# Politecnico di Milano

## Middleware Technologies for Distributed Systems

## 23rd of September 2015

**Rules:**

- **You are not allowed to use books, notes, or other material.**
- **You can answer in Italian or English.**
- **Total time for the test: 1.5 hours.**
- **Check the back of this page for any additional exercises.**

---

1. Implement in **Java** the Executor class. At construction time the class receives a reference to a Runnable instance. After being created it executes the Runnable instance once a second until method stop() is invoked. You can assume that the Runnable passed at construction time only requires a short time to be executed.

2. Write in **TinyOS** a module PacketCounter that implements the receive interface and provides the Read<uint16_t> interface. It counts the number of received packets and returns the count when asked for it. Optional: automatically reset the counter every 10 seconds. The following interfaces may help you:

```
interface Receive {
  event message_t* receive(message_t* msg, void* payload, uint8_t len);
}

interface Timer<precision_tag> {
   command void startPeriodic(uint32_t dt);
   command void startOneShot(uint32_t dt);
   command void stop();
   event void fired();
}

interface Read<val_t> {
   command error_t read();
   event void readDone(error_t err, val_t val);
}
```

3. Use RMI to expose a remote method capable of returning stock quote data given a stock quote symbol. The returned information should contain the stock's current (or last known) value, its opening value for the day, and its minimum and maximum values for the day.

   Provide both the server-side code required to implement and publish the service, and the client-side code required to access the remote method correctly.

4. What is WSDL, and what is its purpose? What are the main components of a WSDL file, and what information does each of these components provide?

5. Tower Records is an online music store that specializes in selling vinyl records. In order to better tailor their offerings the company keeps a detailed log of all the vinyl records that it sells. Each record sold produces a tuple of the kind <date, userID, artist, vinyl, genre, cost>. Date provides the moment in which the record was sold, in the format DD:MM:YYYY-HH:MM:SS. userID is a unique identifier of the person that bought the record, artist is the name of the artist or band that produced the record, vinyl is the name of the record, genre is the musical genre the record belongs to, and cost is the amount the record cost.

    1. Tower Records uses a "gain by record" metric to evaluate its artists. This metric is calculated per artist as the "total amount of money the artist generated over a given amount of time, divided by the total number of records sold by that artist in that same timeframe". Use Hadoop to provide the top ten artists in the 2014-2015 Christmas timeframe, that is from the 27th of November 2014 to the 31st of January 2015.

    2. Tower Records also wants to know how its business varies during the day. To do this it wants to analyze all the records sold in the year 2014, and it wants to calculate the average amount of money made per hour. More precisely, given that the store opens at 8am, it wants to know the average amount of money made between 8am and 9am (in 2014), the average amount of money made between 9am and 10am (in 2014), and so on.