

Programmation Impérative - TP 2

Choses diverses et variées

Rohan Fossé - Léo Mendiboure - Guillaume Mercier
{rohan.fosse,leo.mendiboure}@labri.fr, mercier@enseirb-matmeca.fr

2018-2019

1 Opérations *bitwise*

Écrire un programme qui utilise *un* octet pour simuler le comportement d'une ampoule *intelligente*, dont la couleur et l'intensité sont variables.

Un squelette de programme se trouve ici : `/net/ens/mercier/PG109/TP2/light.c`

Au niveau de l'implémentation, vous utiliserez une variable de type `unsigned char` pour stocker les informations de cette ampoule et il y aura :

- besoin d'un bit pour activer ou non l'ampoule (deux états : on/off)
- besoin de 2 bits pour stocker les 4 couleurs (White/Blue/Red/Green)
- besoin de 4 bits pour stocker l'intensité de l'ampoule (de 10 à 100%, par incrément de 10, donc 10 valeurs).

2 Représentation de dates

Soit la structure de donnée suivante :

```
struct date {  
    int day;  
    int month;  
    int year;  
}
```

Question 1

Écrire la fonction `struct date *date_from_string(char *s)` qui prend en argument une date sous forme de chaîne de caractères `dd/mm/yyyy` et qui retourne un pointeur sur une structure `date`. Nous supposerons la date bien formée.

Question 2

Écrire la fonction `int date_compare(struct date*d1, struct date *d2)` qui retourne un nombre négatif si `d1` est plus petite que `d2`, 0 si les deux dates sont égales ou un nombre positif si `d1` est plus grand que `d2`.

Question 3

Écrire un programme `date_min` qui prend en argument un nombre quelconque de dates (au moins une) sous la forme `dd/mm/yyyy` et qui affiche la date la plus petite sous la forme `17 Mars 2017`.