Travail pratique II

Date de remise : 23 juillet 2007

Louis

Table of Contents

1 Introduction

Dans ce document, vous trouverez le pseudo-code des principaux algorithmes du logiciel. Pour une description détaillée des classes et méthodes, la documentation sera générée par javadoc.

2 Description

Logiciel en Java pour gérer les informations des membres du personnel du Centre communautaire la bonne oeuvre. Le personnel est composé de bénévoles et d'employés.

Il y a quelques ajouts depuis la première version :

- Les bénévoles ont un pourcentage d'acceptation à leur dossier. Lorsqu'on demande la liste des bénévoles disponibles pour une journée, les bénévoles ayant un taux d'acceptation supérieur seront en tête de liste.
- On peut maintenant modifier les informations d'un membre du personnel sauf les taux d'acceptation.

Pour démarrer le logiciel s'il est déjà compilé, il suffit de taper "java Tp1" à l'invite de commandes. S'il n'est pas compilé, vous devez le compiler d'abord en entrant "javac *.java" à l'invite. Pour les deux commandes, vous devez être dans le répertoire où se trouve les fichiers source .java du programme. Toute la documentation est dans le répertoire docs.

3 Variables

Voici une description des principales variables du programme :

nom nom du membre du personnel. Chaîne non vide.

prenom Chaîne non vide. Prénom du membre du personnel

adresse Chaîne non vide. Adresse du membre du personnel.

telephone Chaîne non vide et respectant le format 111-1111. Numéro de télépone du membre du personnel.

disponibilitesSemaine
Tableau de booléens. Tableau de sept éléments contenant les disponibilités hebdomadaires d'un bénévole. Chaque élément correspond à un jour de la semaine.

salaire Réel entre 8.00 et 25.00 inclusivement. Salaire d'un employé

nbRefus Entier. Nombre de fois qu'un bénévole a refuser une offre de bénévolat.

nbOffres Entier. Nombre de fois qu'un bénévole a reçu une offre de bénévolat.

4 Algorithmes

4.1 Algorithmes principaux

On aura besoin de créer quatres objets de la classe RepertoireListeTriee dans la fonction main. Une liste de membres triée par numéros de téléphone, une liste de membre triée par noms, une liste de bénévoles triée par poucentage d'acceptation et une liste triée par prénom que j'ai choisi de créer parce que parfois j'afficherai en ordre de prénoms et parfois en ordre de noms.

Les bénévoles n'ayant eu aucune offre de bénévolat se verront attribué un poucentage d'acceptation de 80%. Ceci sera géré dans l'accesseur getTauxAccepation() de la classe Benevole. Une fois que le bénévole accepte ou refuse une offre, le taux commence à être calculé avec les valeurs nbRefus/nbOffres.

On doit faire attention aux effets de bord car lorsqu'on modifie un membre du personnel, son ordre de tri dans les listes change. La solution est de retirer le membre de la liste et de le rajouter; de cette façon, le membre est rajouté au bon endroit dans les listes.

4.2 Algorithmes de RepertoireListeTriee.java

4.2.1 Fonction ajouter

```
DÉBUT
  insere<-false
  TANTQUE l'élément n'a pas été inséré
    SI suivant==fin OU nouveau<suivant OU nouveau==suivant ALORS
      SI  le suivant c'est la tete (c'est la première itération) ALORS
        tete=nouveau
      SINON
        suivant.getPrecedent().setSuivant(nouveau);
      nouveau.setSuivant(suivant);
      insere<-true
    SINON
      suivant <- next
    FINSI
  RÉPÉTER

  SI on a atteint la fin de la liste et que l'élément n'a pas été inséré
  ALORS
    SI suivant == tete ALORS
      tete = nouveau
    SINON
      suivant.getPrecedent().setSuivant(nouveau);
    FINSI
    nouveau.setSuivant(fin);
  FINSI
FIN
```

4.2.2 Fonction retirer

```
DEBUT
  suivant<-tete
  RÉPÉTER
    SI suivant == aRetirer ALORS
      SI suivant==tete ALORS
        tete = suivant.getSuivant()
      SINON
        suivant.getPrecedent().setSuivant(suivant.getSuivant())
    FINSI
    suivant = suivant.getSuivant()
  TANTQUE suivant != fin
FIN
```

5 Stratégie de vérification

Voici la statégie de vérification pour ce programme. Le fichier trace.txt contient l'application de cette statégie.

1. Ajouter un employé
   1. Entrer un nom vide.
   2. Entrer un prénom vide.
   3. Entrer une adresse vide.
   4. Entrer un numéro de téléphone avec un mauvais format.
   5. Entrer un salaire en dehors des limites permises.
2. Ajouter un bénévole
   1. Entrer un nom vide.
   2. Entrer un prénom vide.
   3. Entrer une adresse vide.
   4. Entrer un numéro de téléphone avec un mauvais format.
   5. Entrer des disponibilités autres que 'o' ou 'n'.
3. Ajouter deux autres employés et deux autres bénévoles mais sans effectuer les tests détaillés comme précédemment. Deux des trois bénévoles doivent avoir les mêmes disponibilités.
4. Lister les membres du personnel.
5. Lister les bénévoles disponibles pour dimanche, mercredi et samedi et comparer avec les valeurs entrées précédemment.
6. Retirer un employé.
7. Retirer un bénévole.
8. Lister les membres.
9. Lister les bénévoles disponibles pour dimanche, mercredi et samedi.
10. Retirer tous les membres.

11. Lister tous les membres.
12. Lister les bénévoles disponibles pour dimanche, mercredi et samedi.
13. Essayer de retirer un bénévole et un employé même s'il n'y en a aucun.
14. Quitter