

Examen – Session 2 (2011-2012)

Dans tous les cas, une solution récursive devra être sous forme terminale.

Exercice 1

On considère la fonction **f** définie récursivement par :

```
let rec f x = if x = 0 then 2 else 3 * f (x-1) - 1
```

Ecrire une nouvelle version **ft** de cette fonction sous forme récursive terminale.

Exercice 2

Des notes sont données sous forme d'une des 5 valeurs symboliques traduisant qu'un travail est très bien, bien, passable, insuffisant, très insuffisant.

1°/ Définir un type énuméré **t_symbol** qui traduit cette information.

2°/ Ecrire une fonction **reel_of_symbol** qui traduit une valeur symbolique en note numérique réelle. On associera 18 à très bien, 14 à bien, 10 à passable, 6 à insuffisant, 2 à très insuffisant. Ecrire une fonction **superieure** booléenne qui vérifie, pour une note donnée de type **t_symbol**, qu'elle correspond à une note supérieure ou égale à une valeur réelle **x** donnée.

3°/ Ecrire une fonction **symbol_of_reel** qui indique, pour une note numérique réelle, la valeur symbolique correspondante. Chaque valeur symbolique est associée à un segment d'amplitude 4 dont la valeur représentative est le centre. [20 , 16] ;]16 , 12] ;]12 , 8] ;]8 , 4] ;]4 , 0] . On utilisera la division de la note par 4.

Un étudiant est caractérisé par un enregistrement à deux champs, l'un indique le nom de l'étudiant, l'autre indique sa note de type **t_symbol**.

Un groupe est caractérisé par une liste d'étudiants.

Une promotion est caractérisée par une liste de groupes

4°/ Définir un type enregistrement **t_etudiant** qui traduit l'information associée à un étudiant.

5°/ Ecrire une fonction **present** booléenne qui teste si un étudiant donné par son nom est dans un groupe.

6°/ Ecrire une fonction **admis_gr** dont le résultat est le nombre d'étudiants d'un groupe donné qui ont obtenu une note symbolique très bien, bien ou passable.

7°/ Ecrire une fonction **moyenne** qui établit, pour un groupe donné, la moyenne numérique réelle associée à ce groupe.

8°/ On suppose que la liste des étudiants dans un groupe est classée par ordre alphabétique des noms. Ecrire une fonction **liste_admis** dont l'argument est un groupe et dont le résultat est la liste des noms des étudiants ayant obtenu très bien, bien ou passable, la liste devra respecter l'ordre alphabétique initial.

9°/ Ecrire une fonction **meilleure** dont l'argument est une promotion et qui a pour résultat la meilleure moyenne numérique obtenue par un groupe.

10°/ Ecrire une fonction **admis_promo** qui indique, pour une promotion donnée, le nombre d'étudiants ayant obtenu une note symbolique très bien, bien ou passable.