TD Algorithmique et programmation

18 mars 2008

TD 1 : les entiers

On définit les nombres **parfaits** par :

Définition

un nombre naturel est parfait s'il est égal à la somme de ses diviseurs propres (c'est-à-dire tous les nombres entiers qui le divisent sauf lui-même)

Par exemple, 9 n'est pas parfait car $9 \neq 1 + 3$ mais 6 est parfait car $6 = 1 + 2 + 3$.

TD 1 : les entiers

1.

Ecrire une fonction Som_Diviseur_propre qui a pour paramètre un entier naturel k et qui renvoie la somme des diviseurs propres de k.

2.

Ecrire une fonction Est_Parfait qui a pour paramètre un entier naturel k et qui renvoie vrai si k est parfait, faux sinon.

TD 1 : les entiers

Programme principal

Ecrire le programme pour qu'il demande à l'utilisateur combien de nombres parfaits il souhaite puis qui les affiche.

TD 2 : les flottants

Soit $x \in \mathbb{R}_+$. La suite $v_{n+1} = \dfrac{v_n + \frac{x}{v_n}}{2}$ converge vers $\sqrt{x}$.

TD 2 : les flottants

Soit $x \in \mathbb{R}_+$. La suite $v_{n+1} = \frac{v_n + \frac{x}{v_n}}{2}$ converge vers $\sqrt{x}$.

Programme

Ecrire un programme qui

- lit un flottant positif x
- lit un flottant positif eps
- affiche le premier terme de la suite v_n tel que $|v_n^2 - x| < eps$

TD 3 : les tableaux

Tableau 1.

Ecrire une fonction Max qui a pour paramètre un tableau et qui renvoie la valeur du plus grand entier du tableau.

TD 3 : les tableaux

Tableau 1.

Ecrire une fonction Max qui a pour paramètre un tableau et qui renvoie la valeur du plus grand entier du tableau.

Tableau 2.

Ecrire une fonction Deux_Max qui a pour paramètre un tableau et qui renvoie la deuxième plus grande valeur du tableau

- en utilisant la fonction Max
- en un seul passage sans utiliser la fonction Max.

TD 4 : le drapeau hollandais

On suppose que l'on a un tableau de longueur n qui ne contient que deux valeurs possibles (0,1)

Ecrire une procedure Tri_Drap_paris qui a pour paramètre un tableau de ce type t_drap_paris et qui le trie en un seul passage (pas de boucle imbriquée).

TD 4 : le drapeau hollandais

On suppose que l'on a un tableau de longueur n qui ne contient que deux valeurs possibles (0,1)

Ecrire une procedure Tri_Drap_paris qui a pour paramètre un tableau de ce type t_drap_paris et qui le trie en un seul passage (pas de boucle imbriquée).

On suppose que l'on a un tableau de longueur n qui ne contient que trois valeurs possibles (0,1,2)

TD 4 : le drapeau hollandais

On suppose que l'on a un tableau de longueur n qui ne contient que deux valeurs possibles (0,1)

Ecrire une procedure Tri_Drap_paris qui a pour paramètre un tableau de ce type t_drap_paris et qui le trie en un seul passage (pas de boucle imbriquée).

On suppose que l'on a un tableau de longueur n qui ne contient que trois valeurs possibles (0,1,2)

Ecrire une procedure Tri_Drap qui a pour paramètre un tableau de ce type t_drapeau et qui le trie en un seul passage (pas de boucle imbriquée).

TD 4 : le drapeau hollandais

Il faut gérer les plages de 0, de 1 et de 2 en même temps.
On va parcourir le tableau de gauche à droite et on envisage les actions à exécuter dans les 3 cas suivants :

- quand on rencontre un 0 il faut le mettre à gauche avant la plage des 1 donc la plage des 1 se trouvera décalée d'un rang vers la droite ; pour éviter un décalage de toute la plage, il suffit d'échanger ce 0 avec **le premier 1** puis de retenir que la plage des 1 démarre un rang plus loin. Puis on avance dans la lecture du tableau
- quand on rencontre un 1 on avance d'un rang
- quand on rencontre un 2 il faut le mettre à droite avec les autres 2. Si on le place à l'extrême droite les autres 2 doivent être décalés d'un rang vers la gauche. Pour éviter ce décalage, on va échanger ce 2 avec la valeur qui se trouve juste avant le premier 2, valeur que l'on ne connaît pas que l'on doit alors traiter. Il faut retenir dans ce cas que l'indice avant le premier 2 démarre un rang plus tôt.

Ne pas oublier les initialisations, et les cas extrêmes pour lesquels il n'y aurait aucun 0 ou aucun 1 ou aucun 2.

TD 5 : la recherche dichotomique

La recherche dichotomique dans un tableau **trié** repose sur le principe suivant : si la recherche s'effectue entre les indices b_i et b_s on compare la valeur recherchée x avec la valeur qui est à égale distance des bornes b_i et b_s, soit à l'indice $m = \frac{b_i + b_s}{2}$

- **si** x est égale à la valeur en m, **alors** la recherche s'arrête positivement, **sinon**
- **si** x est inférieur à la valeur en m **alors** on continue la recherche entre les indices b_i et $m - 1$ **sinon**
- on continue la recherche entre les indices $m + 1$ et b_s.

TD 5 : la recherche dichotomique

Fonction

Ecrire une fonction rech_dicho qui a pour paramètre un tableau que l'on suppose trié, un entier x et qui renvoie vrai si x est dans le tableau faux sinon, selon le principe énoncé ci-dessus.

TD 5 : la recherche dichotomique

Fonction

Ecrire une fonction rech_dicho qui a pour paramètre un tableau que l'on suppose trié, un entier x et qui renvoie vrai si x est dans le tableau faux sinon, selon le principe énoncé ci-dessus.

On précisera notamment les **conditions d'arrêt** de la recherche dans le cas où elle se révèle négative

TD 6 : le tri par insertion

Le principe de ce tri est le suivant :
On suppose trié les valeurs depuis l'indice 1 jusqu'à l'indice i, soit $x_1 < \cdots < x_i$.
On considère la valeur $v = x_{i+1}$: elle est à l'indice $i + 1$. On la compare à sa voisine de gauche et on l'échange avec elle si elle lui est inférieure : v est alors à l'indice i. On poursuit ces échanges jusqu'à ce que v soit à l'indice 1 ou bien que la valeur à gauche de v lui soit inférieure.
Ce principe est appliqué pour $i = 1$ jusqu'à $i = n - 1$.

TD 6 : le tri par insertion

Par exemple, si $i = 5, n = 7$ et le tableau est

1	2	3	4	**5**	6	7
10	14	16	17	**11**	9	12

alors on a

TD 6 : le tri par insertion

Par exemple, si $i = 5, n = 7$ et le tableau est

1	2	3	4	**5**	6	7
10	14	16	17	**11**	9	12

alors on a

1	2	3	**4**	5	6	7
10	14	16	**11**	17	9	12

TD 6 : le tri par insertion

Par exemple, si $i = 5, n = 7$ et le tableau est

1	2	3	4	**5**	6	7
10	14	16	17	**11**	9	12

alors on a

1	2	3	**4**	5	6	7
10	14	16	**11**	17	9	12

1	2	**3**	4	5	6	7
10	14	**11**	16	17	9	12

TD 6 : le tri par insertion

Par exemple, si $i = 5, n = 7$ et le tableau est

1	2	3	4	**5**	6	7
10	14	16	17	**11**	9	12

alors on a

1	2	3	**4**	5	6	7
10	14	16	**11**	17	9	12

1	2	**3**	4	5	6	7
10	14	**11**	16	17	9	12

1	**2**	3	4	5	6	7
10	**11**	14	16	17	9	12

TD 6 : le tri par insertion

Procédure

Ecrire une procédure tri_insert qui a pour paramètre un tableau et qui trie ce tableau selon le principe énoncé ci-dessus.

TD 7 : Valeur majoritaire dans un tableau

Fonction

Ecrire une fonction Val_Maj qui a pour paramètre un tableau que l'on suppose trié, et qui renvoie la valeur qui apparaît le plus souvent dans ce tableau.

TD 7 : Valeur majoritaire dans un tableau

Fonction

Ecrire une fonction Val_Maj qui a pour paramètre un tableau que l'on suppose trié, et qui renvoie la valeur qui apparaît le plus souvent dans ce tableau.

Une variante

Ecrire une fonction Nbr_Maj qui a pour paramètre un tableau que l'on suppose trié, et qui renvoie le **nombre d'occurrences** de la valeur qui apparaît le plus souvent dans ce tableau.

TD 8 : Palindrome

Définition

Une suite de caractères est un palindrome s'il peut se lire de la même façon dans les deux sens (sans tenir compte des espaces).

TD 8 : Palindrome

Définition

Une suite de caractères est un palindrome s'il peut se lire de la même façon dans les deux sens (sans tenir compte des espaces).

Exemple

laval
A man, a plan, a canal : Panama

TD 8 : Palindrome

On s'intéresse aux nombres entiers palindrome.

Par exemple pour 2345432 il faut pouvoir vérifier que le chiffre des unités est égal au chiffre des millions, puis le chiffre des dizaines au chiffre des centaines de mille ...

TD 8 : Palindrome

On s'intéresse aux nombres entiers palindrome.

Par exemple pour 2345432 il faut pouvoir vérifier que le chiffre des unités est égal au chiffre des millions, puis le chiffre des dizaines au chiffre des centaines de mille ...
Comment récupérer ces chiffres dont on ne connaît à priori pas la quantité ?

TD 8 : Palindrome

Procédure

Écrire une procédure Decomp_tab qui a pour paramètre un entier naturel n, un tableau de chiffres t, qui affecte au tableau chaque chiffre composant n depuis le chiffre des unités en t(1) jusqu'au dernier chiffre coefficient de la plus grande puissance de 10 en t(**k**).

TD 8 : Palindrome

Procédure

Écrire une procédure Decomp_tab qui a pour paramètre un entier naturel n, un tableau de chiffres t, qui affecte au tableau chaque chiffre composant n depuis le chiffre des unités en t(1) jusqu'au dernier chiffre coefficient de la plus grande puissance de 10 en t(**k**).

Que faire de k ?

TD 8 : Palindrome

Procédure

Écrire une procédure Decomp_tab qui a pour paramètre un entier naturel n, un tableau de chiffres t, qui affecte au tableau chaque chiffre composant n depuis le chiffre des unités en t(1) jusqu'au dernier chiffre coefficient de la plus grande puissance de 10 en t(**k**).

Que faire de k ?

k doit être un paramètre de Decomp_tab en mode ?

TD 8 : Palindrome

Procédure

Écrire une procédure Decomp_tab qui a pour paramètre un entier naturel n, un tableau de chiffres t, qui affecte au tableau chaque chiffre composant n depuis le chiffre des unités en t(1) jusqu'au dernier chiffre coefficient de la plus grande puissance de 10 en t(**k**).

k doit être un paramètre de Decomp_tab en mode ?

en mode OUT

TD 8 : Palindrome

Fonction

Écrire une fonction Sym_tab qui a pour paramètre un tableau de chiffres t, un entier k et qui renvoie vrai si le tableau est symétrique entre les rangs t'first et k, faux sinon .

Un tableau d'entiers sera symétrique entre les rangs i et j si $t(i + r) = t(j - r)$ pour $r = 0 \ldots \frac{i+j}{2}$.

TD 8 : Palindrome

Fonction

Écrire une fonction Sym_tab qui a pour paramètre un tableau de chiffres t, un entier k et qui renvoie vrai si le tableau est symétrique entre les rangs t'first et k, faux sinon .

Un tableau d'entiers sera symétrique entre les rangs i et j si $t(i + r) = t(j - r)$ pour $r = 0 \dots \frac{i+j}{2}$.

Par exemple

1	2	3	4	5	6	7
10	32	14	16	14	32	10

TD 8 : Palindrome

Programme principal

Écrire un programme qui lit un entier n et qui affiche oui si n est un palindrome et non dans le cas contraire.