

CHOIX TECHNIQUES DU PROJET INFORMATIQUE DU GROUPE 7

Felix Charvin, Alexandre De Boé, Florian Da Silva, Tom Taranto

15/04/2016

Structures de données

```
1 typedef struct listePersonnes_base* listePersonnes;
```

ListePersonnes est une liste triée par priorité décroissante, afin de pouvoir accéder instantanément à la personne de plus haute priorité et pouvoir ainsi lui affecter son voyage en priorité. ListePersonnes contient les informations : un tableau (nom, prénom, choix1, choix2, 6 planètes choix1, 6 planètes choix2, priorité)

```
1 typedef struct dictPriorites_base* dictPriorites;
```

dictPriorites est une structure intermédiaire qui sert à remplir la liste ListePersonnes. La clé est (nom,prénom) et la valeur associée est la priorité de la personne correspondante. On a choisi un dictionnaire car de nombreuses recherches vont avoir lieu dedans.

```
1 typedef struct dictContraintes_base* dictContraintes;
```

dictContraintes est un dictionnaire permettant de rechercher les contraintes associées à une certaine planète. Une clé est le nom d'une planète, et la valeur associée est : (la zone associée à la planète clé, la planète contrainte, et la zone associée à la planète contrainte). On a choisi un dictionnaire car de nombreuses recherches vont avoir lieu dedans.

```
1 typedef struct listeFinale_base* listeFinale;
```

listeFinale est une liste triée par ordre lexicographique sur le tuple (nom, prénom) des voyageurs. Elle contient : un tableau (nom, prénom,choix retenu, les 6 planètes du choix retenu). Elle sert à l'exportation vers le fichier .csv . On a fait ce choix pour faciliter la mise en page du fichier .csv final.

```
1 typedef struct dictDestinations_base* dictDestinations;
```

dictDestination est un dictionnaire dont les clés sont les planètes et la valeur associée à chaque clé est le nombre de places restantes pour chaque planète. On a choisi cette structure car de

nombreuses recherches vont avoir lieu dedans.

Algorithme d'affectation

Result: Renvoie la liste des voyageurs avec le voyage sur lequel ils sont affectés

Créer quatres listes vides;

```
for i parcourant la liste des voyageurs do
    if le choix1 du voyageur i est organisé then
        | voyageur i va dans la liste1;
    else
        | voyageur i va dans la liste2;
    end
end
for i parcourant liste1 do
    if il reste des places pour le choix1 then
        | affecter le voyageur i a son choix1 dans la liste resultat;
    else
        if si son choix2 == libre then
            | insérer voyageur i dans liste2 avec la bonne priorité;
            | son choix1 ← choix2
        else
            if il reste des places pour son choix2 then
                | affecter le voyageur i a son choix2 dans la liste resultat;
            end
            | insérer voyageur i dans liste3;
        end
    end
end
end
```

```

for i parcourant liste2 do
    if le choix1 du voyageur i respecte les contraintes then
        if Il reste des places disponibles pour son choix1 then
            affecter le voyageur i a son choix1 dans la liste resultat;
        else
            if choix2 != libre then
                if si il reste des places pour son choix2 then
                    affecter le voyageur i a son choix2 dans la liste resultat;
                else
                    insérer voyageur i dans liste3;
                end
            else
                insérer voyageur i dans liste3;
            end
        end
    else
        if choix2==libre then
            if si il reste des place pour son choix2 then
                affecter le voyageur i a son choix2 dans la liste resultat;
            else
                insérer voyageur i dans liste3
            end
        else
            insérer voyageur i dans liste3
        end
    end
end

```

reverser liste3;

```

for i parcourant liste3 do
    trouver une croisière pour le voyageur i satisfaisante par rapport à son choix1;
    affecter le voyageur i à cette croisière dans la liste resultat;
end

```

retourner le fichier .csv correspondant à la liste resultat;

Découpage en module

Il a été décidé de découper le programme en 3 modules. Le premier module s'occupe de l'importation et de l'exportation des données .csv en les mettant dans les bonnes structures de données.

Le second module va implémenter les listes et toutes les structures nécessaires au bon fonc-

tionnement du programme.

Le dernier module permet l'affectation de personnes à des croisières.

Une fonction main et un makefile seront aussi disponibles.

Choix du langage

Nous avons choisi le langage C pour résoudre ce problème. En effet, nous ne nous intéressons pas à l'aspect "objet" de ce projet. De plus, de nombreuses structures de données sont utilisées et le langage C nous permet de toutes les définir et de les utiliser facilement. Par ailleurs, le langage C est le langage que nous maîtrisons le mieux.