TD 7

PCO

On souhaite créer une chaine de filtre de caractères s'appuyant sur 6 threads fonctionnant de la sorte : la thread principale (le main) lit des chaines de caractères au clavier et envoie les caractères de cette chaine les uns après les autre à la thread 1 qui remplace toutes les variations accentuées de la lettre e (ê, é, è, ê, ë) en e et laisse les autres caractères inchangés ; cette thread 1 passe tous les caractères reçus (modifiés ou non) à la thread 2 qui s'occupe de remplacer les variations de la lettre o (ô, ò, ö) en o, passe les caractères à la thread 3 qui fait de même avec la lettre i (î,ï) passe à la thread 4 qui traite les variations de a (à, â) et passe à la dernière thread qui les affiche à l'écran ; vous utiliserez pour se faire des instances de la classe Buffer1 pour « relier » les thread filtre entre elles.

Ainsi si l'utilisateur saisit la chaine « j'aime être près des vôtres », le programme doit afficher « j'aime etre pres des votres ».

Voici la base du code d'une thread Filtre :

```
public class Filtre extends Thread {
    private Buffer1 buffGauche;
    private Buffer1 buffDroit;
    private String pattern;
    private char remplacement;
    public Filtre1(Buffer1 buffEntree, Buffer1 buffSortie, String pattern, char remplacement) {
        this.buffGauche = buffEntree;
        this.buffDroit = buffSortie;
        this.pattern = pattern;
        this.remplacement = remplacement;
    }
    public void run() {
            // À écrire
    }
```

Écrire le code de la classe Filtre et le code du main.

On utilisera la méthode *s*.matches(*pattern*) pour savoir si la chaine *s* est conforme au paterne *pattern* donné sous la forme d'une expression régulière ;
ex : si s= "toto" alors s.matches(".*[abc].* ") retourne faux car s ne contient ni 'a' ni 'b' ni 'c' alors que s.matches(".*[o].* ") retourne vrai car s contient la caractère 'o'.