

LIFAP1 – Partie A - Algorithmique

Contrôle Continu Terminal (Durée : 2h)

Jeudi 20 décembre 2018

***Recommandations :** Les documents, calculatrice, téléphone portable sont interdits. La qualité de l'écriture et de la présentation seront prises en compte dans la note finale. Vous veillerez à **respecter** les notations et les règles d'écriture des algorithmes vues en cours et en TD.*

1. Génération automatique de mots Insensé non !!!

On souhaite écrire un programme permettant la génération automatique de chaînes de caractères. Chacun des caractères de la chaîne sera choisi de manière aléatoire parmi tous les caractères alphabétiques minuscules (de 'a' à 'z').

Les tirages aléatoires pourront se faire en utilisant une fonction `alea()` qui retourne une valeur entière comprise entre 0 et une constante `RANDMAX` du système.

- a- Écrire l'algorithme d'une **fonction** `genere_lettre` qui retourne un caractère alphabétique compris entre 'a' et 'z'. Pour rappel, l'alphabet comporte 26 lettres.

NOM :

.....

PRENOM :

.....

Numéro Etudiant :

.....

b- En utilisant la fonction précédente, écrire l'algorithme d'un sous-programme `genere_mot` qui permet de générer une chaîne de caractères dont la longueur sera passée en paramètre. Exemple : si on souhaite générer un mot de 8 caractères, on pourra obtenir : "hdoqpndi".

c- Ecrire l'algorithme du programme principal qui demande à l'utilisateur de choisir la longueur de la chaîne de caractère à générer (on recommencera la saisie de cette valeur tant qu'elle n'est pas strictement positive), de générer la chaîne, puis de l'afficher.

2. Ca ne tourne pas très rond !

- a- Ecrire l'algorithme d'une **fonction** `saisie_valeur` qui retourne une valeur `n` impaire et supérieure ou égale à 9 choisie par l'utilisateur. Attention la saisie devra être recommencée tant que la valeur proposée ne satisfait pas ces contraintes.

[illegible][illegible]

- b-Ecrire l'algorithme d'un sous-programme `affiche_cercle_plein` permettant d'afficher, dans la console, le cercle ci-contre. Le caractère utilisé ainsi que le rayon du cercle seront passés en paramètres. Pour rappel un point B (x,y) appartient à un cercle de rayon R et de centre O (0,0) si $OB \leq \sqrt{x^2 + y^2}$.

+++++

+++++

+++++

+++++

+++++

+++++

+++++

- c- Ecrire l'algorithme du programme principal permettant de demander à l'utilisateur un caractère `c1` et un entier impair `lg` supérieur ou égal à 9 (on utilisera la fonction écrite en a) et d'afficher le motif demandé.

LIFAP1 – Partie B – Langage C/C++

Contrôle Continu Terminal (Durée : 2h)

Jeudi 20 décembre 2018

***Recommandations :** Les documents, calculatrice, téléphone portable sont interdits. La qualité de l'écriture et de la présentation seront prises en compte dans la note finale. Vous veillerez à **respecter** les notations et les règles d'écriture des programmes vues en cours et en TP.*

Gestion d'une maternité

On souhaite écrire un programme permettant la gestion d'une maternité où naissent chaque année des bébés.

- Un bebe est identifié par son `prenom`, sa `date_de_naissance` (sous la forme jour, mois et annee), son `poids` et la `duree` de son séjour dans la maternité.
- Une `maternite` tient à jour un registre des naissances pour une année entière. Elle dispose ainsi d'un compteur du nombre de naissances depuis le début de l'année et d'un `tableau_de_bebes`.

1. Définir 2 constantes `CHMAX` (qui sera utilisée pour les chaines de caractères) et `MAXNAISSANCE` ayant pour valeurs respectives 32 et 20.

2. Définir les structures `date`, `bebe` et `maternite` permettant de répondre aux contraintes fixées.

NOM :

.....

PRENOM :

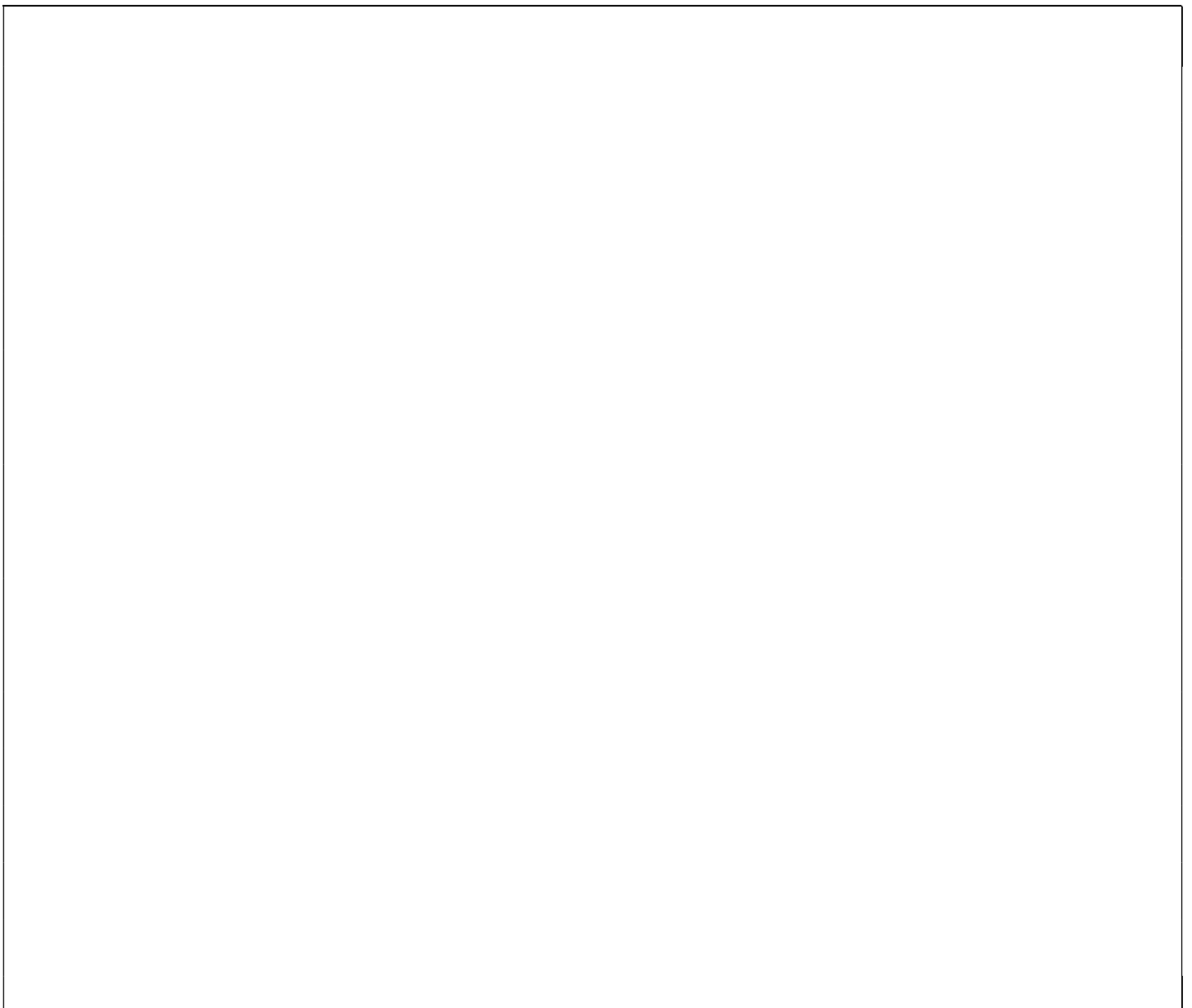
.....

Numéro Etudiant :

.....



3. Ecrire en langage C/C++ une **procédure** `remplir_bebe` permettant de remplir une structure `bebe` avec des informations choisies par l'utilisateur. Attention, la saisie du `poids` et de la `duree` du séjour devront être recommencées tant que les valeurs ne sont pas strictement positives.



4. Ecrire en langage C/C++ une **procédure** `ajoute_un_bebe` permettant d'ajouter un `bebe` à une `maternite` s'il reste encore de la place. On fera appel à la procédure écrite en 3.

5. Ecrire en langage C/C++ une **procédure** `afficher_dernier_bebe` permettant d'afficher les informations relatives au dernier `bebe` ajouté à la maternité.

6. Ecrire en langage C/C++ une **procédure** `bebe_par_mois` permettant de remplir et de "retourner" un tableau de 12 entiers contenant le nombre de naissances par mois dans la maternité.

7. En utilisant le sous-programme précédent, écrire en langage C/C++ un sous-programme `statistiques_annuelles` permettant de "retourner" l'indice du mois ayant le plus de naissances et l'indice du mois ayant le moins de naissances dans la maternité.

8. Ecrire en langage C/C++ une **fonction** `cherche_prenom` permettant de compter le nombre de bébés dans une maternité ayant un prénom choisi par l'utilisateur et passé en paramètre.

9. Ecrire en langage C/C++ une **fonction** `menu` permettant d'afficher à l'utilisateur les possibilités de tests (cf. ci-contre) et de renvoyer le choix effectué.

```
Vous avez les possibilites suivantes :  
1 : Quitter  
2 : Ajouter un bebe  
3 : Afficher le dernier bebe enregistre  
4 : Statistiques annuelles  
5 : Compter un prenom  
Votre choix ?
```

10. Ecrire en langage C/C++ le programme principal qui affiche le menu à l'utilisateur tant qu'il souhaite poursuivre le test, et exécute le sous-programme approprié.