

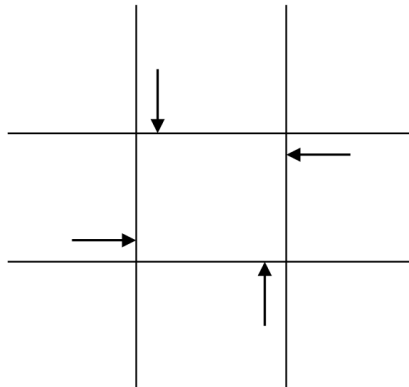
TP 5 : Sémaphores

Question:

Si on veut autoriser la traversée du carrefour à deux voitures simultanément, On a besoin de deux jetons, c-à-d initialiser le sémaphore à 2 avec (Initsem(C,2));

Exercice:

Le but de l'exercice est de refaire un exemple de cours en simulant quatre voitures par ajout d'un appel pour la 4ème voiture avec un délai aléatoire, et en permettant l'accès au carrefour routier à deux voitures à la fois (par initialiser le sémaphore C à 2).



Exclusion mutuelle dans un carrefour

tout d'abord on a créé une mini bibliothèque tubesem.h qui contient les déclarations des fonctions utilisées dans tubesem.c :

```
#include<unistd.h>
#include<sys/types.h>
#include<sys/wait.h>
typedef int Semaphore [2];

void Initsem(Semaphore S, int N);
void P(Semaphore S);
void V(Semaphore S);
```

```
void attente(int N);
void message(int i, char* s);
```

tubese.c contient les définitions des fonctions: Initsem, P, V, attente et message, aussi que la définition de la fonction voiture qui nous permettons de créer des voitures qui crée un processus fils avec fork() dont il va exécuter des fonction message pour donner le nom de voiture et son état, et aussi des fonctions attente pour faire des delay et il est le corps du module sur les semaphores

```
#include<stdio.h>
#include<stdlib.h>
#include "tubese.h"

Semaphore C;

main(){
    int i ;
    void voiture(int,int);
    printf("\n%20s%20s%20s%20s\n","VOITURE 1","VOITURE 2","VOITURE 3","VOITURE 4");
    Initsem(C,2);voiture(1,3);
    voiture(2,1);voiture(3,3);
    voiture(4,2);
    for(i=1;i<=4;i++)wait(0);}
void V (Semaphore S){
    char c='a';write(S[1],&c,1);}
    /* Attendre un nombre aleatoire de secondes entre 0 et N-1*/
    void attente(int N){
        sleep(rand() % N);
    }/* Ecrire un message s dans la i eme colonne, la premiere colonne a le
numero 1 */
    void message(int i, char* s){
        #define colonne 20
        int Nb, j;
        Nb=(i-1)*colonne;
        for(j=0; j<Nb; j++) putchar(' '); printf("%s\n" ,s);fflush(stdout);
    }
    void Initsem(Semaphore S, int N){
        int i;
        char c='a';
        pipe(S);
        for(i=1;i<=N;i++)write(S[1],&c,1);
    }
    /* P sur le semaphore S, prendre un jeton 'a' */
    void P (Semaphore S){
        char c;read(S[0],&c,1);
    }

void voiture(int n,int duree){
    if(fork()==0){attente(duree);
        message(n,"Arrivee");
```

```

        P(C);
        message(n,"Traversee");
        attente(duree);
        message(n,"Depart");
        V(C);
        exit(0);}
}

```

L 'execution de programme :

```

MBP-de-DevMQK:Chapitre 5 devmqk$ ./tub

VOITURE 1      VOITURE 2      VOITURE 3      VOITURE 4
               Arrivee
               Traversee
               Depart
Arrivee
Traversee

               Arrivee

               Arrivee
               Traversee

Depart

               Traversee

               Depart

MBP-de-DevMQK:Chapitre 5 devmqk$ █

```