

Langage C

TP3

Itheri Yahiaoui && Stéphane Cormier

Exercice 1 : « Représentation mémoire »

Rappel: La librairie <limits.h> détermine diverses propriétés des différents types de variables. Les macros définies dans cet en-tête, limitent les valeurs de divers types de variables comme char, int et long. Ces limites spécifient qu'une variable ne peut stocker aucune valeur au-delà de ces limites, par exemple un caractère non signé peut stocker jusqu'à une valeur maximale de 255.

- Tester le code exemple suivant afin d'afficher les valeurs associées à quelques constantes définies par des macros dans <limits.h>.

```
#include <stdio.h>
#include <limits.h>
int main() {
    printf("Le nombre de bits pour représenter le type CHAR = %d\n", CHAR_BIT);
    printf("La valeur minimale pour une variable de type SIGNED CHAR = %d\n", SCHAR_MIN);
    printf("La valeur maximale pour une variable de type SIGNED CHAR = %d\n", SCHAR_MAX);
    printf("La valeur maximale pour une variable de type UNSIGNED CHAR = %d\n", UCHAR_MAX);
    printf("La valeur minimale pour une variable de type SHORT INT = %d\n", SHRT_MIN);
    printf("La valeur maximale pour une variable de type SHORT INT = %d\n", SHRT_MAX);
    printf("La valeur minimale pour une variable de type INT = %d\n", INT_MIN);
    printf("La valeur maximale pour une variable de type INT = %d\n", INT_MAX);
    printf("La valeur minimale pour une variable de type CHAR = %d\n", CHAR_MIN);
    printf("La valeur maximale pour une variable de type CHAR = %d\n", CHAR_MAX);
    printf("La valeur minimale pour une variable de type LONG = %ld\n", LONG_MIN);
    printf("La valeur maximale pour une variable de type LONG = %ld\n", LONG_MAX);
    return(0);
}
```

- En consultant le lien suivant https://en.wikipedia.org/wiki/C_data_types#limits.h , ou <http://www.cplusplus.com/reference/climits/> compléter l'exemple précédent afin d'afficher les limites d'autres types de variables entières.

- Ecrire un autre programme pour afficher les propriétés des types réels en utilisant `<float.h>`. Vous pouvez consulter les différentes macros définies dans `<float.h>` via le lien <http://www.cplusplus.com/reference/cfloat/> ou https://en.wikipedia.org/wiki/C_data_types#float.h.
- Soit le nombre réel x de type double. Ecrire un programme qui affiche un entier long, formé à partir du réel x , tronqué de sa partie décimale. Si le résultat de la troncature ne se trouve pas dans l'intervalle de définition du type long int, le programme est interrompu en renvoyant le code (-1).
- Ecrire un programme qui affiche la taille, en octets et en bits, des différents types de variables. Vous pouvez utiliser la fonction `sizeof()` qui renvoie un résultat de type « `size_t` ». L'unité du résultat est de type char. Un char ne mesure pas forcément un octet.
- Ecrire un programme qui vous permet de calculer, en utilisant les résultats de la question précédente les limites des différents types « entier ». Comparer vos résultats avec ceux du code exemple présenté au début de cet exercice.

Exercice 2 : « Tableaux statiques à une dimension »

- Ecrire un programme qui affiche tous les éléments pairs présents dans un tableau.
- Ecrire un programme qui cherche un nombre entier « x » saisi par l'utilisateur dans un tableau « T » rempli aléatoirement et qui affiche le plus petit rang i tel que $T[i] = x$ sinon -1.
- Ecrire un programme qui fait la somme de deux tableaux réels, de la même dimension, dans un troisième tableau, la somme se fera élément par élément.
- Ecrire un programme qui réalise une recherche dichotomique d'une valeur saisie par l'utilisateur dans un tableau initialisé par vous même.
- Ecrire un programme pour afficher les éléments uniques d'un tableau.
- Ecrire un programme qui teste si deux tableaux d'entiers, saisis par l'utilisateur, comportent les mêmes éléments.
- Ecrire un programme qui stocke dans un tableau les nombres parfaits inférieurs à N donné par l'utilisateur. Un entier positif est parfait s'il est égal à la somme de ses diviseurs (lui-même étant exclus)
- Ecrire un programme qui remplit un tableau de dimension N d'une manière aléatoire de 0 et 1, ensuite il affiche l'indice de début de la plus longue suite de 1.
- Ecrire un programme qui tasse un tableau de N nombres positifs, c'est à dire qui détecte les éléments nuls du tableau et qui récupère leur case en décalant vers le début du tableau tous les autres éléments.
- Ecrire un programme pour fusionner deux tableaux de même taille. Les éléments dans le nouveau tableau doivent être triés par ordre décroissant.

Exercice 3 : « Pointeurs & Adresses »

- Ecrire des programmes qui réalisent les opérations suivantes en utilisant un accès indirect avec des pointeurs:
 - calcul de la somme et du maximum entre deux variables entières,
 - stockage « et affichage » de n éléments dans un tableau statique,
 - échange des éléments de deux tableaux statiques de la même taille,
 - affichage des caractères d'une chaîne stockée dans un tableau statique (un caractère par ligne),
 - comparer deux chaînes de caractères,
 - concaténer deux chaînes de caractères.