

Algorithmique et Bases de la Programmation

Durée : 3 heures – Session 1 du 25 janvier 2022

Ce sujet comporte : 6 pages

Matériel autorisé : Aucun

IMPORTANT : PRENDRE LE TEMPS DE LIRE LES CINQ REMARQUES CI-DESSOUS

Remarque 1 :

Réalisez la partie 1 et la partie 2 sur des copies différentes, pour permettre la correction.

Merci par avance.

Remarque 2 :

La partie 1 sera notée sur 10 et la partie 2 sera notée sur 10 également, donc essayez de consacrer une heure et 30 minutes à chacune de ces deux parties.

Remarque 3 :

Les programmes de la partie 1 doivent impérativement être écrits en langage algorithmique.

Remarque 4 :

Les programmes de la partie 2 doivent impérativement être écrits en langage C

Remarque 5 :

Les programmes seront exclusivement rédigés de façon manuscrite.

PARTIE 1 : LANGAGE ALGORITHMIQUE

PROGRAMMES A ECRIRE IMPERATIVEMENT EN LANGAGE ALGORITHMIQUE

EXERCICE 1 : DES MOTS ET DES PHRASES

On considère un tableau de caractères TAB, défini de la façon suivante :

CONSTANTE

NBMAX = 100 ;

TYPE

TYPTAB = TABLEAU [1 .. NBMAX] DE CARACTERE ;

VARIABLE

TAB : TYPTAB ;

Question 1 :

Ecrire une procédure "Remplissage" qui permet de saisir les DIM éléments du tableau TAB.

Le tableau TAB est initialisé avec des caractères ' '.

Le nombre de caractères DIM est défini par lecture à l'intérieur de la procédure et est passé en paramètre de la procédure.

Exemple :

Pour le tableau TAB suivant :

E	N	G	A	G	E		L	E		J	E	U		Q	U	E		J	E		L	E		G	A	G	N	E
---	---	---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	--	---	---	--	---	---	--	---	---	---	---	---

DIM = 29.

Question 2 :

Ecrire une procédure "Affichage" qui permet d'afficher les DIM éléments du tableau TAB.

Question 3 :

Un **palindrome** est un mot ou une phrase, pour lequel la signification est la même dans les deux sens de lecture, de la gauche vers la droite ou de la droite vers la gauche.

Exemples de palindromes :

RADAR, KAYAK, ETE, ICI, REVER, ARA, COLOC, ELLE, BOB

Ecrire une procédure "Palindrome" qui permet d'afficher un message précisant si un mot ou une phrase est un palindrome ou non.

Le message affiché sera donc : « Ceci est un palindrome. » ou « Ceci n'est pas un palindrome. »

On utilisera pour cela un tableau BIS contenant les mêmes caractères que le tableau TAB, après avoir supprimé les espaces (caractères ' ').

Le tableau BIS ne sera pas affiché.

Exemple :

Pour le tableau TAB défini précédemment, le tableau BIS obtenu est le suivant :

E	N	G	A	G	E	L	E	J	E	U	Q	U	E	J	E	L	E	G	A	G	N	E
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Le message affiché est le suivant : **Ceci est un palindrome.**

Question 4 :

Ecrire une procédure "ComptageMots" qui permet de compter le nombre de mots contenus dans une phrase (ou une expression) représentée dans un tableau TAB.

Exemple :

Pour le tableau TAB suivant :

A	L	B	E	R	T		E	I	N	S	T	E	I	N
---	---	---	---	---	---	--	---	---	---	---	---	---	---	---

ComptageMots (TAB, DIM, NBMOTS) va renvoyer la valeur : NBMOTS = 2

Question 5 :

Ecrire une procédure "Inversion" qui permet d'inverser l'ordre des DIM éléments du tableau TAB.
Ne pas utiliser d'autre tableau que le tableau TAB.

Exemple :

Pour le tableau TAB suivant :

L	U	M	I	E	R	E
---	---	---	---	---	---	---

Inversion (TAB, DIM) va renvoyer le tableau TAB modifié suivant :

E	R	E	I	M	U	L
---	---	---	---	---	---	---

Question 6 :

Ecrire une procédure "NombreOccurrences" qui permet de compter le nombre d'occurrences du caractère VAL (passé en paramètre de la procédure) dans le tableau TAB.
Ne pas utiliser d'autre tableau que le tableau TAB.

Exemple :

Pour le tableau TAB suivant :

E	L	U		P	A	R		C	E	T	T	E		C	R	A	P	U	L	E
---	---	---	--	---	---	---	--	---	---	---	---	---	--	---	---	---	---	---	---	---

Si VAL = 'E' alors NombreOccurrences (TAB, DIM, VAL, NB) va renvoyer la valeur : NB = 4

Question 7 :

On appelle **bigramme** une suite de deux caractères.

Ecrire une fonction "Bigramme" qui permet de compter le nombre d'occurrences d'un bigramme dans le tableau TAB.

Le bigramme sera défini par lecture à l'intérieur de la procédure et représenté sous la forme d'un tableau de deux caractères appelé BIG.

Exemple :

Soit le tableau BIG suivant :

T	T
---	---

Bigramme (TAB, DIM) va renvoyer la valeur : 1

EXERCICE 2 : CONVERSION D'UN NOMBRE ENTIER EN BASE 8

Ecrire un programme qui permet de réaliser la conversion d'un nombre entier quelconque (base 10) défini par lecture, en octal (base 8).

Le résultat sera stocké dans un tableau OCT, dont chaque élément contiendra une valeur de 0 à 7.

Le tableau OCT sera ensuite affiché.

Exemple :

Le nombre entier défini par lecture est 17.

$$17_{10} = 21_8$$

$$17_{10} = 2 * 8^1 + 1 * 8^0$$

Le tableau OCT contiendra donc les valeurs suivantes :

2	1
---	---

EXERCICE 3 : CONVERSION D'UN NOMBRE ENTIER EN BASE 10

Ecrire un programme qui permet de réaliser la conversion d'un nombre écrit en octal (base 8), en décimal (base 10).

Le nombre écrit en octal est mémorisé dans un tableau OCT, dont chaque élément contient une valeur de 0 à 7.

Le résultat sera ensuite affiché.

Exemple :

Le tableau OCT contient les valeurs suivantes :

1	4	4
---	---	---

$$144_8 = 100_{10}$$

$$144_8 = 1 * 8^2 + 4 * 8^1 + 4 * 8^0 = 64 + 32 + 4 = 100$$

Le résultat affiché sera donc : 100.

EXERCICE 4 : NOMBRE PARFAIT

Ecrire un programme qui permet d'afficher si un nombre, défini par lecture, est parfait ou non.

Définition :

Un nombre parfait est un nombre présentant la particularité d'être égal à la somme de tous ses diviseurs, excepté lui-même.

Exemples :

6 est parfait (car $6 = 1 + 2 + 3$)

28 est parfait (car $28 = 1 + 2 + 4 + 7 + 14$)

PARTIE 2 : LANGAGE C

PROGRAMMES A ECRIRE IMPERATIVEMENT EN LANGAGE C

Les programmes des exercices qui suivent doivent être écrits en langage C.

L'annexe vous rappelle les prototypes de quelques fonctions de la bibliothèque standard du langage C.

EXERCICE 1 :

Dans cet exercice, **vous n'avez pas le droit d'utiliser les fonctions de la bibliothèque standard**. Les fonctions demandées aux questions Q1, Q2, Q3 et Q4 doivent être sans saisie et sans affichage.

Q1. Ecrire une fonction qui calcule et renvoie la longueur d'une chaîne de caractères.

Remarque : dans la table des codes ASCII, les codes ASCII des chiffres « se suivent » :

- le code ASCII du chiffre '0' est égal à 48 en décimal (0x30 en hexadécimal) ;
- le code ASCII du chiffre '1' est égal à 49 en décimal (0x31 en hexadécimal) ;
- le code ASCII du chiffre '2' est égal à 50 en décimal (0x32 en hexadécimal) ;
- ...
- le code ASCII du chiffre '9' est égal à 57 en décimal (0x39 en hexadécimal) ;

Q2. Ecrire une fonction qui convertit une chaîne de caractères numérique en entier.

La chaîne de caractères reçue en paramètre représente un entier écrit en décimal.

Les éventuels espaces placés en début de chaîne et les éventuels caractères non valides situés en fin de chaîne doivent être ignorés.

Exemples :

Chaîne de caractères à convertir	Résultat
" 456"	456
"abc"	0
"-0"	0
"-+15"	0
"-17hello "	-17
" +305test"	305

Q3. Ecrire une fonction qui convertit un entier en chaîne de caractères qui le représente en décimal.

La fonction reçoit notamment en paramètre l'adresse du premier élément d'un tableau de caractères alloué dans la fonction appelante.

On supposera que ce tableau a toujours une taille suffisante pour pouvoir recevoir le résultat de la conversion.

Q4. Ecrire une fonction qui recherche la première occurrence d'une sous-chaîne *needle* dans une chaîne *haystack*.

Les caractères '\0' ne sont pas comparés.

Cette fonction renvoie un pointeur sur le début de la sous-chaîne *needle* (dans la chaîne *haystack*), ou NULL si la sous-chaîne *needle* n'est pas trouvée.

Q5. Ecrire une fonction `main()` la plus simple possible qui appelle la fonction définie à la question Q3.

EXERCICE 2 : TABLEAU D'ENTIERS

Q1. Ecrire une fonction qui renvoie un entier aléatoire compris entre deux bornes B1 et B2 incluses (c'est-à-dire dans l'intervalle [B1 ; B2]). Les bornes B1 et B2 seront transmises en paramètre à la fonction.

N.B. : Vous utiliserez la fonction **rand()** qui renvoie un entier pseudo-aléatoire compris entre 0 et RAND_MAX, bornes incluses (c'est-à-dire dans l'intervalle [0 ; RAND_MAX]).

RAND_MAX est une constante dont la valeur peut varier d'un compilateur à l'autre, mais elle est forcément d'au moins 32767.

On supposera que $(B2 - B1)$ est toujours inférieur ou égal à RAND_MAX.

Q2. Ecrire une fonction qui initialise un tableau d'entiers avec des nombres aléatoires compris entre deux bornes B1 et B2 incluses.

Les bornes B1 et B2 seront notamment transmises en paramètre à la fonction.

Q3. Ecrire une fonction qui compte le nombre de valeurs d'un tableau d'entiers égale à une valeur passée en paramètre.

Cette fonction doit également permettre de récupérer l'indice de la première occurrence de la valeur recherchée.

Si la valeur recherchée n'est pas trouvée, l'indice récupéré doit être égale à la taille du tableau.

Cette fonction doit être sans saisie et sans affichage.

Q4. Ecrire une version récursive de la fonction de la question précédente.

Cette fonction ne doit pas utiliser de boucle for, while ou do while.

Q5. Ecrire la définition d'une fonction qui trie dans l'ordre croissant un tableau d'entiers suivant l'algorithme « tri par sélection ».

Q6. Ecrire une définition main qui appelle les fonctions définies aux questions Q2 et Q3.

La valeur recherchée sera saisie par l'utilisateur, et les résultats seront affichés sur la sortie standard.

Annexe : prototypes de fonctions de la bibliothèque standard

```
int rand(void);
void srand(unsigned int seed);
int printf(const char *format, ...);
int scanf(const char *format, ...);
```