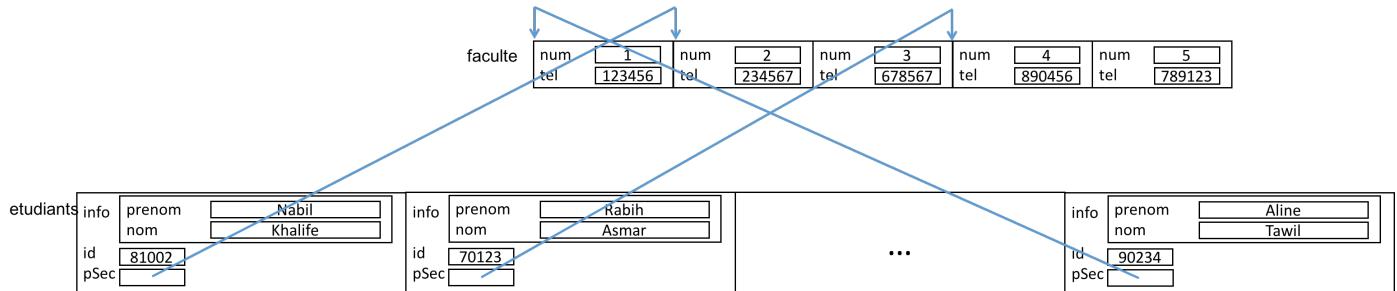


Exercice I (11 points)

On considère la représentation suivante pour modéliser les inscriptions des étudiants dans les différentes sections de la Faculté :



1. Définir un type de structure **Info** contenant deux champs: prenom et nom.
2. Définir un type de structure **Section** contenant deux champs: num (int) et tel (int).
3. Définir un type de structure **Etudiant** contenant trois champs: info (Info), id (int) et un pointeur de type de base Section.
4. Écrire la fonction «trouver» qui, à partir d'un numéro de section, un nom et un prénom d'étudiant, un tableau d'étudiants, renvoie le nombre d'étudiants inscrits dans cette section portant ce nom complet.
5. Écrire la fonction «transferer» qui, étant donné le tableau des sections, un tableau d'étudiants, un identificateur d'étudiant (id) et un numéro de section, modifie les données correspondantes de l'étudiant pour qu'il soit inscrit à la section dont le numéro est donné. À noter qu'on ne peut pas modifier le champ num de l'instance de la section car il y aurait d'autres étudiants inscrits à cette section qui ne veulent pas subir un transfert.

Exercice II (7 points)

Écrire la fonction **réursive** "printRWords" qui, étant donnée une chaîne de caractères, affiche les mots de cette chaîne dans l'ordre inverse. Exemple: chaîne donnée : I am writing an email → la fonction affiche sur l'écran: email an writing am I

Exercice III (4 points)

Dessiner l'état de la mémoire du programme suivant.

```
void m(int *t, int *pt[]){
    *pt[-1]=*t*2;
}
void f(int *a, int *pa[]){
    *pa=a+1;
    m(a+2, pa-1);
}
void main(){
    int t[]={1,3,5,7,9,8,6,4}, * pt[8], i;
    for(i=0; i<8; i++)
        *(pt+i)=t+(i+5)%8;
    f(t+1, pt+3);
}
```