



Rappels et compléments de langage C

LaBRI, UMR CNRS 1308
Université Bordeaux 1

Serge Chaumette
Serge.Chaumette@labri.fr

1

[Rappels de langage C]



Objectifs de ce cours

■ Objectifs

- Assurer un niveau homogène
- Assurer un niveau minimum garanti

■ Évaluation

- Exercices
- Au fur à mesure des manipulations
- Au fur à mesure des autres cours

Serge.Chaumette@labri.fr

2



Ce qui est supposé maîtrisé

- La gestion dynamique de la mémoire
- Les pointeurs
- L'encapsulation des types
- Les pointeurs de fonctions



Exercice

Gestion dynamique de la mémoire

15 minutes

- Écrire un programme qui lit sur son entrée standard un entier représentant la longueur d'un tableau. Le programme lit ensuite deux tableaux de cette longueur là, élément par élément, puis calcule, au sens vectoriel du terme, la somme de ces deux tableaux en stockant le résultat dans un troisième. Enfin, le programme affiche ce résultat et libère la mémoire inutile.

Exercice

Pointeurs

5 minutes

- Écrire une fonction permettant de modifier l'indice du premier élément d'un tableau. On pourra par exemple faire la chose suivante :

```
char *monTableau = (char *) malloc(100);  
monTableau=shifter(monTableau, -10);  
printf("%s\n", monTableau[-10]);
```

Exercice

Encapsulation des types

20 minutes

- Écrire un module de pile implémentant le type `Pile` et ses fonctions classiques, i.e. `empiler`, `depiler`, `pileVide`.

5 minutes

- Utiliser le module ci-dessus pour lire une suite d'entiers au clavier puis les afficher dans l'ordre inverse de celui dans lequel ils ont été lus.

10 minutes

- Assurez vous que vous pouvez changer votre implémentation de `Pile` sans changer le programme qui l'utilise.



Exercice

Pointeurs de fonctions

30 minutes

- Écrire une fonction qui reçoit en paramètre le nom d'une fonction sous la forme d'une chaîne de caractères, charge le .o de cette fonction à partir du répertoire courant puis exécute cette fonction. On utilisera pour cela la bibliothèque dl. Par exemple `execFonction(``toto"); :`
 - Charge dynamiquement le code de toto contenu dans le fichier toto.o
 - Exécute la fonction `void toto(void)`