

L'objectif de ce TP est la réalisation d'un outil en langage Java pour mesurer les performances d'une file d'attente M/M/S (file d'attente à plusieurs serveurs) à régime stationnaire.

A partir du taux d'arrivée  $\lambda$  et du taux de service  $\mu$ , le programme calcule et affiche d'abord le nombre minimum de serveurs nécessaires pour avoir un système stationnaire, puis calcule et affiche les mesures de performances suivantes :

1. Le taux d'utilisation des serveurs ( $a$ )
2. La probabilité que le système est vide (Aucun client dans le système)  $P_0$
3. La probabilité d'attente (probabilité que tous les serveurs soient occupés) (Erlang C)
4. Le nombre moyen de clients dans le système ( $\bar{N}$ )
5. Le nombre moyen de clients en attente ( $\bar{N}_Q$ )
6. Le nombre moyen de clients en service ( $\bar{N}_S$ )
7. Le temps d'attente moyen dans le système ( $\bar{T}$ )
8. Le temps d'attente moyen dans la file d'attente ( $\bar{T}_Q$ )

### Structure du code :

```
import java.util.Scanner;
public class MMS {
    private double lambda;
    private double mu;
    private int S;

    public MMS(double lambda, double mu, double S) {
        this.lambda=lambda;
        this.mu=mu;
        this.S=S;
    }

    /*
     * Le taux d'utilisation des serveurs (a)
     */
    public double a() {
        return lambda/(S*mu);
    }

    /*
     * La probabilité que le système est vide (Aucun client dans le système) P0
     */
    public double p0() {
        //Code
    }

    /*
     * La probabilité d'attente (probabilité que tous les serveurs c soient occupés)
     (Erlang C)
     */
    public double erlangC() {
        //Code
    }

    /*
```

```

    * Le nombre moyen de clients dans le système (N)
    */
    public double N() {
        //Code
    }

    /*
    * Le nombre moyen de clients en attente (Nq)
    */
    public double Nq() {
        //Code
    }

    /*
    * Le nombre moyen de clients en service (Ns)
    */
    public double Ns() {
        //Code
    }

    /*
    * Le temps d'attente moyen dans le système (T)
    */
    public double T() {
        //Code
    }

    /*
    * Le temps d'attente moyen dans la file d'attente (Tq)
    */
    public double Tq() {
        //Code
    }

    /*
    * Factoriel
    */
    public int factoriel(int n) {
        //Code
    }

    //....

    public static void main(String[] args) {
        double lambda, mu;
        int S;
        Scanner sc=new Scanner(System.in);
        System.out.println("Taux moyen d'arrivée des clients:");
        lambda=sc.nextDouble();
        //Code...

        MMS mms=new MMS(lambda,mu,S);

        //Code...

        System.out.println("Taux d'utilisation des serveurs:"+mms.a());

        //Code...
    }
}

```