

Tp noté Langage C (ienac 15)

Consignes à respecter :

Vous devez pour ce TP noté vous connectez sous le login « enacaudio », mot de passe « enacaudio ».

Vous récupérez sous Ecampus Enac, cours « IP4000 », section « Séance 10 », le sujet du tp noté ainsi que les fichiers (« tp.h » et « tp_main.c »).

Vous créez dans le bureau un répertoire comportant votre nom et prénom dans lequel vous mettrez tous vos fichiers. A la fin, vous déposerez dans Ecampus le fichier demandé. N'effacez pas votre répertoire et vos fichiers à la fin de votre connexion (cela nous permettra de les récupérer sur votre machine en cas de problèmes suite à un mauvais dépôt).

Sujet :

On souhaite réaliser un programme qui gère des données sur des aéroports.

On utilisera un menu dans notre application qui permettra :

- de saisir dans un tableau les données d'un aéroport : son nom, son nombre de pistes et le nom de chacune des pistes.
- D'afficher les aéroports du tableau.
- De trier les aéroports du tableau (en fonction du nombre de pistes).
- De sauvegarder les données (nom, nombre de pistes nom de chaque piste ...) de chaque aéroport dans un fichier texte (un aéroport par ligne).

Vous avez à récupérer un fichier « tp_main.c » contenant le programme principal et un fichier d'en-tête « tp.h » contenant la définition des types et des déclarations de fonctions à développer.

Vous devez créer le fichier « tp.c » contenant les définitions des fonctions et le déposer sur le site Ecampus Enac à la fin.

1) Description du programme principal et des déclarations de fonctions:

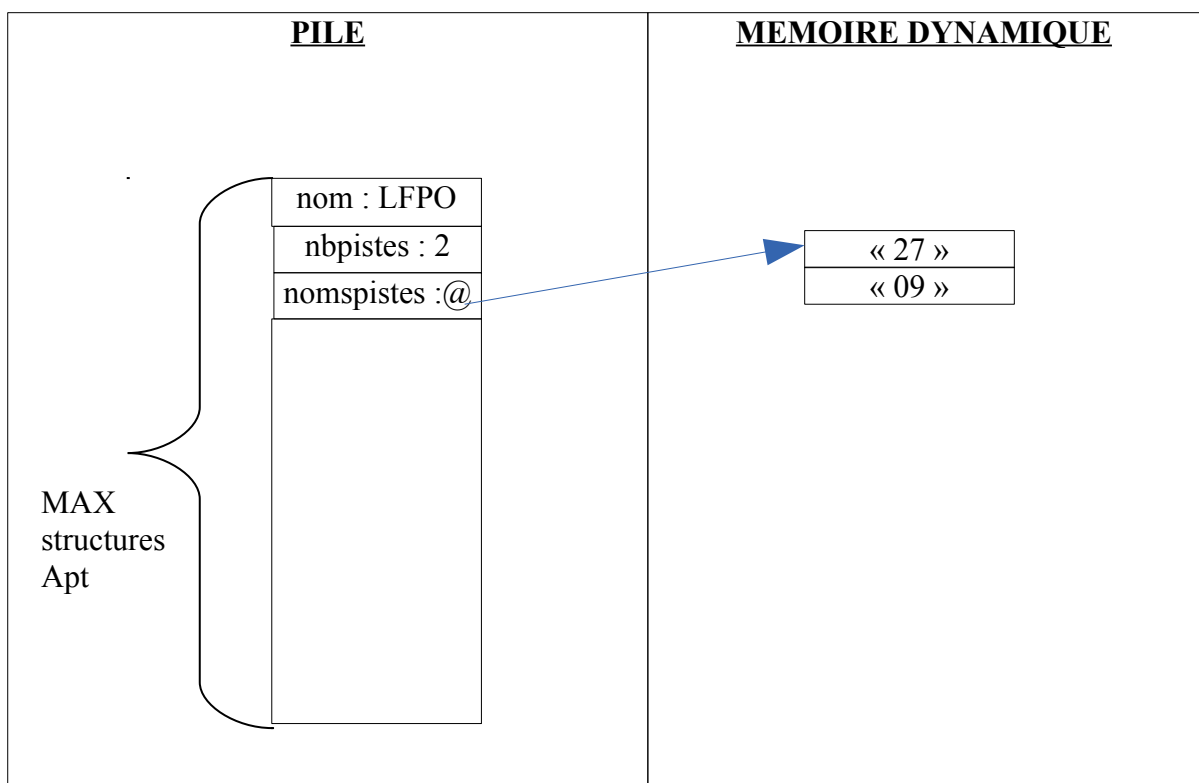
Le programme principal « main » vous est donné (fichier « tp_main.c ») ainsi que le fichier d'en-tête. Vous ne devez pas modifier les parties existantes.

2) Définitions des fonctions à développer :

Vous devez **créer le fichier « tp.c »** contenant les fonctions à définir et les réponses à ces questions :

- **a) Fonction « initApt » :** fonction qui initialise chaque case du tableau en affectant la valeur de l'expression INITAPT au nom de chaque aéroport.

- b) Fonction « saisirApt » :** fonction qui ajoute dans le tableau un nouvel aéroport : on demandera à l'utilisateur le nom, le nombre de pistes et le nom de chacune des pistes.
 Pour trouver la première case disponible du tableau, il suffit de rechercher celle qui a un nom égal à la valeur de l'expression INITAPT.
 Pour stocker tous les noms des pistes de ce nouvel aéroport, il faudra allouer dynamiquement le champ « nomspistes ».
 Par exemple, pour l'aéroport de nom « LFPO », de 2 pistes nommées « 27 » et « 09 », nous obtiendrons en mémoire la configuration suivante :



- c) Fonction « afficherApt » :** la fonction affiche les aéroports stockés dans le tableau : on affichera le nom, le nombre de pistes et le nom de chacune des pistes de chaque aéroport.
- d) Fonction « sauvegarderApt » :** la fonction sauvegarde dans un fichier texte chacun des aéroports stockés dans le tableau. On écrira les données d'un aéroport sur une ligne du fichier.
 Avec l'exemple précédent, on écrira la ligne suivante :
 LFPO 2 27 09

- **d) Fonction « trierApt » :** la fonction réalise un tri du tableau en fonction du nombre de pistes croissant. On pourra utiliser un tri par sélection (voir algo ci-dessous) sur le tableau en utilisant le champ « nbpistes » comme clé de tri.

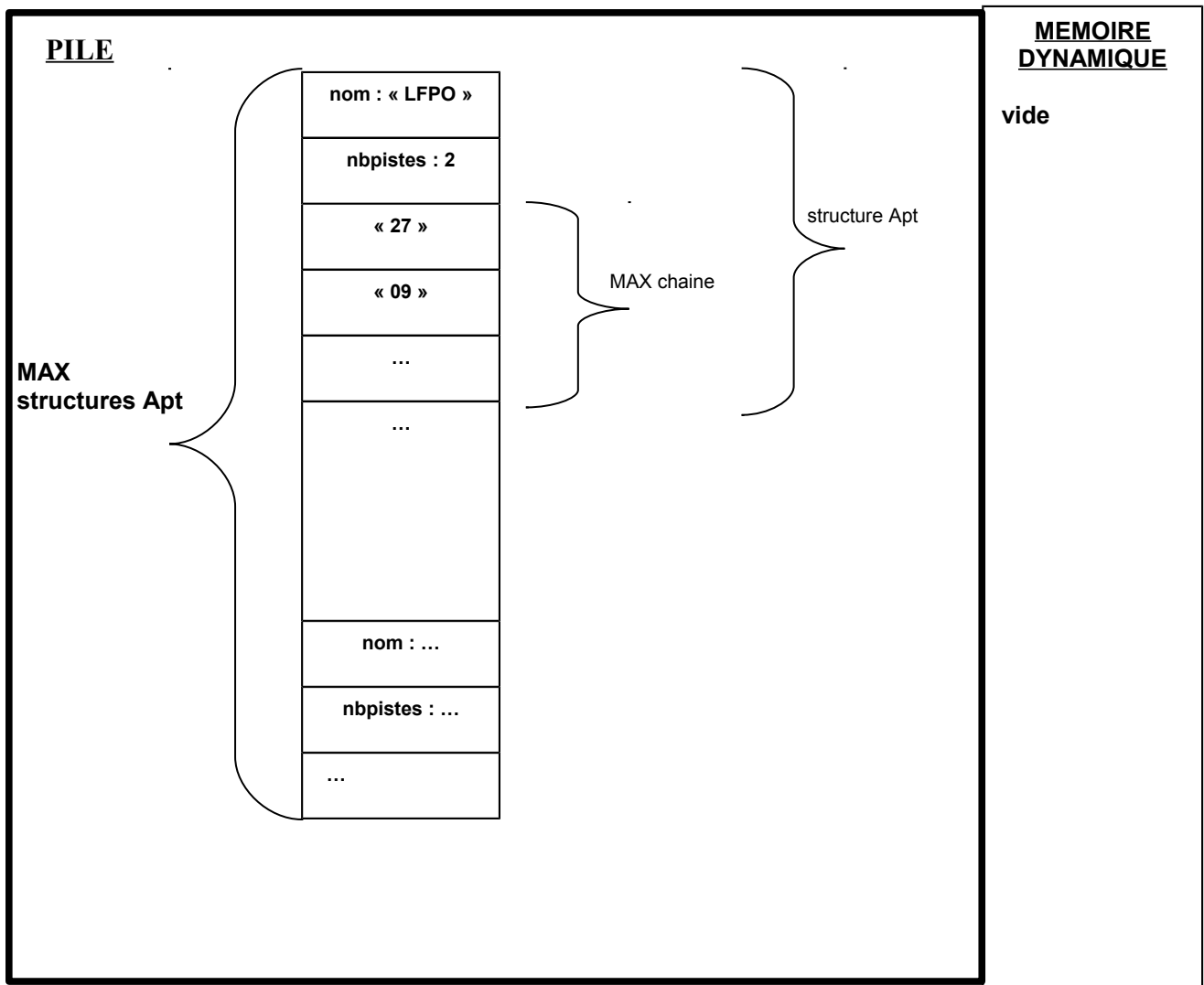
Pour un tableau de n éléments :

```

pour i de 0 à n-2
    min ← i
    pour j de i + 1 à n-1
        si t[j] < t[min] alors
            min ← j
    fin pour
    si min ≠ i alors échanger t[i] et t[min]
fin pour

```

- **e) Changement de type :** On souhaite modifier les types initiaux définis dans le fichier d'en-tête de façon à ce que le tableau du « main » ressemble en mémoire à la configuration correspondant au schéma ci-dessous.



Réécrire les types qui permettent d'obtenir cette configuration mémoire et donner l'instruction de déclaration de la variable tableau du « main » avec ces nouveaux types. Vous répondrez à cette question en ajoutant la réponse en commentaire à la fin de votre fichier.

Barème :

Question a) : 2 pts

Question b) : 6 pts

Question c) : 4 pts

Question d) : 6 pts

Question e) : 2 pts