

Série 3 : Introduction aux algorithmes

1. Donner trois exemples de problèmes pour lesquels on peut développer un algorithme.
2. Pourquoi un algorithme de résolution d'un problème nous aide à écrire un programme pour résoudre ce problème à l'aide d'un ordinateur ?
3. En 1990, le nombre de transistors sur un microprocesseur était 10^6 . D'après la loi de Moore, quel est ce nombre en : a. 1992 / b. 1994 / c. 2000 / d. 2021 ?
4. Dites si chaque problème est *décisionnel* ou *de construction* (ou *d'évaluation*) :
 - a) Trouver le chemin le plus court entre deux points.
 - c) trouver si un nombre donné est un nombre premier.
5. Cherchez une solution au problème de la rivière.
Utilisez la force brute (envisager toutes les possibilités raisonnables)
Utilisez l'écriture proposée sous forme de quadruplet.
6. Expliquez à partir d'un exemple ce qu'est l'approche brutale (ou force brute) d'un problème algorithmique.
7. Expliquez à partir d'un exemple en quoi le temps de calcul peut être une limitation importante dans la résolution d'un problème algorithmique.
8.
 - a. Écrivez un pseudo-code pour illustrer une fonctionnalité simple d'un distributeur de billets. On lui donne la somme à distribuer et il donne cette somme au client (si possible en ne donnant pas trop de billets)...
 - b. Construisez l'organigramme de cette même fonctionnalité.
9. Soit l'algorithme suivant écrit en pseudo-code :

```
AFFICHER «Combien de fois allez-vous aux bains des Pâquis par an ?»  
DEMANDER nombre  
SI nombre > 25  
    AFFICHER «Il vaut la peine de prendre un abonnement annuel.»  
SINON  
    AFFICHER «Il n'est pas préférable de prendre l'abonnement annuel.»  
FIN
```

- a. Quel problème cet algorithme est-il censé résoudre ?
- b. Entourez les trois parties suivantes :
 - i. entrée de données (en rouge)
 - ii. traitement de données (en bleu)
 - iii. sortie de résultats (en noir)
- c. Écrivez cet algorithme sous forme d'un organigramme.