

TP n°8

Les segments de mémoire partagée

Le but de ce TP est de manipuler les segments de mémoire partagée System V.

1 Message partagé

Nous souhaitons créer un programme qui prend en arguments une clé IPC pour le segment de mémoire partagée et un message (une chaîne de caractères) de taille quelconque. Si le segment de mémoire partagée n'existe pas, il est créé. Le programme y place alors son PID puis le message. Si le segment existe déjà, le programme affiche d'abord les anciennes valeurs puis y place les nouvelles



Il est conseillé de définir une taille maximale pour la chaîne de caractères (via une constante) : elle sera utile pour construire le segment de mémoire partagée et pour éviter les débordements.

2 Grille partagée

Nous considérons deux applications nommées *coordinateur* et *client*. Le *coordinateur* crée une grille de 15 cases par 30 (des unsigned char) dans un segment de mémoire partagée. Les cases sont initialisées à 0. Le *coordinateur* affiche la grille à l'écran toutes les 0,5 secondes. Quant aux *clients* (ils peuvent être exécutés autant de fois que l'on souhaite), ils réalisent en boucle les actions suivantes : ils choisissent une position aléatoire, ils y dessinent une croix puis ils se mettent en pause pendant un temps aléatoire (entre 0,5 secondes et 2 secondes).



Utilisez la fonction `usleep` pour réaliser les pauses.

Pour dessiner les croix, nous choisissons 6 couleurs numérotées de 0 à 5 (0 étant le noir). Une fois la position choisie, nous prenons la valeur maximale des 5 cases puis nous ajoutons 1 modulo 5 et nous obtenons la couleur de la croix. La valeur des 5 cases est alors remplacée par cette couleur.

1. Écrivez le code du *coordinateur* qui permet de créer le segment de mémoire partagée, ainsi que le tableau de sémaphores. Il affiche régulièrement la grille à l'écran.

Pour écrire une croix, il faut bloquer les sémaphores des zones concernées, choisir la valeur maximale des 5 cases, remplacer les valeurs des cases et enfin libérer les sémaphores.

2. Écrivez une fonction qui prend en paramètre une position (x,y), un tableau d'opérations de sémaphores (de type `sembuf`) et une opération (1 ou -1). Elle initialise les champs `sem_num` du tableau et retourne le nombre de sémaphores nécessaires.



Le tableau est de 3 cases car il ne peut y avoir plus de 3 sémaphores pour dessiner une croix.

3. Écrivez une fonction permettant de dessiner une croix à une position aléatoire. Elle utilise la fonction précédente pour sélectionner les sémaphores à bloquer/débloquer.



Vous pouvez choisir une position aléatoire en excluant la première et dernière ligne, ainsi que la première et dernière colonne.

4. Écrivez le *client*.
5. Admirez le résultat.