

TD 10

pointeur, fonctions et structures

Vincent Thomas (vthomas@loria.fr)

29 avril 2001

1 Pointeurs

Algorithme et programmation :

Créer un entier entier.

Afficher son adresse à l'écran, créer un pointeur `ptrentier` qui pointe vers cet entier, afficher la valeur contenue dans l'entier à l'aide du pointeur `ptrentier`, modifier la valeur contenue dans l'entier à l'aide du pointeur `ptrentier`.

Créer un second pointeur `pointentier` et allouez le dynamiquement (en vérifiant que la mémoire n'est pas saturée), modifiez la valeur vers laquelle il pointe, et détruisez la variable en libérant la mémoire allouée.

2 Fontions , passage par valeur, par adresse

Ecrire et programmer une fonction qui renvoie pour un entier donné la valeur de cet entier multipliée par 2. Appeler cette fonction dans le main pour remplacer le contenu d'une variable de type entier par le double de sa valeur.

Ecrire et programmer une procédure qui pour un entier affiche à l'écran le double de sa valeur. Effectuer un appel de fonction dans le main.

Ecrire et programmer une procédure qui remplace directement la valeur stockée dans une variable par le double de cette valeur. Effectuer un appel de fonction dans le main.

Ecrire et programmer une fonction qui pour un tableau de taille donnée modifie les valeurs contenues dans ce tableau pour les remplacer par leur double.

3 Création d'un nouveau type : les tableaux dynamiques

Jusqu'à présent nous nous sommes intéressés à des tableaux de taille constante, avec les pointeurs, on se rend compte qu'il est possible d'allouer dynamiquement des tableaux de taille variable. Cependant, cela pose des problèmes puisqu'il n'est toujours pas possible de connaître le nombre de cellules réservées en mémoire et donc la taille du tableau.

Pour contourner cela, nous allons définir un nouveau type : le type "tableaumentier".

Ce type sera défini comme le produit cartésien de deux champs :

- un entier taille qui nous donnera le nombre d'éléments du tableau.
- un pointeur sur entier alloué dynamiquement qui représentera le tableau et dont on pourra faire varier la taille.

Bien entendu (et c'est tout l'intérêt d'une telle structure ..), on supposera que la structure est cohérente et qu'elle le reste après chaque fonction ou procédure, c'est à dire que taille correspond bien aux nombre de cases allouées.

A faire :

- Définir et programmer cette structure.
- Créer un tableau de type "tableau entier" dans le main capable de contenir 5 variables.
- Créer une fonction contenance qui prend un "tableaumentier" comme argument et qui renvoie le nombre d'éléments que peut contenir ce tableau.
- Créer une procédure "Allouer" vers laquelle on passera par adresse une variable de type "tableaumentier" et un entier n par valeur. Cette procédure se chargera d'allouer le "tableaumentier" afin qu'il puisse contenir n variable. Pourquoi faire un passage par adresse?
- Créer une procédure "Détruire" qui se chargera de libérer la mémoire réservée pour le "tableaumentier" passé par adresse.
- Créer une procédure qui ajoute un élément n à une variable de type "tableau entier" (en augmentant la taille du tableau, c'est à dire : en modifiant le champ taille, en allouant en nouveau pointeur, en copiant les anciennes valeurs dans le nouveau tableau , en ajoutant l'élément n à la fin de ce tableau et en détruisant l'ancien tableau)

4 Pile et Liste