
TP de Programmation en JAVA RMI

TP 2

Exercice 1 :

Implémenter un serveur de calcul qui permettra d'effectuer à distance des opérations d'addition, de soustraction, de multiplication et de division pour des réel de type `double`. Ecrivez ensuite un client permettant de tester votre programme puis de les afficher.

Exercice 2 : Distribution du calcul de la valeur de PI

Il s'agit ici de construire une application répartie permettant d'utiliser les ressources de plusieurs serveurs pour calcul la valeur de PI avec une précision donnée. Dans un premier temps, nous allons appliquer une méthode permettant de calculer cette valeur. En 1995 Bailey, Borwein et Plouffe (BBP) ont montré la relation suivante :

$$\pi = \sum_{j=0}^{\infty} \left(\frac{4}{8j+1} - \frac{2}{8j+4} - \frac{1}{8j+5} - \frac{1}{8j+6} \right) \cdot \frac{1}{16^j}$$

qui permet à chaque itération sur `j` d'ajouter une décimale exacte à la représentation décimale de Pi. Comme le langage JAVA permet de travailler en arithmétique multiple précision (objets `BigDecimal` du package `java.math`), il est très simple d'écrire un petit programme qui calcule (avec la formule BBP) et affiche les `n` premières décimales de PI.

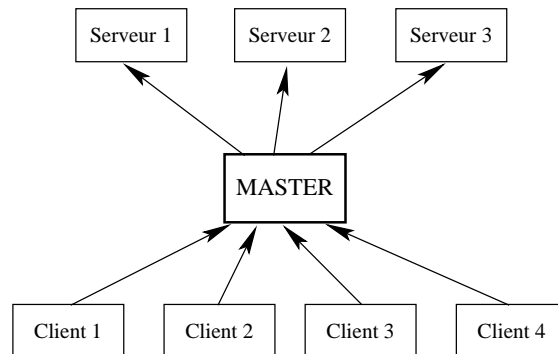
```
import java.math.*;

public class CalculDePI {

    public String calcul (int nbDecimal){
        BigDecimal somme = BigDecimal.ZERO;
        for (int j=0; j<nbDecimal; j++){
            BigDecimal part1 = new BigDecimal(4.0/(8*j+1));
            BigDecimal part2 = new BigDecimal(2.0/(8*j+4));
            BigDecimal part3 = new BigDecimal(1.0/(8*j+5));
            BigDecimal part4 = new BigDecimal(1.0/(8*j+6));
            BigInteger part5 = BigInteger.valueOf(16).pow (j);
            part1 = part1.subtract(part2);
            part1 = part1.subtract(part3);
            part1 = part1.subtract(part4);
            part1 = part1.divide (new BigDecimal(part5));
            somme = somme.add (part1);
        }
        // On ne retourne que les décimales demandées car le reste est faux
        BigDecimal nbd = new BigDecimal (10.0).pow (nbDecimal);
        // On multiplie par 10^nbdecimal
        somme = somme.multiply (nbd);
        // On prends la partie entière
        BigInteger tempSomme = somme.toBigInteger ();
        return ((new BigDecimal(tempSomme).divide (nbd)).toString ());
    }

    public static void main (String args []){
        if (args.length==0) {
            System.out.println ("Syntaxe : java CalculDePI ####");
            System.out.println ("#### représente le nom de décimal de PI");
        }
        else {
            CalculDePI cdpi = new CalculDePI ();
            for (int i=1; i<Integer.parseInt(args[0]); i++)
                System.out.println (cdpi.calcul (i));
        }
    }
}
```

Il va maintenant s'agir de distribuer ce programme de façon à que les demandes de calcul effectuées par des clients soient réparties sur plusieurs serveurs. Nous faisons le choix de l'architecture suivante :



Lorsqu'un client veut calculer la valeur de PI avec une certaine précision, il effectue sa demande auprès d'un serveur WEB qui va répartir la charge de calcul en fonction du nombre de serveurs dont il dispose. Puis, il récoltera les résultats de chaque partie du calcul pour fournir au client le résultat final. On supposera que chaque serveur doit s'identifier auprès du serveur maître lorsqu'il est lancé. Si le serveur maître ne dispose d'aucun serveur alors celui-ci effectue lui même le calcul.

À rendre

L'exercice sur le calcul de PI en mode décimal est à rendre par courriel à mon adresse qui est la suivante : Claude.Duvallet@gmail.com. Le sujet de ce message devra impérativement être "[MOSTA] TP de Java RMI". Votre message comportera une archive (zip ou tar.gz ou encore tgz) qui comportera un répertoire avec votre nom de famille dans lequel se trouvera tout votre travail. Un fichier README au format texte ou word expliquera comment compiler et exécuter votre projet. Dans ce même fichier une trace d'exécution pourra être donnée sous forme de texte ou de capture d'écran.

Pour l'exercice en lui-même, deux versions pourront être rendues :

1. Version simplifiée : le calcul de PI se fera au moyen d'un unique serveur qui renverra le résultat au client. Le client n'a pas forcément besoin de disposer d'une interface graphique. On pourra fournir le nombre de décimal sur la ligne de commande.
2. Version complète : il s'agit de la version telle qu'elle est présentée dans l'énoncé de l'exercice. Pour le client, la même remarque que précédemment est valable.

Date limite de rendu : Vendredi 6 février à minuit (00:00).