

- Use a spell checker.
- This is the way you use “quotes.”
- Avoid paragraphs of one or two sentences. As a rule of thumb, paragraphs should be three to seven sentences.
- When you refer to a paper, mention the author(s).
- Instead of et al, use et al. (al. is short for alia).
- Instead of figure 3, table 2 and section 4, use Figure 3, Table 2 and Section 4.

Avoid references in abstracts or provide all details.

*In this addendum, we correct some typos and fill a gap in the proof of Theorem 21 of [F. van Breugel, C. Hermida, M. Makkai, J. Worrell. Recursively defined metric spaces without contraction. Theoretical Computer Science 380(1/2)(2007)143–163]. We reprove Theorem 21 and fill the gap by Lemmas 2–4 of this paper.*

- Instead of `/` (`$1$`), use `ℓ` (`$\ell$`).
- Instead of `i.e.`, use `i.e.\` to get the correct spacing.
- Instead of `\cite{paper}`, `\cite{another paper}`, use `\cite{paper, another paper}`.
- Instead of `line 28`, use `line~28` so that line and 28 are glued together.

- Provide as much information as possible (unless you are dealing with a page limit).
- Avoid abbreviations (unless you are dealing with a page limit).
- To get capitals in, for example, the title, use  
title = "A genetic algorithm for channel  
routing in {VLSI} curcuits",
- Mention all the authors (do not use et al.).

# Concurrent Object Oriented Languages

## Concurrent Programming in Java

<https://wiki.cse.yorku.ca/course/6490A>

- Brian Goetz, Tim Peierls, Joshua Bloch, Joseph Bowbeer, David Holmes and Doug Lea. Java Concurrency in Practice. Addison-Wesley, 2006.
- Mary Campione, Kathy Walrath and Alison Huml. The Java Tutorial. Lesson: Threads: Doing Two or More Tasks At Once.
- James Gosling, Bill Joy, Guy L. Steele Jr. and Gilad Bracha. The Java Language Specification. Third edition.

# Thread Creation

In Java, threads are created dynamically:

```
// create and initialize a Thread object
Thread thread = new Thread();
// start the thread
thread.start();
```

The class `Thread` is part of package `java.lang` (and, hence, does not need to be imported).

## Question

Develop a Java class called `Printer` that is a `Thread` and prints its name 1000 times.



# Two Concurrent Printers

## Question

Develop an app that creates two `Printers` with names 1 and 2 and run them concurrently.

# Java Only Supports Single Inheritance

The following is **not** allowed in Java.

```
public class Printer extends Applet, Thread
```

# Thread Creation

```
// create and initialize a Runnable object
Runnable runnable = new ...();
// create and initialize a Thread object
Thread thread = new Thread(runnable);
// start the thread
thread.start();
```

The interface `Runnable` is part of package `java.lang` (and, hence, does not need to be imported).

# Runnable is an Interface

In Java, you cannot create instances of an interface.

```
public class Printer implements Runnable
{
    ...
}
```

The assignment

```
Runnable printer = new Printer();
```

is valid since the class `Printer` implements the interface `Runnable`.

## Question

Develop a Java class called `Printer` that implements `Runnable` and prints the thread's name 1000 times.

# Semaphores

The Java class library contains the class  
`java.util.concurrent.Semaphore`.

The method `acquire` represents the P-operation and the  
method `release` represents the V-operation.

# The Readers-Writers Problem

```
int readers = 0;  
semaphore mutex = 1;  
semaphore token = 1;
```

# Reader

```
P(mutex);  
readers++;  
if (readers == 1)  
    P(token);  
V(mutex);  
read  
P(mutex);  
readers--;  
if (readers == 0)  
    V(token);  
V(mutex);
```



```
P(token);  
write  
V(token);
```