

Challenge d'algorithmique libres

Niveau très facile - Catégories aux échecs

On demande l'âge d'un individu à l'utilisateur. Ensuite, on lui donne sa catégorie de joueurs d'échec :

- Petits poussins : moins de 8 ans
- Poussins : 8 et 9 ans
- Pupilles : 10 et 11 ans
- Benjamins : 12 et 13 ans
- Minimes : 14 et 15 ans
- Cadets : 16 et 17 ans
- Juniors : 18 et 19 ans
- Seniors : de 20 à 49 ans
- Seniors + : de 50 à 64 ans
- Vétérans : 65 ans et plus

Niveau facile – Fusion de tableaux

On demande à l'utilisateur de saisir deux tableaux de nombres, dont la taille est aléatoirement comprise entre 4 et 8. L'algorithme construit ensuite un troisième tableau qui correspond à la fusion des deux tableaux. Ainsi, [2][9][8][0] et [1][66][3][8] donnera [2][9][8][0][1][66][3][8]

Options intermédiaires :

- Lors de la fusion, il faut enlever les éléments en doublon ([2][9][8][0][1][66][3])
- Lors de la fusion, il faut trier les éléments ([0][1][2][3][8][9][66])

Niveau facile – Détection de voyelles

On demande une phrase à l'utilisateur, puis on lui indique le nombre de voyelles contenues dans la phrase. L'algorithme sera insensible à la casse. On pourra écrire deux solutions. La première déploie une condition composée bien fastidieuse. La deuxième, utilisant un tableau de voyelles est plus propre.

Option intermédiaire :

- Rendez votre algorithme insensible aux accents (« é » compte comme une voyelle)

Niveau facile – Machine à café

L'algorithme demande à l'utilisateur ce qu'il veut boire : café court, café long, cacao, potage.

Ensuite, si c'est pertinent, on demande la quantité de sucre qu'il souhaite (entre 0 et 5).

Ensuite, on lui demande s'il possède déjà un gobelet ou s'il en veut un neuf.

Enfin : on lui demande de payer en espèces ou par carte bancaire.

Après chaque choix, vous pouvez implémenter des messages rigolos. À la fin, l'algo récapitule la commande (« Cotre café court non-sucré avec gobelet neuf réglé par carte bancaire est en cours de préparation... »)

Option intermédiaire :

- Selon le moyen de paiement, adaptez et implémentez l'algorithme de rendu de monnaie et/ou de vérification du numéro de carte bleu.

Niveau intermédiaire – Jeu du loto

L'utilisateur indique 5 numéros compris entre 1 et 39. Les numéros doivent être différents.

L'algorithme effectue ensuite un tirage aléatoire de 5 numéros compris entre 1 et 39 (là encore différents).

Le résultat du tirage apparaît ensuite, et l'algorithme indique à l'utilisateur combien de bon numéro il a obtenu.

Vous afficherez un message différent selon le nombre de numéros obtenus.

Niveau intermédiaire – Premier tour des élections présidentielles

L'utilisateur choisit un nombre de candidat. Puis, pour chaque candidat, l'utilisateur doit indiquer le nombre de vote comptabilisés au premier tour. Le logiciel affiche :

Le nombre total de votant

Le pourcentage des voix de chaque candidat

Le nom du candidat élu (si un candidat a recueilli 50 % des voix il est élu) ou alors les noms des deux candidats qui se présenteront au second tour.

Niveau intermédiaire – Recherche dans une chaîne

L'utilisateur saisit une phrase, qui est gardée en mémoire. Ensuite, on demande à l'utilisateur de saisir une chaîne. L'algorithme effectue une recherche dans la phrase, et dit si la recherche a fonctionné. L'index découvert est alors affiché. (Par exemple, recherche de « pizza » dans « J'ai vraiment faim, tu préfères une pizza ou un burger ? » indiquera : « Mot trouvé à la position 36 ! »). L'algorithme sera insensible à la casse. Seules les fonctions vues en cours pourront être utilisées.

Option avancée :

- Rendez votre algorithme insensible aux accents et aux caractères spéciaux

Niveau intermédiaire – Recherche dichotomique

L'algorithme génère seul un tableau composé de 200 valeurs aléatoirement comprises entre 1 et 500. Le tableau est ensuite trié, selon l'un des algorithmes vus en cours. On demande à l'utilisateur un nombre qu'il veut rechercher. La recherche s'opère de manière dichotomique (voir principe de cours ou explications sur Internet)

Niveau intermédiaire – Jeu du pendu

L'algorithme demande un mot à un utilisateur. Le mot est ensuite converti en majuscule (ex : « BOOLEEN »). L'algorithme effectue ensuite 14 sauts de lignes (pour que le mot n'apparaisse plus à l'écran), puis affiche des « _ » à la place des lettres (exemple : « _ _ _ _ _ »).

Un autre utilisateur peut maintenant proposer des lettres (qui seront converties en majuscules), afin d'essayer de deviner le mot. Si l'utilisateur réussit, les lettres sont affichées à la place des « _ » (exemple : « _ _ _ _ E E _ »). Sinon : rien ne se passe. Si l'utilisateur échoue 7 fois : il perd la partie, et le mot est révélé.

À la fin de la partie, on propose à l'utilisateur de rejouer ou de quitter.

Option avancée :

- L'utilisateur pourra demander à voir la liste des lettres proposées qui n'existent pas dans le mot en appuyant sur « ? »

Niveau avancé – Score au Bowling

Savez-vous comment fonctionnent les scores au Bowling ?

Concevez un algorithme de comptage des scores du Bowling. L'algorithme sera conçu pour un seul et unique joueur. Lorsqu'il fait un Strike, le joueur indique « X ». Lorsqu'il fait un Spare, le joueur indique « / ». Dans les autres cas, le joueur indique le nombre de quilles tombées lors de son lancé.

C'est à l'algorithme de gérer le déroulement de la partie de manière automatique : le joueur n'a pas à indiquer la fin des manches ou autre.

Une fois la partie terminée, un récapitulatif des scores est affiché, et l'algorithme demande si l'on veut faire une nouvelle partie ou quitter.

Option experte :

- À tout moment, le joueur pourra annuler sa dernière saisie en indiquant « Erreur ».

Niveau avancé – Suite de Conway

Connaissez-vous la suite de Conway ?

L'utilisateur choisit une quantité.

Puis, votre algorithme construit la Suite de Conway : « 1 », « 11 », « 21 », « 1211 », « 111221 », « 312211 », etc. Il s'arrête après avoir affiché la quantité demandée.

Option experte :

- Utilisez une fonction récursive pour générer la suite de Conway

Niveau avancé – Calculatrice de nombres romains

Connaissez-vous les chiffres Romain ?

On demande deux nombres romains à l'utilisateur (exemple : « CXI » pour 111 et « IX » pour « 9 »).

On convertit les chiffres romains en nombre, puis les additionne, et puis affiche le résultat à l'utilisateur sous la forme d'un chiffre romain (exemple : « CXX » pour 120).

Niveau avancé – Calcul d'amortissement

Connaissez-vous le principe d'amortissement d'un prêt (intérêts, mensualités, taux) ?

On demande à l'utilisateur le montant qu'il veut emprunter, la durée de son emprunt (en année), et le taux (en %) proposé. On calcul ensuite ses mensualités. Un tableau d'amortissement apparaît, indiquant la date du virement, le capital remboursé et la part des intérêts. À la fin, le montant total à rembourser apparaît.

Exemple : Pour un emprunt de « 1000 € » sur 1 an (12 mois) à taux « 1 % », les mensualités sont de 167,57 €, et le tableau d'amortissement est le suivant :

Date	Capital remboursé	Intérêts
02/2021	165,91 €	1,67 €
03/2021	331,96 €	1,53 €
04/2021	498,14 €	1,39 €
05/2021	664,47 €	1,25 €
06/2021	830,93 €	1,11 €
07/2021	997,53 €	0,97 €

Date	Capital remboursé	Intérêts
08/2021	1164,27 €	0,84 €
09/2021	1331,15 €	0,70 €
10/2021	1498,17 €	0,56 €
11/2021	1665,33 €	0,42 €
12/2021	1832,62 €	0,28 €
13/2021	2000,00 €	0,14 €

Niveau expert – Machine à sous type Poker

Connaissez-vous les combinaisons du Poker ? Connaissez-vous les machines à sous de type Poker ?



On demande à l'utilisateur son nombre de crédits, puis le jeu démarre.

Chaque partie coûte 1 crédit. On affiche à l'utilisateur 5 cartes à jouer (jeu de 52 cartes).

L'utilisateur peut choisir entre 0 et 5 cartes. Une fois que l'utilisateur termine son choix, les cartes choisies sont conservées, et les autres sont remplacées avec de nouvelles cartes.

Ensuite, la machine indique la meilleure combinaison de Poker ainsi formée à l'écran.

Si aucune combinaison gagnante n'apparaît : l'utilisateur perd le crédit, et une nouvelle partie démarre.

Si l'utilisateur gagne, il a le choix entre :

- Garder son gain (dépendant de la combinaison obtenue) ; une nouvelle partie démarre.
- Déclencher un « Quitte ou double ? » :
 Dans un pareil cas, il remet son gain en jeu. La machine révèle une carte face visible, et quatre cartes face cachées. L'utilisateur doit alors choisir une des cartes qui est révélé par la machine. Trois situations peuvent survenir :
 - Si la carte révélée par l'utilisateur est plus forte que celle de la machine, les gains sont doublés, et l'utilisateur peut à nouveau choisir entre garder son gain (doublé) et déclencher un nouveau « Quitte ou double ? »
 - Si la carte révélée par l'utilisateur est égale à celle de la machine, l'utilisateur doit choisir une autre carte.
 - Si la carte révélée par l'utilisateur est moins forte que celle de la machine, l'utilisateur perd ses gains ; une nouvelle partie démarre.

Avant chaque nouvelle partie, l'utilisateur peut choisir d'arrêter de jouer.