
Tableaux, Enregistrements et Fonctions

Exercice 1. Soit une société qui a à gérer des informations sur son personnel. Elle emploie au maximum 100 personnes. Pour chaque employé, elle enregistre le nom, le prénom, l'adresse, le sexe et, s'il s'agit d'un homme, la situation militaire (libéré, exempté, réformé ou incorporable), s'il s'agit d'une femme, son nom de jeune fille. Une adresse contient un numéro, un nom de rue, un code postal et un nom de localité. Un nom ne dépasse jamais 30 caractères. Proposez les structures de données nécessaires pour représenter ces informations.

Exercice 2. En utilisant les structures de données définies dans l'exercice 1, écrire un programme en C permettant de saisir des informations sur le personnel d'une société de taille (le nombre des employés) choisie par l'utilisateur.

Exercice 3. Modifier le programme de l'exercice 2 pour qu'il présente un menu initial (utiliser l'instruction `switch`) permettant de : i) Saisir les informations sur le personnel (exercice 2) ou ii) Consulter les données d'un / d'une employé(e) en donnant l'indice lui correspondant dans le tableau qui représente la société.

Exercice 4. Modifier le programme de l'exercice 3, en permettant de consulter les données d'un/une employé(e) à partir de son nom de famille.

Exercice 5. Modifier le choix 1 du menu dans le programme de l'exercice 4 (Saisir les informations sur le personnel), en permettant d'ajouter un/une employé(e) à la société, tant qu'il y a de la place disponible.

Exercice pour le TP. Coder le programme de l'exercice 5 en utilisant trois fonctions pour : i) coder le menu initial ; ii) ajouter un/une employé(e), et iii) consulter les données d'un/une employé(e) à partir de son nom de famille.

Corrigée

Exercice 1. Voici une solution en utilisant la définition des nouveaux types (pages 2–3 du polycopie Troisième Partie) et les enregistrements (pages 4–8). Etant donné qu'un nom ne dépasse jamais 30 caractères, on peut définir le type `typeNom` comme un tableau de 30 caractères.

```
typedef char typeNom[30];
```

On peut utiliser les énumérations (page 3) pour définir les types : i) `typeSexe` comme l'énumération de constantes symboliques `feminin` et `masculin`, et ii) `typeSitMil` (situation militaire) comme l'énumération de constantes symboliques `libéré`, `exempté`, `réformé` et `incorporable`.

```
typedef enum {
    feminin;
    masculin;
} typeSexe;
```

```
typedef enum {
    libere;
    exempte;
    reforme;
    incorporable;
} typeSitMil;
```

On peut définir le type `typeAdresse` comme un type d'enregistrement :

```
typedef struct {
    unsigned numero;
    typeNom rue;
    unsigned code;
    typeNom localite;
} typeAdresse;
```

Pour définir l'employé, on peut utiliser un type enregistrement appelé `typeEmploye`, incluant des champs nécessaires pour l'information complémentaire (situation militaire et nom de jeune fille).

```
typedef struct {
    typeNom nom;
    typeNom prenom;
    typeAdresse adresse;
    typeSexe sexe;
    typeSitMil sitMil;
    typeNom nomJF;
} typeEmploye;
```

Evidemment, l'information complémentaire peut être défini comme une type union `typeInfoComp`, mais les unions ne sont pas abordées dans le cours. Si vous le considère pertinent, voici la définition.

```
typedef union {
    typeSitMil sitMil;
    typeNom nomJF;
} typeInfoComp;
```

Indiquez aux étudiants que tous les champs d'une union sont implantés à la même adresse de memoire, ayant comme résultat, pour le cas `typeInfoComp`, qu'une variable de ce type et composé de un membre `SitMil` ou (exclusive) d'une membre `nomJF` ! Si on utilise l'union, la définition de `typeEmploye` est :

```

typedef struct {
    typeNom      nom;
    typeNom      prenom;
    typeAdresse  adresse;
    typeSexe     sexe;
    typeInfoComp complement;
} typeEmploye;

```

La société est définie comme un type tableau des employés de taille 100.

```

typedef typeEmploye typeSociete[100];

```

EXERCICE 2 : Le code :

```

#include<stdio.h>

#define TAILLEMAXNOM 30
#define TAILLEMAXENTREPRISE 100

int main() {

    typedef char typeNom[TAILLEMAXNOM];

    typedef enum {
        feminin,
        masculin
    } typeSexe;

    typedef enum {
        libere,
        exempte,
        reforme,
        incorporable
    } typeSitMil;

    typedef struct {
        unsigned numero;
        typeNom  rue;
        unsigned code;
        typeNom  localite;
    } typeAdresse;

    typedef union {
        typeSitMil sitMil;
        typeNom    nomJF;
    } typeInfoComp;

    typedef struct {
        typeNom      nom;
        typeNom      prenom;
        typeAdresse  adresse;
        typeSexe     sexe;
        typeInfoComp complement;
    } typeEmploye;

    typedef typeEmploye typeSociete[TAILLEMAXENTREPRISE];

    typeSociete monSociete;
    int          tailleMonSociete;
    int          i;
    int          respSex, respSitMil;

```

```

do{
    printf("Nombre des employes dans votre societe (max 100) : ");
    scanf("%d",&tailleMonSociete);
} while (tailleMonSociete < 1 || tailleMonSociete >100);

printf("\nCapture Information du Personnel \n");

for(i=0; i<tailleMonSociete; i++) {

    printf("\nDonnees pour l'employe %d. \n", i+1);

    printf("\tNom : ");
    scanf("%s", &monSociete[i].nom);

    /* Rappelez aux etudiants que scanf avec la chaine de
       control "%s", s'arrete au premier space rencontre.
       Dans cet exercice je utilise '_' a la place des
       spaces, i.e., "Alphonse Daudet" devient
       "Alphonse_Daudet" On peut utiliser aussi fflush et
       gets, i.e. fflush(stdin); gets(monSociete[i].prenom); */

    printf("\tPrenom : ");
    scanf("%s", &monSociete[i].prenom);

    printf("\tAdresse : \n");

    printf("\t\tNumero : ");
    scanf("%d", &monSociete[i].adresse.numero);

    printf("\t\tRue : ");
    scanf("%s", &monSociete[i].adresse.rue);

    printf("\t\tCode : ");
    scanf("%d", &monSociete[i].adresse.code);

    printf("\t\tLocalite : ");
    scanf("%s", &monSociete[i].adresse.localite);

    do{
        printf("\tSexe \t[1]feminin / [2]masculin : ");
        scanf("%d", &respSex);
    } while (respSex < 1 || respSex > 2);

    switch (respSex) {
        case 1 :
            monSociete[i].sexe = feminin;
            printf("\tNom de Jeune Fille : ");
            scanf("%s", &monSociete[i].complement.nomJF);
            break;
        case 2 :
            monSociete[i].sexe = masculin;

            do{
                printf("\tSituation Militaire \t[1]libere / [2]exempte / \n
                    \t\t\t\t\t[3]reforme / [4]incorporable : ");
                scanf("%d", &respSitMil);
            } while (respSitMil < 1 || respSitMil > 4);

            switch (respSitMil) {
                case '1' :
                    monSociete[i].complement.sitMil = libere;
                    break;
            }
        }
    }
}

```

```

        case 'e' :
            monSociete[i].complement.sitMil = exempte;
            break;
        case 'r' :
            monSociete[i].complement.sitMil = reforme;
            break;
        case 'i' :
            monSociete[i].complement.sitMil = incorporable;
            break;
    }
    break;
}
}

printf("\nInformation du personnel saisi\n");

return(0);
}

```

Et un exemple de son exécution :

Nombre des employes dans votre societe (max 100) : 2

Capture Information du Personnel

Donnees pour l'employe 1.

```

Nom : Julio
Prenom : Cortazar
Adresse :
    Numero : 6
    Rue : Alphonse_Daudet
    Code : 75014
    Localite : Paris
Sexe [1]feminin / [2]masculin : 2
Situation Militaire [1]libere / [2]exempte /
    [3]reforme / [4]incorporable : 2

```

Donnees pour l'employe 2.

```

Nom : Louise
Prenom : Cortazar
Adresse :
    Numero : 6
    Rue : Alphonse_Daudet
    Code : 75014
    Localite : Paris
Sexe [1]feminin / [2]masculin : 1

```

Nom de Jeune Fille : Brooks

Information du personnel saisi

EXERCICE 3 : Le code :

```

#include<stdio.h>

#define TAILLEMAXNOM 30
#define TAILLEMAXENTREPRISE 100
int main() {

    /* ajouter la definition des types idem a l'exercice 2 */

```

```

typeSociete monSociete;
int         tailleMonSociete;
int         i;
int         respMenu; /* le response aux choix du Menu*/
int         respSex, respSitMil, respEmp; /* autres responses */

do{
do {

/* Menu Principal */

printf("\n\n*** System du Personnel *** \n\n");
printf("\t[1] Saisir les donnees du personnel. \n");
printf("\t[2] Consulter les donnees du personnel. \n");
printf("\t[3] Quitter the system. \n\n");
printf("\t    Tapper votre choix : ");

scanf("%d",&respMenu);

} while ((respMenu < 1) || (respMenu > 3));

switch (respMenu) {
case (1) :
printf("\n** Saisie Information du Personnel ** \n\n");

do{
printf("\tNombre des employes dans votre societe (max 100) : ");
scanf("%d",&tailleMonSociete);
} while (tailleMonSociete < 1 || tailleMonSociete >100);

for(i=0; i<tailleMonSociete; i++) {

printf("\n\tDonnees pour l'employe %d. \n", i+1);

printf("\tNom : ");
scanf("%s", &monSociete[i].nom);

/* Rappelez aux etudiants que scanf avec la chaine de
control "%s", s'arrete au premier space rencontre.
Dans cet exercice je utilise '_' a la place des
spaces, i.e., "Alphonse Daudet" devient
"Alphonse_Daudet" */

printf("\tPrenom : ");
scanf("%s", &monSociete[i].prenom);

printf("\tAdresse : \n");

printf("\t\tNumero : ");
scanf("%d", &monSociete[i].adresse.numero);

printf("\t\tRue : ");
scanf("%s", &monSociete[i].adresse.rue);

printf("\t\tCode : ");
scanf("%d", &monSociete[i].adresse.code);

printf("\t\tLocalite : ");
scanf("%s", &monSociete[i].adresse.localite);

do{
printf("\tSexe \t[1]feminin / [2]masculin : ");

```

```

scanf("%d", &respSex);
} while (respSex < 1 || respSex > 2);

switch (respSex) {
case (1) :
    monSociete[i].sexe = feminin;
    printf("\tNom de Jeune Fille : ");
    scanf("%s", &monSociete[i].complement.nomJF);
    break;
case (2) :
    monSociete[i].sexe = masculin;

do{
    printf("\tSituation Militaire \t[1]libere / [2]exempte /
        \n\t\t\t\t[3]reforme / [4]incorporable : ");
    scanf("%d", &respSitMil);
} while (respSitMil < 1 || respSitMil > 4);

switch (respSitMil) {
case 'l' :
    monSociete[i].complement.sitMil = libere;
    break;
case 'e' :
    monSociete[i].complement.sitMil = exempte;
    break;
case 'r' :
    monSociete[i].complement.sitMil = reforme;
    break;
case 'i' :
    monSociete[i].complement.sitMil = incorporable;
    break;
}
break;
}
}

printf("\n** Information du personnel saisi **\n");
break;
case (2) :
    printf("\n** Consulte Information du Personnel ** \n\n");

do{
    printf("\tNombre de l'employe a consulter (1...%d) : ",
        tailleMonSociete);
    scanf("%d",&respEmp);
} while (respEmp < 1 || respEmp > tailleMonSociete);

printf("\n\tEmploye No : %d\n\n", respEmp);
respEmp--;
printf("\tNom      : %s\n", monSociete[respEmp].nom);
printf("\tPrenom   : %s\n", monSociete[respEmp].prenom);
if (monSociete[respEmp].sexe == feminin)
    printf("\tNom JF : %s\n",
        monSociete[respEmp].complement.nomJF);
printf("\tAdress  : %d, %s, %d, %s\n",
    monSociete[respEmp].adresse.numero,
    monSociete[respEmp].adresse.rue,
    monSociete[respEmp].adresse.code,
    monSociete[respEmp].adresse.localite);

/* Rappelez aux etudiants que pour le types enumeration
    les constantes symboliques dans le type, s'identifient

```

```

    a la liste de premier entiers naturels, donc pour sexe,
    feminin = 0 et masculin = 1 *\

switch (monSociete[respEmp].sexe){
    case (0) : printf("\tSex      : feminin\n"); break;
    case (1) : printf("\tSex      : masculin\n"); break;
}

if (monSociete[respEmp].sexe == masculin)
    switch (monSociete[respEmp].complement.sitMil){
        case (0) : printf("\tMili      : libere\n"); break;
        case (1) : printf("\tMili      : exempte\n"); break;
        case (2) : printf("\tMili      : reforme\n"); break;
        case (4) : printf("\tMili      : incorporable\n");
    }
    break;

case (3) :
    printf("A bientot.\n\n");
    return(0);
}
}while(respMenu != 3);
}

```

Et un exemple de son exécution :

```
*** System du Personnel ***
```

```

[1] Saisir les donnees du personnel.
[2] Consulter les donnees du personnel.
[3] Quitter the system.

```

```
    Tapper votre choix : 1
```

```
** Saisie Information du Personnel **
```

```
    Nombre des employes dans votre societe (max 100) : 2
```

```
    Donnees pour l'employe 1.
```

```
    Nom : Louise
```

```
    Prenom : Brooks
```

```
    Adresse :
```

```
        Numero : 5
```

```
        Rue : Edgar_Quinet
```

```
        Code : 75014
```

```
        Localite : Paris
```

```
    Sexe      [1]feminin / [2]masculin : 1
```

```
    Nom de Jeune Fille : Schmidt
```

```
    Donnees pour l'employe 2.
```

```
    Nom : Fuentes
```

```
    Prenom : Carlos
```

```
    Adresse :
```

```
        Numero : 3
```

```
        Rue : Victoire
```

```
        Code : 75001
```

```
        Localite : Paris
```

```
    Sexe      [1]feminin / [2]masculin : 2
```

```
    Situation Militaire      [1]libere / [2]exempte /
```

```
                            [3]reforme / [4]incorporable : 2
```

```
** Information du personnel saisi **
```


*** System du Personnel ***

- [1] Saisir les donnees du personnel.
- [2] Consulter les donnees du personnel.
- [3] Quitter the system.

Tapper votre choix : 2

** Consulte Information du Personnel **

Nombre de l'employe a consulter (1...2) : 1

Employe No : 1

Nom : Louise
Prenom : Brooks
Nom JF : Schmidt
Adress : 5, Edgar_Quinet, 75014, Paris
Sex : feminin

*** System du Personnel ***

- [1] Saisir les donnees du personnel.
- [2] Consulter les donnees du personnel.
- [3] Quitter the system.

Tapper votre choix : 2

** Consulte Information du Personnel **

Nombre de l'employe a consulter (1...2) : 2

Employe No : 2

Nom : Fuentes
Prenom : Carlos
Adress : 3, Victoire, 75001, Paris
Sex : masculin
Mili : libere

*** System du Personnel ***

- [1] Saisir les donnees du personnel.
- [2] Consulter les donnees du personnel.
- [3] Quitter the system.

Tapper votre choix : 3

** A bientot **

EXERCICE 4 : Le code :

```
#include<stdio.h>
#include<string.h> /* Pour comparer des chaines de caracteres */

#define TAILLEMAXNOM 30
#define TAILLEMAXENTREPRISE 100
```

```

#define VRAI 1
#define FAUX 0

int main() {

    /* ajouter la definition des types idem a l'exercice 2 */

    typeSociete monSociete;
    int         tailleMonSociete;
    int         i;
    int         respMenu; /* le response aux choix du Menu*/
    int         respSex, respSitMil, respEmp; /* autres responses */
    typeNom     nomEmp; /* nom de l'employe a consulter */
    int         trouve; /* TRUE si on a trouve l'employe */

    do{
        do {

            /* Menu Principal */

            printf("\n\n*** System du Personnel *** \n\n");
            printf("\t[1] Saisir les donnees du personnel. \n");
            printf("\t[2] Consulter les donnees du personnel. \n");
            printf("\t[3] Quitter the system. \n\n");
            printf("\t    Tapper votre choix : ");

            scanf("%d",&respMenu);

        } while ((respMenu < 1) || (respMenu > 3));

        switch (respMenu) {
            case (1) :
                printf("\n** Saisie Information du Personnel ** \n\n");

                do{
                    printf("\tNombre des employes dans votre societe (max 100) : ");
                    scanf("%d",&tailleMonSociete);
                } while (tailleMonSociete < 1 || tailleMonSociete >100);

                for(i=0; i<tailleMonSociete; i++) {

                    printf("\n\tDonnees pour l'employe %d. \n", i+1);

                    printf("\tNom : ");
                    scanf("%s", &monSociete[i].nom);

                    /* Rappelez aux etudiants que scanf avec la chaine de
                     control "%s", s'arrete au premier space rencontre.
                     Dans cet exercice je utilise '_' a la place des
                     spaces, i.e., "Alphonse Daudet" devient
                     "Alphonse_Daudet" */

                    printf("\tPrenom : ");
                    scanf("%s", &monSociete[i].prenom);

                    printf("\tAdresse : \n");

                    printf("\t\tNumero : ");
                    scanf("%d", &monSociete[i].adresse.numero);

                    printf("\t\tRue : ");
                    scanf("%s", &monSociete[i].adresse.rue);
                }
            }
        }
    }
}

```

```

printf("\t\tCode : ");
scanf("%d", &monSociete[i].adresse.code);

printf("\t\tLocalite : ");
scanf("%s", &monSociete[i].adresse.localite);

do{
    printf("\tSexe \t[1]feminin / [2]masculin : ");
    scanf("%d", &respSex);
} while (respSex < 1 || respSex > 2);

switch (respSex) {
case (1) :
    monSociete[i].sexe = feminin;
    printf("\tNom de Jeune Fille : ");
    scanf("%s", &monSociete[i].complement.nomJF);
    break;
case (2) :
    monSociete[i].sexe = masculin;

    do{
        printf("\tSituation Militaire \t[1]libere / [2]exempte /
        \n\t\t\t\t[3]reforme / [4]incorporable : ");
        scanf("%d", &respSitMil);
    } while (respSitMil < 1 || respSitMil > 4);

    switch (respSitMil) {
        case (1) :
            monSociete[i].complement.sitMil = libere;
            break;
        case (2) :
            monSociete[i].complement.sitMil = exempte;
            break;
        case (3) :
            monSociete[i].complement.sitMil = reforme;
            break;
        case (4) :
            monSociete[i].complement.sitMil = incorporable;
            break;
    }
    break;
}
}

printf("\n** Information du personnel saisi **\n");
break;

case (2) :
    printf("\n** Consulte Information du Personnel ** \n\n");
    printf("\tNom de l'employe a consulter : ");
    scanf("%s",&nomEmp);

    trouve = FAUX;
    for (i=0; i<tailleMonSociete; i++) {
        if (!strcmp(nomEmp, monSociete[i].nom)) {
            trouve = VRAI;
            printf("\n\tEmploye No : %d\n\n", i+1);
            printf("\tNom      : %s\n", monSociete[i].nom);
            printf("\tPrenom : %s\n", monSociete[i].prenom);

            if (monSociete[i].sexe == feminin)

```

```

    printf("\tNom JF : %s\n", monSociete[i].complement.nomJF);

printf("\tAdress : %d, %s, %d, %s\n",
    monSociete[i].adresse.numero,
    monSociete[i].adresse.rue,
    monSociete[i].adresse.code,
    monSociete[i].adresse.localite);

/* Rappelez aux etudiants que pour le types enumeration
   les constantes symboliques dans le type, s'identifient
   a la liste de premier entiers naturels, donc pour sexe,
   feminin = 0 et masculin = 1 */

switch (monSociete[i].sexe){
    case (0) : printf("\tSex      : feminin\n"); break;
    case (1) : printf("\tSex      : masculin\n"); break;
}

if (monSociete[i].sexe == masculin)
    switch (monSociete[respEmp].complement.sitMil){
        case (0) : printf("\tMili    : libere\n"); break;
        case (1) : printf("\tMili    : exempte\n"); break;
        case (2) : printf("\tMili    : reforme\n"); break;
        case (4) : printf("\tMili    : incorporable\n");
    }
}
}
if (trouve == FAUX)
    printf("\n\tLe nom %s n'est pas trouve. \n\n", nomEmp);
break;

case (3) :
    printf("\n** A bientot **\n\n");
    return(0);
}
}while(respMenu != 3);
}

```

Et un exemple de son execution :

```

*** System du Personnel ***

    [1] Saisir les donnees du personnel.
    [2] Consulter les donnees du personnel.
    [3] Quitter the system.

    Tapper votre choix : 1

** Saisie Information du Personnel **

/* idem a l'exercice precedente */

*** System du Personnel ***

    [1] Saisir les donnees du personnel.
    [2] Consulter les donnees du personnel.
    [3] Quitter the system.

    Tapper votre choix : 2

** Consulte Information du Personnel **

```

Nom de l'employe a consulter : Brooks

Employe No : 1

Nom : Brooks
Prenom : Louise
Nom JF : Schmidt
Adress : 4, Edgar_Quinet, 75014, Paris
Sex : feminin

*** System du Personnel ***

[1] Saisir les donnees du personnel.
[2] Consulter les donnees du personnel.
[3] Quitter the system.

Tapper votre choix : 2

** Consulte Information du Personnel **

Nom de l'employe a consulter : Fuentes

Employe No : 2

Nom : Fuentes
Prenom : Carlos
Adress : 1, Victoire, 75014, Paris
Sex : masculin

*** System du Personnel ***

[1] Saisir les donnees du personnel.
[2] Consulter les donnees du personnel.
[3] Quitter the system.

Tapper votre choix : 2

** Consulte Information du Personnel **

Nom de l'employe a consulter : Lopez

Le nom Lopez n'est pas trouve.

*** System du Personnel ***

[1] Saisir les donnees du personnel.
[2] Consulter les donnees du personnel.
[3] Quitter the system.

Tapper votre choix : 3

** A bientot **

EXERCICE 5 : Le code :

```
#include<stdio.h>
#include<string.h> /* Pour comparer des chaines de caracteres */
```

```

#define TAILLEMAXNOM 30
#define TAILLEMAXENTREPRISE 100
#define VRAI 1
#define FAUX 0

int main() {

    /* ajouter la definition des types idem a l'exercice 2 */

    typedef typeEmploye typeSociete[TAILLEMAXENTREPRISE];

    typeSociete monSociete;
    int         tailleMonSociete=0;
    int         i;
    int         respMenu; /* le response aux choix du Menu*/
    int         respSex, respSitMil, respEmp; /* autres responses */
    typeNom     nomEmp; /* nom de l'employe a consulter */
    int         trouve; /* TRUE si on a trouve l'employe */

    do{
        do {

            /* Menu Principal */

            printf("\n\n*** System du Personnel *** \n\n");
            printf("\t[1] Ajouter les donnees d'un/une employe(e). \n
                \t   Space utilise : %d de %d enregistrements disponibles.\n",
                tailleMonSociete, TAILLEMAXENTREPRISE);
            printf("\t[2] Consulter les donnees du personnel. \n");
            printf("\t[3] Quitter the system. \n\n");
            printf("\t   Tapper votre choix : ");

            scanf("%d",&respMenu);

        } while ((respMenu < 1) || (respMenu > 3));

        switch (respMenu) {
            case (1) :
                printf("\n** Saisie Information du Personnel ** \n\n");

                if (tailleMonSociete < TAILLEMAXENTREPRISE) {

                    printf("\n\tDonnees pour l'employe %d. \n", tailleMonSociete+1);

                    printf("\tNom : ");
                    scanf("%s", &monSociete[tailleMonSociete].nom);

                    /* Rappelez aux etudiants que scanf avec la chaine de
                        control "%s", s'arrete au premier space rencontre.
                        Dans cet exercice je utilise '_' a la place des
                        spaces, i.e., "Alphonse Daudet" devient
                        "Alphonse_Daudet" */

                    printf("\tPrenom : ");
                    scanf("%s", &monSociete[tailleMonSociete].prenom);

                    printf("\tAdresse : \n");

                    printf("\t\tNumero : ");
                    scanf("%d", &monSociete[tailleMonSociete].adresse.numero);
                }
            }
        }
    }
}

```

```

printf("\t\tRue : ");
scanf("%s", &monSociete[tailleMonSociete].adresse.rue);

printf("\t\tCode : ");
scanf("%d", &monSociete[tailleMonSociete].adresse.code);

printf("\t\tLocalite : ");
scanf("%s", &monSociete[tailleMonSociete].adresse.localite);

do{
    printf("\tSexe \t[1]feminin / [2]masculin : ");
    scanf("%d", &respSex);
} while (respSex < 1 || respSex > 2);

switch (respSex) {
    case (1) :
        monSociete[tailleMonSociete].sexe = feminin;
        printf("\tNom de Jeune Fille : ");
        scanf("%s", &monSociete[tailleMonSociete].complement.nomJF);
        break;
    case (2) :
        monSociete[tailleMonSociete].sexe = masculin;

        do{
            printf("\tSituation Militaire \t[1]libere / [2]exempte / \n
                \t\t\t\t[3]reforme / [4]incorporable : ");
            scanf("%d", &respSitMil);
        } while (respSitMil < 1 || respSitMil > 4);

        switch (respSitMil) {
            case (1) :
                monSociete[tailleMonSociete].complement.sitMil = libere;
                break;
            case (2) :
                monSociete[tailleMonSociete].complement.sitMil = exempte;
                break;
            case (3) :
                monSociete[tailleMonSociete].complement.sitMil = reforme;
                break;
            case (4) :
                monSociete[tailleMonSociete].complement.sitMil = incorporable;
                break;
        }
        break;
    }

    tailleMonSociete ++;

    printf("\n** Information saisi ** \n");

} else {
    printf("\n** Il n'y a pas de space pour un/une nouveau(elle) employe(e). \n");
}

break;

case (2) :
    printf("\n** Consulte Information du Personnel ** \n\n");
    printf("\tNom de l'employe a consulter : ");
    scanf("%s", &nomEmp);

    trouve = FAUX;

```

```

for (i=0; i<tailleMonSociete; i++) {
    if (!strcmp(nomEmp, monSociete[i].nom)) {
        trouve = VRAI;
        printf("\n\tEmploye No : %d\n", i+1);
        printf("\tNom      : %s\n", monSociete[i].nom);
        printf("\tPrenom   : %s\n", monSociete[i].prenom);

        if (monSociete[i].sexe == feminin)
            printf("\tNom JF : %s\n", monSociete[i].complement.nomJF);

        printf("\tAdress : %d, %s, %d, %s\n",
            monSociete[i].adresse.numero,
            monSociete[i].adresse.rue,
            monSociete[i].adresse.code,
            monSociete[i].adresse.localite);

        /* Rappelez aux etudiants que pour le types enumeration
           les constantes symboliques dans le type, s'identifient
           a la liste de premier entiers naturels, donc pour sexe,
           feminin = 0 et masculin = 1 */

        switch (monSociete[i].sexe){
            case (0) : printf("\tSex      : feminin\n"); break;
            case (1) : printf("\tSex      : masculin\n"); break;
        }

        if (monSociete[i].sexe == masculin)
            switch (monSociete[respEmp].complement.sitMil){
                case (0) : printf("\tMili    : libere\n"); break;
                case (1) : printf("\tMili    : exempte\n"); break;
                case (2) : printf("\tMili    : reforme\n"); break;
                case (4) : printf("\tMili    : incorporable\n");
            }
        }
    }
    if (trouve == FAUX)
        printf("\tLe nom %s n'est pas trouve. \n\n", nomEmp);
    break;

case (3) :
    printf("\n** A bientot **\n\n");
    return(0);
}
}while(respMenu != 3);
}

```

Et un exemple de son execution :

*** System du Personnel ***

```

[1] Ajouter les donnees d'un/une employe(e).
    Space utilise : 0 de 100 enregistrements disponibles.
[2] Consulter les donnees du personnel.
[3] Quitter the system.

```

Tapper votre choix : 1

** Saisie Information du Personnel **

```

Donnees pour l'employe 1.
Nom : Guerra

```



```
Prenom : Alejandro
Adresse :
    Numero : 6
    Rue : Victoire
    Code : 75001
    Localite : Paris
Sexe [1]feminin / [2]masculin : 2
Situation Militaire [1]libere / [2]exempte /
                    [3]reforme / [4]incorporable : 1
```

** Information saisi **

*** System du Personnel ***

```
[1] Ajouter les donnees d'un/une employe(e).
    Space utilise : 1 de 100 enregistrements disponibles.
[2] Consulter les donnees du personnel.
[3] Quitter the system.
```

Tapper votre choix : 1

** Saisie Information du Personnel **

```
Donnees pour l'employe 2.
Nom : Greenspan
Prenom : Alain
Adresse :
    Numero : 15
    Rue : Bourse
    Code : 75010
    Localite : Paris
Sexe [1]feminin / [2]masculin : 2
Situation Militaire [1]libere / [2]exempte /
                    [3]reforme / [4]incorporable : 1
```

** Information saisi **

/* le reste idem aux exercices precedentes */

EXERCICE POUR LE TP : Le code (l'exécution est identique a celle de l'exercice precedent) :

```
#include<stdio.h>
#include<string.h> /* Pour comparer des chaines de caracteres */

#define TAILLEMAXNOM 30
#define TAILLEMAXENTREPRISE 100
#define VRAI 1
#define FAUX 0

/* Definition global de types pour que les fonctions les
    utilisent */

typedef char typeNom[TAILLEMAXNOM];

typedef enum {
    feminin,
    masculin
} typeSexe;
```

```

typedef enum {
    libere,
    exempte,
    reforme,
    incorporable
} typeSitMil;

typedef struct {
    unsigned numero;
    typeNom rue;
    unsigned code;
    typeNom localite;
} typeAdresse;

typedef union {
    typeSitMil sitMil;
    typeNom nomJF;
} typeInfoComp;

typedef struct {
    typeNom nom;
    typeNom prenom;
    typeAdresse adresse;
    typeSexe sexe;
    typeInfoComp complement;
} typeEmploye;

typedef typeEmploye typeSociete[TAILLEMAXENTREPRISE];

/* int menuPrincipal

Fonction pour le menu principal. Le parametre tailleSoc est
necessaire pour indiquer le space utilise. La fonction retourne
le choix del utilisateur (resp) */

int menuPrincipal (int tailleSoc) {
    int resp;

    do {
        printf("\n\n*** System du Personnel *** \n\n");
        printf("\t[1] Ajouter les donnees d'un/une employe(e). \n
\t Space utilise : %d de %d enregistrements
disponibles.\n", tailleSoc, TAILLEMAXENTREPRISE);
        printf("\t[2] Consulter les donnees du personnel. \n");
        printf("\t[3] Quitter the system. \n\n");
        printf("\t Tapper votre choix : ");

        scanf("%d",&resp);

    } while ((resp < 1) || (resp > 3));
    return resp;
}

/* void saisie

La fonction saisie utilise une variable local Emp pour capturer
les donnees saisiees pour l'utilisateur. Le parametre tailleSoc
s'utilise pour indiquer le nombre d'utilisateur qu'on est en
train de saisir. Il est passe par adresse pour permettre son
actualisation dans la fonction (taille precedent + 1). La

```

```

fonction renvoi comme resultat l'employe saisie.*/

typeEmploye saisie (int *tailleSoc) {
    typeEmploye Emp;
    int      respSex, respSitMil, respEmp;    /* autres responses */

    printf("\n** Saisie Information du Personnel ** \n\n");
    printf("\n\tDonnees pour l'employe %d. \n", *tailleSoc+1);

    printf("\tNom : ");
    scanf("%s", &Emp.nom);

    /* Rappelez aux etudiants que scanf avec la chaine de
       control "%s", s'arrete au premier space rencontre.
       Dans cet exercice je utilise '_' a la place des
       spaces, i.e., "Alphonse Daudet" devient
       "Alphonse_Daudet" */

    printf("\tPrenom : ");
    scanf("%s", &Emp.prenom);

    printf("\tAdresse : \n");

    printf("\t\tNumero : ");
    scanf("%d", &Emp.adresse.numero);

    printf("\t\tRue : ");
    scanf("%s", &Emp.adresse.rue);

    printf("\t\tCode : ");
    scanf("%d", &Emp.adresse.code);

    printf("\t\tLocalite : ");
    scanf("%s", &Emp.adresse.localite);

    do{
        printf("\tSexe \t[1]feminin / [2]masculin : ");
        scanf("%d", &respSex);
    } while (respSex < 1 || respSex > 2);

    switch (respSex) {
        case (1) :
            Emp.sexe = feminin;
            printf("\tNom de Jeune Fille : ");
            scanf("%s", &Emp.complement.nomJF);
            break;
        case (2) :
            Emp.sexe = masculin;

            do{
                printf("\tSituation Militaire \t[1]libere / [2]exempte / \n
                    \t\t\t\t[3]reforme / [4]incorporable : ");
                scanf("%d", &respSitMil);
            } while (respSitMil < 1 || respSitMil > 4);

            switch (respSitMil) {
                case (1) :
                    Emp.complement.sitMil = libere;
                    break;
                case (2) :
                    Emp.complement.sitMil = exempte;
                    break;
            }
    }
}

```

```

        case (3) :
            Emp.complement.sitMil = reforme;
            break;
        case (4) :
            Emp.complement.sitMil = incorporable;
            break;
    }
    break;
}

*tailleSoc = *tailleSoc + 1;

printf("\n** Information saisi ** \n");

return Emp;
}

/* void consulte

    consulte cherche dans la societe monSoc de taille tailleSoc
    un/une employe de nom donne par l'utilisateur. */

/* Rappelez aux etudiants la difference entre parametre par
    valeur et par adresse */

void consulte (typeSociete monSoc, int tailleSoc) {
    typeNom nomEmp;

/* Rappelez aux etudiants que trouve etais une variable dans la
    fonction main, et maintenant est une variable local dans la
    fonction consulte. */

    int trouve;
    int i;

    printf("\n** Consulte Information du Personnel ** \n\n");
    printf("\tNom de l'employe a consulter : ");
    scanf("%s",&nomEmp);
    trouve = FAUX;

    for (i=0; i<tailleSoc; i++) {
        if (!strcmp(nomEmp, monSoc[i].nom)) {
            trouve = VRAI;
            printf("\n\tEmploye No : %d\n\n", i+1);
            printf("\tNom      : %s\n", monSoc[i].nom);
            printf("\tPrenom : %s\n", monSoc[i].prenom);

            if (monSoc[i].sexe == feminin)
                printf("\tNom JF : %s\n", monSoc[i].complement.nomJF);

            printf("\tAdress : %d, %s, %d, %s\n",
                monSoc[i].adresse.numero,
                monSoc[i].adresse.rue,
                monSoc[i].adresse.code,
                monSoc[i].adresse.localite);

/* Rappelez aux etudiants que pour le types enumeration
    les constantes symboliques dans le type, s'identifient
    a la liste de premier entiers naturels, donc pour sexe,
    feminin = 0 et masculin = 1 */

            switch (monSoc[i].sexe){

```

```

        case (0) : printf("\tSex      : feminin\n"); break;
        case (1) : printf("\tSex      : masculin\n"); break;
    }

    if (monSoc[i].sexe == masculin)
        switch (monSoc[i].complement.sitMil){
            case (0) : printf("\tMili    : libere\n"); break;
            case (1) : printf("\tMili    : exempte\n"); break;
            case (2) : printf("\tMili    : reforme\n"); break;
            case (4) : printf("\tMili    : incorporable\n");
        }
    }
}
if (trouve == FAUX)
    printf("\tLe nom %s n'est pas trouve. \n\n", nomEmp);
}

int main() {
    typeSociete monSociete;
    int         tailleMonSociete=0;
    int         respMenu; /* le response aux choix du Menu*/

do{

    /* respMenu initialise en utilisant la fonction menuPrincipal*/

    switch (respMenu= menuPrincipal(tailleMonSociete)) {
        case (1) :
            if (tailleMonSociete < TAILLEMAXENTREPRISE) {

                /* Rappelez aux etudiants que saisie renvoi comme resultat
                un/une employee(e) capture. Aussi que le parametre de
                saisie est passe par adresse */

                monSociete[tailleMonSociete] = saisie(&tailleMonSociete);
            } else {
                printf("\n** Il n'y a pas de space pour un/une nouveau(elle) employe(e). \n");
            }
            break;

        case (2) :
            consulte(monSociete,tailleMonSociete);
            break;

        case (3) :
            printf("\n** A bientot **\n\n");
            return(0);
    }
}while(respMenu != 3);
}

```