Licence Informatique
Université Bordeaux I

Initiation à la Programmation
Travaux Dirigés

Structure

Créer le répertoire `~/IP/6_structure`.

Une structure (ou enregistrement, *record* en anglais) est un *alignement* d'objets différents (on parle d'agrégation d'objets, de juxtaposition d'objets). On la définit ainsi :

```
struct identificateur
{
  type1 champ1;
  type2 champ2;
  type3 champ3;
};
```

par exemple :

```
struct complexe
{
  double reel;
  double imaginaire;
};
```

Pour déclarer un nombre complexe on écrit alors : `struct complexe c;`
On accède aux champs ainsi : `c.reel = c.imaginaire = 0;`
L'affectation est possible (mais pas la comparaison)

```
struct complexe c, d;
c.reel = c.imaginaire = 0;
d = c;
```

Exercice 1. Écrire et tester une fonction qui retourne la somme de deux complexes.

Exercice 2. Déterminer la taille des objets suivants :

```
struct
{
  double r;
  char a;
} obj1;
```

```
struct
{
  char a;
  double r;
} obj2;
```

```
struct
{
  char a;
  char b;
  double r;
} obj3;
```

```
struct
{
  char a;
  double r;
  char b;
} obj4;
```

```
struct
{
  double r;
  char a;
  char b;
} obj5;
```

En utilisant GDB, représenter la “cartographie” mémoire de ces objets.

Exercice 3. Soit les définitions suivantes :

```
typedef struct
{
  double valeur[MAX_DIM];
  size_t dimension;
} vecteur ;
```

```
void
nulle(vecteur v, size_t dim)
{
  size_t d;
  for(d=0; d < dim; d++)
    v.valeur[d] = 0;
  v.dimension = dim;
}
```

Que fait la fonction la fonction `nulle` ? La corriger.