

Sinf2 : Allocation dynamique

15.IV.2021

Réponse à un besoin : tableau de taille non connue lors de la compilation du programme.

Un fichier txt "il fait beau"

"le chien aboie et le chaton pisse"

lors de la compilation
du programme

Tableau à taille fixe

Tableau à taille

dynamique

lors de l'exécution
du programme

Lire le Fichier txt → transférer son contenu dans un chaîne (tableau)

Option 1: on use à tableau suffisamment grand. ? combien de caractères ?

Option 2: a) Calcul de la taille du fichier

b) créer un tableau correspondant au besoin

Mode opératoire: ① Type de donnée à mettre dans le tableau

② Nombre d'éléments a-t-on besoin

Ex: type : double

n : 10

1) Créer une variable qui va contenir l'adresse de début de la zone

double *t = NULL;

2) demander au système une zone minimale de la "bonne" taille
taille : n * sizeof(double) → en octets

t = (double *) malloc (n * sizeof(double));

↳ transtypage (on modifie explicitement le type renvoyé par la fonction malloc (void*) par le type attendu (double*))

3) Vérifier que l'allocation est correcte.

Si erreur: t == NULL

↳ NE PAS UTILISER LE TABLEAU !

Si ok: t != NULL

↳ on peut utiliser le tableau: t[0] à t[n-1]

{ IMPÉRATIF
DE TESTER }

④ Après l'utilisation du tableau (plus besoin): rend la zone au système.

free(t); // précaution pour éviter un erreur en cas
t = NULL; d'utilisation du tableau après libération.

< stdlib.h >

double *p = (double *) malloc (...);

p \leftarrow 2000

p[0] = 3.14 ;

p[0]

ZONE
ALLUEE

3.14 2.718

2000

DANGEREUX

free(p)

~~p[0] = 2.718 ;~~

) OPTION 1

free(p)

p = NULL

) OPTION 2

p[0] = 2.718

un Génér. autom. à l'input
sig-fault.

/app

Windows

TD 2020415

1) allocation dynamique d'un tableau de 7 éléments du type int64_t ;

2) affichez la valeur du pointeur alloué.

3) setez les valeurs du malloc

4) libérez la mémo. allouée.

