

TP2 AP3

Le but de ce TP est de concevoir une file d'attente avec priorité (dénommée file dans la suite du sujet) permettant de représenter le classement des candidats à une formation. Aidez vous de ce que vous avez déjà vu en Cours/TD/TP et de ce qui est disponible sur moodle. Bien lire tout le sujet avant de commencer !

- 1) Définir un module représentant un candidat (**data.h** et **data.c**) et les fonctionnalités utiles à la manipulation de celui-ci.
 1. Définir le type individu, afin d'éviter les doublons, un individu aura (entre autre) un identifiant unique (de type entier)
 2. Écrire les fonctions de saisie d'un individu et d'affichage d'un individu
 3. Écrire une fonction qui ajoute un individu dans un fichier texte (le nom du fichier sera passé en paramètre de la fonction), vu en AP2...
- 2) Définir un module représentant une file modifiable (**file.h** et **file.c**)
 1. Définir le type file
 2. Définir l'organisation de vos données dans la file afin de minimiser les couts d'exploitation
 3. Écrire les fonctions
 1. Ajouter un individu à la file
 2. Tester la file vide
 3. Obtenir l'individu dont c'est le tour de sortir et le supprimer de la file
 4. Afficher tous les individus de la file
 5. Sauvegarder la file dans un fichier texte dont le nom sera passé en argument
- 3) Dans ce genre de candidature, il arrive parfois (même souvent) qu'un candidat se désiste avant la publication des résultats. Sans modifier les 2 précédents modules, écrire un module additionnel (**modadd.h** et **modadd.c**) permettant de
 1. Retirer de la file un individu en fonction de son identifiant.
 2. Prendre les n individus les mieux classés dans la file et en faire une nouvelle file
- 4) Écrire le module de simulation/vérification de vos modules précédents. Pour cela écrire un programme principal qui devra :
 1. Créer automatiquement 100 candidats, l'importance est mise sur l'identifiant (par exemple de 1 à 100) et la priorité de l'individu (inférieure ou égale à max_int) et sauvegarder la file dans un fichier
 2. Charger une file sauvegardée dans un fichier et :
 1. Proposer de supprimer un nombre n de candidats(n fourni par l'utilisateur), le choix des candidats se fera aléatoirement.
 2. Afficher la file des n candidats retenus pour s'inscrire à la formation (n étant demandé à l'utilisateur)
 3. Sauvegarder la file dans un fichier
- 5) Écrire un programme principal permettant l'utilisant concrète et réelle de la file (ajout manuel d'un candidat et retrait d'un candidat en fonction de son identifiant saisi par l'utilisateur)
- 6) Bonus : Modifier le module **data** (relatif à l'individu) de manière à manipuler une file de patients aux urgences ainsi que le programme principal de simulation, on supposera cette fois que le patient à retirer de la file a dû être soigné immédiatement sans attendre car son

état de santé s'était dégradé.

Comme toujours, faites valider par l'enseignant vos choix de structures et vos algorithmes.
Commentez utilement votre code.

Bon courage.