

## TP 5 – Sémaphores

### Exercice 1 : Sémaphore

En utilisant les fonctions qui gèrent un sémaphore. On adoptera les prototypes suivants :

```
void init sem(key t cle, int valeur) ;
void P(key t cle, int N) ;
void V(key t cle, int N) ;
void detruire sem(key t cle) ;
```

- Tester ces fonctions à l'aide d'un programme qui utilisera deux processus :

Le père affichera les multiples de 2 de la façon suivante : 2 4 6 8 10 12 ...

Le fils affichera les multiples de 3 de la façon suivante : 3 6 9 12 15 ...

Ces deux processus devront s'exécuter alternativement jusqu'à ce que les deux auront atteint une valeur supérieure à 100. Le passage de relais se fera lorsque la valeur affichée sera un multiple de 5.

On pourra obtenir les affichages suivants :

```
Père : 2 4 6 8 10
Fils : 3 6 9 12 15
Père : 12 14 16 18 20
...
```

### Exercice 2 :

Implémentez l'algorithme suivant qui repose sur le modèle producteur/consommateur.

*char T[N] ; // Tableau de N caractères – mémoire partagée*  
*Semaphore Plein = 0, Vide =N, Mutex=1 ;*

#### Producteur

```
{
    Int ip=0, M ;
    char ch[N] ;
    Repeter
    {
        M=Lire(ch, N) ;

        Deposer(ch, M, ip) ;

        Ip=(ip+M)%N ;
    }
}
```

}

#### Consommateur

```
{
    Int ic=0 ;
    char c ;
    Repeter
    {
        c=Retirer (ic) ;
        ic = (ic + 1 )%N ;

        Traiter(c) ;
    }
}
```

}

La fonction « **int Lire(char ch[], int N);** » construit, dans ch, une chaîne de caractères aléatoire de longueur comprise entre 1 et N. Elle retourne la longueur de la chaîne.

La fonction « **void Deposer(char ch[], int M, int ip);** » insère, dans le tampon T, la chaîne de caractères ch. M est la longueur de la chaîne.

La fonction « **char Retirer(int ic);** » retire un caractère du tampon T. Elle retourne le caractère retiré.

La fonction « **void Traiter(char c);** » traite le caractère.