues dans lesquelles nous aurions déjà exploité cette stratégie de résolution. Ce faisant, nous créons des liens entre des expériences distinctes. Ces prises de conscience viennent alors enrichir notre propre expérience et nous préparent à mieux affronter des situations inédites.

## 3 Fonctions cognitives

Les fonctions cognitives listées dans cette section sont issues des travaux de plusieurs chercheurs :

- Christine Mayer qui reprend les travaux de Reuven Feurenstein dans son *Manuel de métapédagogie* (2005) ;
- Pierre Vianin, tableau récapitulatif dans *Neurosciences cognitives et pédagogie spécialisée* (2010) ;
- Todd Lubart, Franck Zenasni, Baptiste Barbot, dans *Créativité en éducation et formation* (2016) ;
- Todd Lubart, Christophe Mouchiroud, Sylvie Tordjman, Franck Zenasni, dans la *Psychologie de la créativité* (2015).

Le lien entre l'algorithme de résolution de problèmes algorithmique et les fonctions cognitives listées ci-dessous est réalisé par un mot clé en gras et entre crochets dans le texte des chapitres précédents. Il identifie l'étape du processus durant lequel nous considérons que les fonctions cognitives associées sont plus particulièrement susceptibles d'être mobilisées par l'élève. Nous avons regroupé les différentes étapes identifiées dans trois phases de l'algorithme : (a) lecture et compréhension de l'énoncé, (b) résolution du problème et (c) expression de la solution. Elles sont représentées dans le diagramme ci-dessous (figure 2) :

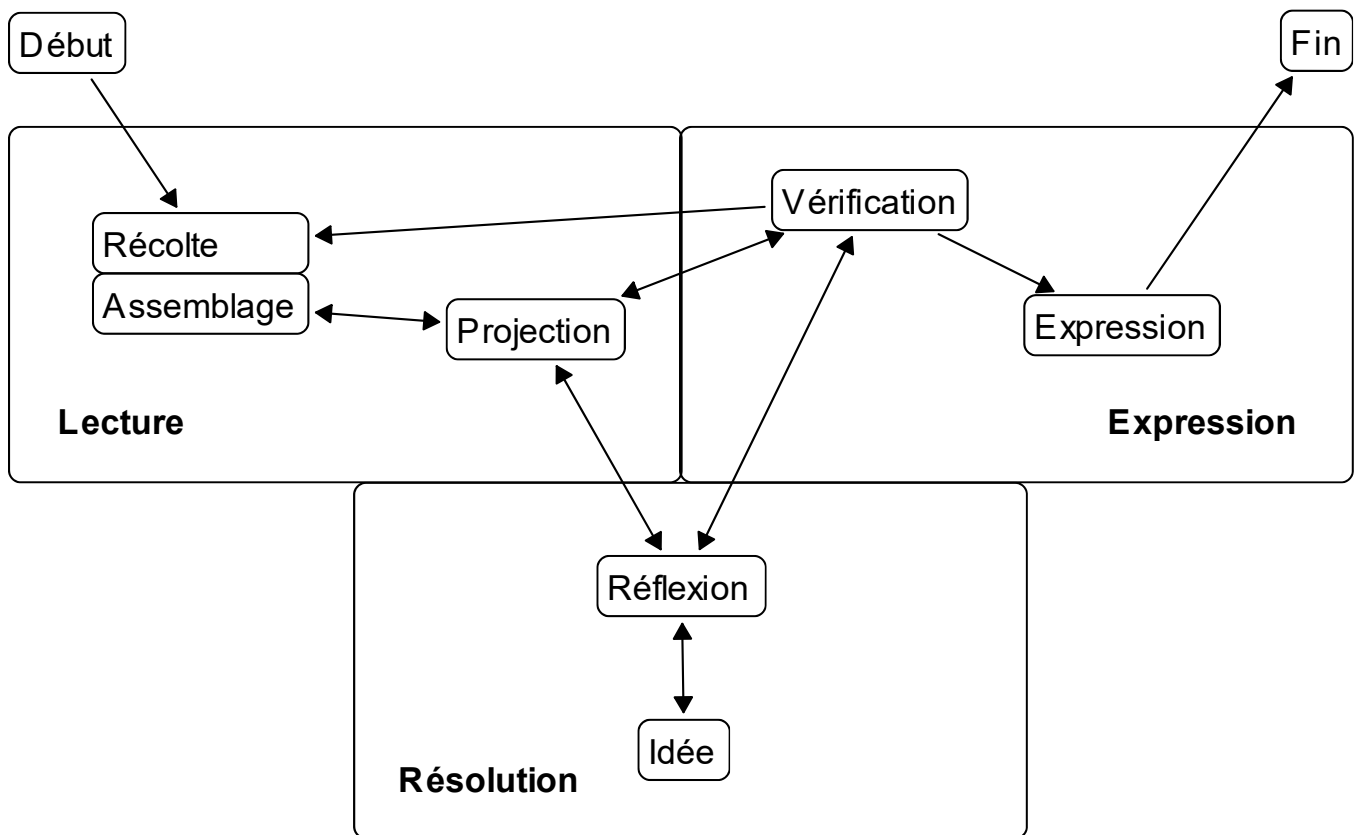


FIGURE 2 – Les étapes de l’algorithme de résolution de problèmes algorithmiques



Par souci de concision, nous listons ici uniquement les intitulés des fonctions cognitives sans en donner une description précise, que le lecteur retrouvera dans les travaux cités en référence. Ce travail de synthèse est un outil utile pour l'enseignant qui chercherait à faire une analyse clinique des éventuelles causes de blocage chez l'élève en difficulté.

- **[Récolte]** *La récolte des données* : attention, exploration, méthode, rigueur, identification, gestion.
- **[Assemblage]** *L'assemblage des données* : évocation, connexion, catégorisation, organisation, comparaison, sériation, sélection.
- **[Projection]** *La projection d'une stratégie de résolution* : reformulation, délimitation, fluidité, problématisation, anticipation, structuration, planification, modélisation.
- **[Réflexion]** *L'élaboration d'une réflexion* : inférence, abduction, rétention, déduction, induction, intériorisation.
- **[Idée]** *La recherche d'une idée* : désinhibition, flexibilité, pensée convergente, pensée divergente, pensée analogique, pensée intuitive.
- **[Vérification]** *La vérification d'une réponse* : récapitulation, justification, précision, circonspection, monitoring.
- **[Expression]** *L'expression d'une réponse ou d'une solution* : régulation, décentration, dénomination, autocontrôle.

Cette liste offre aux enseignants de nouvelles lunettes de compréhension sur les principaux mécanismes cognitifs potentiellement mis en œuvre par l'élève dans les étapes de conception d'une solution à un problème algorithmique. Elle ouvre de nouvelles pistes de travail de remédiation ciblée, lorsqu'une éventuelle défaillance cognitive est suspectée ou identifiée. Loin d'être exhaustive, cette liste pragmatique se concentre sur les 42 fonctions cognitives que nous considérons être les plus pertinentes dans la tâche de résolution de problèmes algorithmiques. Il se peut que les raisons du blocage de l'élève soient ailleurs, mais nous espérons que dans la majorité des situations l'enseignant trouvera dans ce répertoire une des origines possibles à ses difficultés d'apprentissage.

## 4 Stratégies cognitives

Les stratégies cognitives listées dans cette section reprennent une bonne partie des idées présentées dans l'algorithme de résolution de problèmes algorithmiques et les reformulent pour les rendre accessibles aux élèves. Certaines de ces stratégies sont directement inspirées de Mayer (2005). D'autres ont été élaborées à partir des discussions et réflexions de groupe faites en classe. C'est le fruit d'un travail empirique de médiation (Chappaz, 1995) de la part de l'enseignant réalisant, avec les élèves, une analyse réflexive et métacognitive des stratégies concrètes et opérationnelles utilisées. Elles sont utiles pour aider les élèves en difficulté à trouver des solutions aux problèmes algorithmiques proposés. Elles sont présentées aux élèves sous la forme d'une phrase mnémotechnique (fournie ici en italique), suivi d'un texte simple et compréhensible, leur permettant de se souvenir de la stratégie qui aura été au préalable discutée en classe. Ces 16 stratégies portent sur 4 moments distincts du processus de résolution d'un problème algorithmique :

1. Avant de se mettre au travail, pour apprendre à se concentrer et se mettre en condition de travail :
  - *Je suis capable de le faire. Qu'est-ce que je dois faire ? Est-ce que j'ai bien toutes les données ? C'est OK de ne pas y arriver du premier coup ou de faire des erreurs : garde confiance en toi. Si c'était trop facile, tu n'apprendrais rien.*

- *Je me concentre pour réussir.* Est-ce que tu as besoin qu'on te résume ce qu'il faut faire ? Focalise ton attention sur la tâche que tu es en train d'accomplir. Tu penseras au reste plus tard. Il y a un temps pour apprendre et un temps pour s'amuser.
- *Je résiste à donner la première réponse sans vérifier.* Est-ce que tu as pris le temps avant de donner ta réponse ? Vérifie bien l'exactitude de ton idée. Parfois, la vérité se cache ailleurs. Une approche critique et réfléchie permet d'éviter les pièges.
- *Je pense avant d'agir.* Comment ne pas répondre au hasard ? Demande-toi, si tu fais ça, alors qu'est-ce qui se passe ? Tu exploreras très rapidement beaucoup plus de possibilités en raisonnant dans ta tête qu'en essayant pour de vrai.

2. Pendant la recherche d'une solution, pour apprendre à résoudre des problèmes :

- *Je fais des liens avec ce que je connais.* Qu'est-ce qui est pareil ? Qu'est-ce qui est différent ? Cherche des similitudes avec ce que tu sais déjà et fais une hypothèse "c'est comme si...". Observe les différences et essaye de voir si ton intuition est juste.
- *Je me construis une image mentale du problème.* Qu'est-ce qui est utile ? Qu'est-ce qui ne l'est pas ? Qu'est-ce que je garde dans ma tête ? Représente-toi visuellement ce qu'il faut faire avant de commencer, ça va t'aider à mieux réfléchir.
- *Je manipule mon image mentale pour résoudre le problème.* Utilise l'image mentale que tu as construite pour chercher une solution. Déplace les choses avec tes yeux pour vérifier tes idées.
- *Je cherche à simplifier le problème.* Découpe le problème en plus petits problèmes ou bien résous un problème semblable, mais plus simple. Qu'est-ce qui est constant ? Qu'est-ce qui est variable ? Essaye de généraliser ta solution.

3. Au moment où l'on est bloqué, pour apprendre à être créatif :

- *J'explore une autre piste.* Lorsqu'une piste de réflexion ne mène à rien, fais marche arrière et envisage d'autres alternatives partout où tu as dû faire des choix.
- *Je change de chemin.* Tu es bloqué et tu as listé toutes les possibilités et leurs alternatives ? Cherche alors à changer d'approche, de stratégie ou à faire totalement autrement.
- *Je trouve de nouvelles idées.* Ferme les yeux et évacue toutes les pensées parasites en utilisant ton attention pour faire taire la petite voix dans ta tête. Des idées vont émerger d'elles-mêmes.
- *J'utilise l'intelligence du groupe.* Vraiment bloqué ? Cherche de l'aide et lorsque tu as trouvé, regarde si tu peux à ton tour aider les autres à trouver par eux-mêmes. Demande-toi si les autres vont comprendre ce que tu as fait ou ce que tu dis.

4. Lorsque le travail est terminé, dans une étape d'analyse réflexive pour enrichir les apprentissages :

- *Quand la tempête est passée, j'apprends à baisser les tours.* Lorsque je ne trouve pas, que je suis bloqué, je ne dois pas stresser ni paniquer. Je dois juste m'habituer à accepter le doute et l'incertitude. C'est la condition pour laisser émerger de nouvelles idées.
- *Je réfléchis aux chemins pris pour apprendre.* Quelles sont les étapes que tu as suivies ? Juste après avoir résolu un problème, pose-toi la question de ce qui t'a permis de résoudre le problème pour t'en souvenir plus tard.
- *Je réfléchis plus pour travailler moins et apprendre mieux.* Qu'est-ce que tu as vu ou entendu dans ta tête ? Comment as-tu su que c'était correct ? Dirige ton attention sur ta manière d'apprendre, pour apprendre mieux et avec moins d'efforts.
- *Je réfléchis aux applications futures.* Quel principe général peux-tu dégager de cette expérience ? Comment l'appliquer ailleurs ou autrement ? Imagine comment réutiliser ce que tu apprends pour pouvoir faire des liens avec des situations futures.

## 5 Conclusion

Finalement, le meilleur moyen de développer l'intelligence algorithmique de l'élève, c'est de la mobiliser sur des problèmes simples au début pour alléger sa charge cognitive (Chanquoy, Tricot, Sweller, 2007) développer la précision de son modèle mental (Johnson-Laird, 1983) et la puissance de sa mémoire de travail en activant ses neurones de manière répétée et espacée (Masson, 2020), puis sur des problèmes variés de plus en plus complexes, pour qu'il puisse acquérir de nouvelles stratégies de résolution de problème et enrichir ses expériences. Mais dans tous les cas, il faut être capable d'accepter le doute de ne pas savoir faire, pour dépasser cet état et apprendre, petit à petit, à développer sa créativité, son esprit pratique et sa logique mathématique : les trois composantes fondamentales de l'intelligence algorithmique.

L'aptitude à résoudre des problèmes algorithmiques chez l'élève dépendra de 5 paramètres :

- Sa faculté à rester concentrer sur la tâche en cours, malgré les perturbations extérieures, afin de réussir à se construire une représentation mentale fidèle de l'état du système durant chaque étape de son processus de transformation, au fil de l'avancement de la réflexion.
- L'étendue de sa mémoire de travail et le nombre d'objets mentaux qu'il sera à même de prendre en considération simultanément afin de se constituer un modèle mental fidèle du système avec lequel il doit travailler.
- De bonnes hypothèses et stratégies cognitives de résolution de problèmes, avec un riche panel d'expériences et de stratégies ayant eu du succès auparavant. Cette richesse ne s'acquiert qu'avec l'expérience. Il faut donc s'entraîner à concevoir des algorithmes pour développer cette faculté.
- Son aptitude à gérer ses doutes et ses émotions lorsqu'il sera bloqué et qu'il ne saura plus quoi faire. Sa capacité à créer un état de tranquillité intérieure, en situation de stress, afin de laisser au cerveau la possibilité de trouver de nouvelles idées face au problème rencontré. Sa faculté à cultiver sa créativité en situation de blocage en acceptant sereinement de ne pas savoir quoi faire, sans paniquer, sans douter, sans se soucier du fait de ne pas savoir.
- La puissance du plaisir narcissique qu'il ressentira en résolvant les premiers problèmes et qui déterminera sa motivation à explorer davantage de problèmes algorithmiques.

Ainsi, un élève qui manquera de confiance dans sa capacité d'apprendre à résoudre des problèmes algorithmiques, se retrouvera en difficulté non pas parce qu'il ne peut pas apprendre à les résoudre, mais par ce qu'il ne supportera pas l'état de doute et d'incertitude nécessaire pour traverser la phase où l'on ne sait ni quoi faire ni comment faire. Une autre cause possible à son manque de persévérance est son sentiment d'impuissance perçue (Le Bossé, 2016) : "à quoi bon faire l'effort de continuer d'essayer si de toute façon je perds mon temps, car je n'en suis pas capable?". Il faut accepter avec humilité le fait que l'acte intellectuel qui conduit à la maîtrise de l'intelligence algorithmique n'est pas inné, mais qu'il se développe avec du temps, des efforts et en utilisant de bonnes stratégies.

Nous pensons que l'explicitation de fonctions et de stratégies cognitives en classe est une aide précieuse pour l'enseignement de l'intelligence algorithmique au plus grand nombre. Être capable d'identifier les blocages potentiels chez les élèves en difficulté demande une connaissance fine des mécanismes cognitifs à l'œuvre dans la dynamique d'apprentissage de l'algorithmique. L'algorithme de résolution de problèmes algorithmiques est une aide, à la fois pour l'enseignant et pour les élèves, qui permet d'avoir une vision d'ensemble des processus cognitifs en œuvre lors de la conception d'une solution algorithmique à un problème posé. Son apprentissage permet de prendre conscience de la complexité de l'acte de résolution de problèmes algorithmiques et des difficultés qui y sont liées.

L'étude du développement de l'intelligence algorithmique, c'est-à-dire des intelligences créative, pratique et logico-mathématique des élèves, est en soi un vaste domaine. Il est nécessaire de se préoccuper de l'expérience vécue par l'élève dans son ensemble : de ne pas se limiter à la chose à apprendre, mais d'étendre notre enseignement au développement, en classe, de l'affect et de toutes les intelligences de l'élève, quelque soit la discipline enseignée, y compris dans l'acte même d'apprendre. Ainsi, nous pensons qu'il faudrait compléter la formation des enseignants par l'étude approfondie des mécanismes de développement des intelligences de l'élève. Nous nous devons d'étudier l'acte de compréhension de l'élève via la perception, l'imagination et la mémoire : c'est-à-dire étudier le *noos* (dans l'œuvre de Platon « intelligence, esprit en tant qu'il perçoit et qu'il pense ») ou la *noèse* (dans la pensée phénoménologique d'Edmund Husserl (Leclercq et Richard, 2016) « l'activité du sujet connaissant »). Pour ce faire, nous proposons d'envisager la création d'une nouvelle discipline, complémentaire à la didactique et la pédagogie, fondée sur la psychologie de la créativité, la psychologie cognitive et les neurosciences de l'éducation, dont l'objet d'étude serait le développement des intelligences de l'élève à l'école et que nous pourrions nommer : la *noésiologie*.

## 6 Références

- Chanquoy, L. Tricot, A. Sweller, J. (2007). *La charge cognitive. Théorie et applications*. Paris : Armand Colin.
- Chappaz, G. (1995). *Comprendre et construire la médiation*. Marseille : CRDP, Université de Provence.
- Fournier, J-Y. (1999). *À l'école de l'intelligence*. Paris : ESF.
- Johnson-Laird, P. N. (1983). *Mental Models*. Cambridge, MA : Harvard University Press.
- Le Bossé, Y. (2016). *Sortir de l'impuissance. Tome 2 : aspects pratiques*. Québec : Ardis.
- Leclercq, B. Richard, S. (2016). « Husserl (A) ». Dans M. Kristanek, *l'Encyclopédie philosophique*. En ligne : <https://encyclo-philo.fr/husserl-a>
- Lemaire, P. Didierjean, A. (2018). *Introduction à la psychologie cognitive*. Bruxelles : De Boeck Supérieur.
- Lubart, T. Mouchiroud, C. Tordjman, S. Zenasni, F. (2015) *Psychologie de la créativité*. Paris : Armand Colin.
- Lubart, T. Zenasni, F. Barbot, B. (2016). Le potentiel créatif : de la mesure à son développement. Dans I. Capron Puozzo, *La créativité en éducation et formation* (pp. 65-78). Bruxelles : De Boeck Supérieur.
- Masson, S. (2020). *Activer ses neurones : pour mieux apprendre et enseigner*. Paris : Odile Jacob.
- Mayer, C. (2005). *Manuel de métapédagogie*. Menoncourt : Upbrainning. En ligne : <https://www.upbrainning.net/produit/manuel-de-metapedagogie/>
- Sternberg, R. J. (1985). *Beyond I.Q. : A triarchic theory of human intelligence*. New York : Cambridge University Press.
- Vianin, P. (2009). *L'aide stratégique aux élèves en difficulté scolaire – Comment donner à l'élève les clés de sa réussite ?* Bruxelles : De Boeck.
- Vianin, P. (2010). *Neurosciences cognitives et pédagogie spécialisée : un exemple d'évaluation diagnostique des processus cognitifs*. Berne : Centre Suisse de Pédagogie Spécialisée.