

TD 3

semaine du 29/3/00

Objectifs : Comprendre le chargement en mémoire

Tous les programmes sont disponibles sous unix dans /home/users/TPINFO/sys1 et sous Windows dans TPINFO/sys1.

1. Démarrez sous Windows. Copiez `memoire.c` et ouvrez-le sous Borland C. Que fait ce programme ?

Avec le bouton droit de la souris dans la fenêtre source, choisissez TargetExpert. Sélectionnez Application –DOS standard –large. Compilez et exécutez le programme. Notez le résultat :
Choisissez à nouveau TargetExpert, sélectionnez Application-Windows-console. Compilez et exécutez le programme. Notez le résultat :
Passez sous linux. Copiez le programme, compilez le (`make memoire`) et exécutez-le. Notez le résultat :

2. **Emplacement et initialisation des tableaux.** Copiez `tableaux.c`. Lisez le code, puis exécutez-le. Notez les résultats :

Variable	portée	durée de vie	adresse	taille	valeur initiale
<code>gtab1</code>					
<code>gvar</code>					
<code>gtab2</code>					
<code>i</code>					
<code>j</code>					
<code>ltab1</code>					
<code>ltab2</code>					
<code>ltptr</code>					

Comment les tableaux ont-ils été placés en mémoire :

3. **Où sont le code et les variables.** Copiez `adr_var_fon.c`. Lisez le code, puis exécutez-le. Notez les résultats et complétez quand vous pouvez :

Variable	portée	durée de vie	adresse	taille
<code>main</code>				
<code>wher_var</code>				
<code>gtab</code>				
<code>gvar</code>				
<code>lc</code>				
<code>lc[]</code>				
<code>mi</code>				
<code>mj</code>				
<code>mtab[0]</code>				
<code>mtab[9]</code>				
<code>i</code>				
<code>j</code>				
<code>tab[0]</code>				
<code>tab[9]</code>				
<code>ip</code>				
<code>jp</code>				
<code>ptab0</code>				
<code>ptab9</code>				
<code>li</code>				
<code>slj</code>				
<code>ltab</code>				

Quelle est la place prise en mémoire par un appel a where_var ?

4. **Taille maximale de la pile** : Copiez taille_pile.c. Lisez le code, expliquez ce qu'il fait. Essayez de prévoir comment le programme s'arrete, puis exécutez-le. Notez les résultats. Quelle est la place prise par un appel à topstack ? Modifiez le programme pour avoir directement la taille maximale de la pile en Ko.
5. **Allocation dynamique et désallocation**. Copiez alloc_free.c. Lisez le code, expliquez ce qu'il fait. Exécutez-le. Etudiez le résultat. Comment se fait la désallocation et la ré-allocation ? Quelle est la différence d'adresse entre deux zones allouées successives ? Expliquez.
6. Repassez sous Windows et comparez les résultats pour les question 2 à 5, avec un modèle DOS standard puis Windows Console