

Sockets

Práctica 1

1) Se deberá desarrollar una calculadora distribuida. Dividirse en dos grupos:

- Grupo 1: Desarrollará un programa que permita al usuario ingresar un número, luego elegirá una operación (+, -, *, /) de un menú y luego ingresará otro número. La operación no la realizará este programa sino que será solicitada en forma remota al servidor.
- Grupo 2: Desarrollará el programa servidor. El mismo recibirá una petición de una operación de suma, resta, multiplicación o división y devolverá el resultado.

Todas las operaciones se harán con tipos de datos float y se utilizarán sockets TCP.

TIP: Deberemos definir un protocolo de comunicación entre el cliente y el servidor, para el correcto envío y recepción de los datos.

Propuesta de protocolo ascii:

Petición de cálculo al servidor: ">CMD:%c,%f,%f\n" donde:

1. Comienza con '>'
2. %c se reemplaza por un caracter que indica el comando: '+', '-', '*', y '/'
3. %f se reemplaza por el número float ingresado
4. Finaliza con '\n'

Respuesta de petición hacia el cliente: ">CMD:%c,r:%f\n". Por ejemplo: ">CMD:+,r:3.14\n" donde:

1. Comienza con '>'
2. %c se reemplaza por un caracter que indica el comando: '+', '-', '*', y '/'
3. %f se reemplaza por el número float resultado de la operación

Respuesta en caso de error: ">ERR:%s\n". Por ejemplo ">ERR:Division por cero\n" donde

1. Comienza con '>'
2. %s se reemplaza por el mensaje de error.

Propuesta de protocolo binario:

Definir una struct para el mensaje hacia el servidor y otra para el mensaje hacia el cliente con los campos necesarios. Usar las mismas structs en ambos programas y enviarlas y recibirlas usando read y write.

Discutir qué ventajas y desventajas tiene esta opción.