

Introduction to operating systems

Module 0

Operating systems 2019

1DT044, 1DT096 and 1DT003

Karl Marklund

Course responsible lecturer for OS (1DT044)
and OSPP (1DT096)

Lecturer for the OS part of DSP (1DT003)

Contact

Primarily use **Piazza** for questions where the answer may be
of interest to other students

Only contact me directly if Piazza is not suitable

karl.marklund@it.uu.se

Room P1440

Piazza

<https://piazza.com>

<http://bit.ly/os-ospp-dsp-2019-piazza>

Announcements from the teaching staff

Post questions

Discuss questions with students and teaching staff

Tag all questions with module number and other appropriate tags

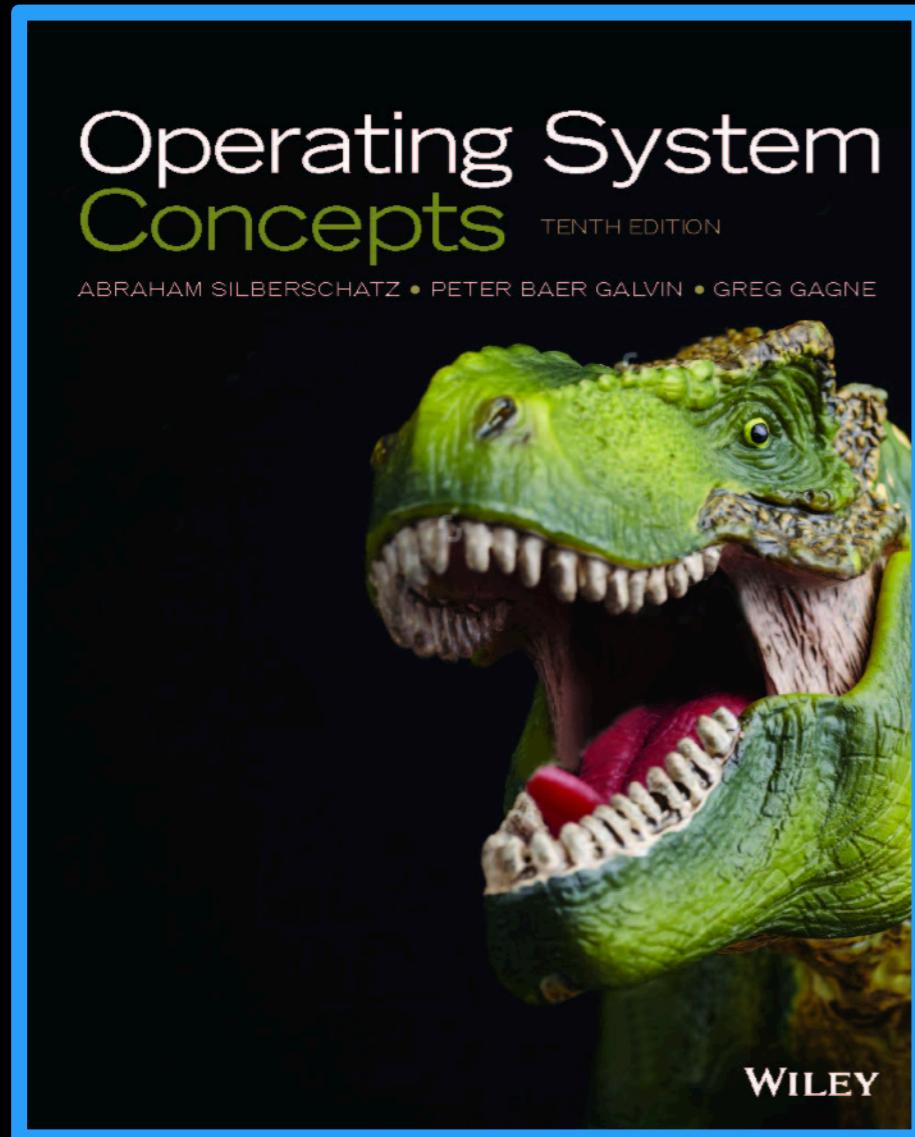
Slides for all lectures will be
added to the following
GitHub repositories right
before the lecture starts

<https://github.com/uu-os-2019/os-ospp-slides>

<https://github.com/uu-os-2019/dsp-slides>

No text book required

There is no required text book for the OS content of the courses. If you would like to have a text book, the following book is highly recommended.



Operating System Concepts

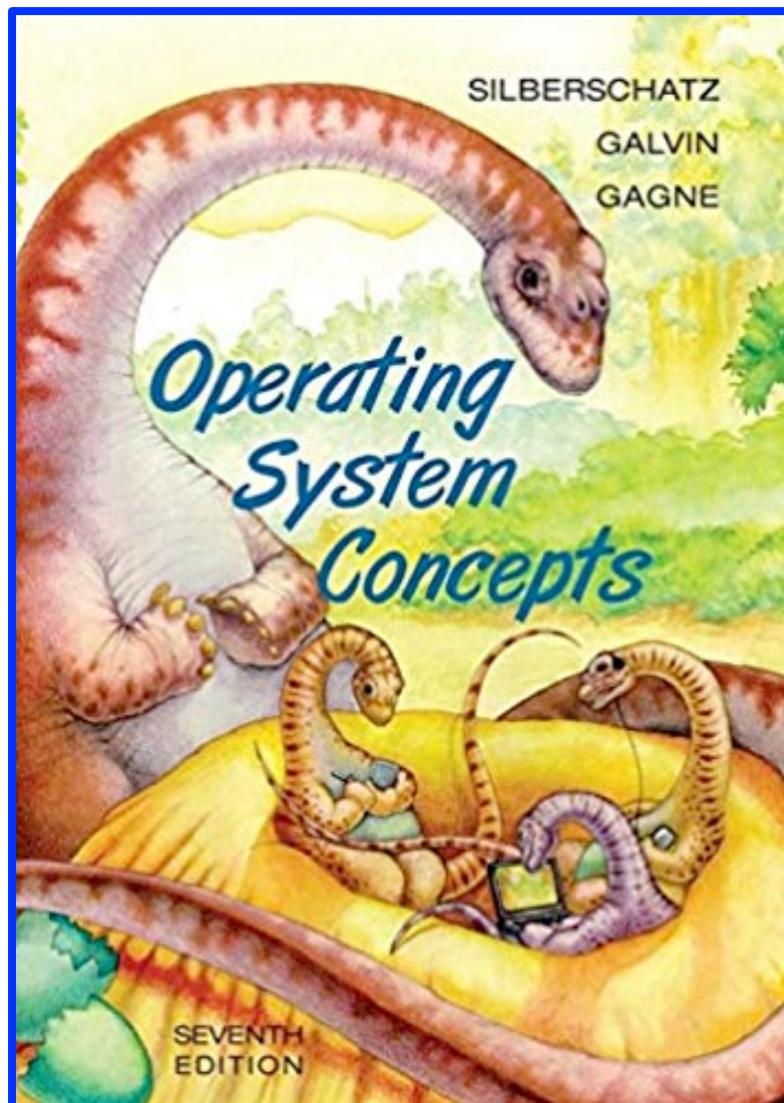
10th edition

Avi Silberschatz

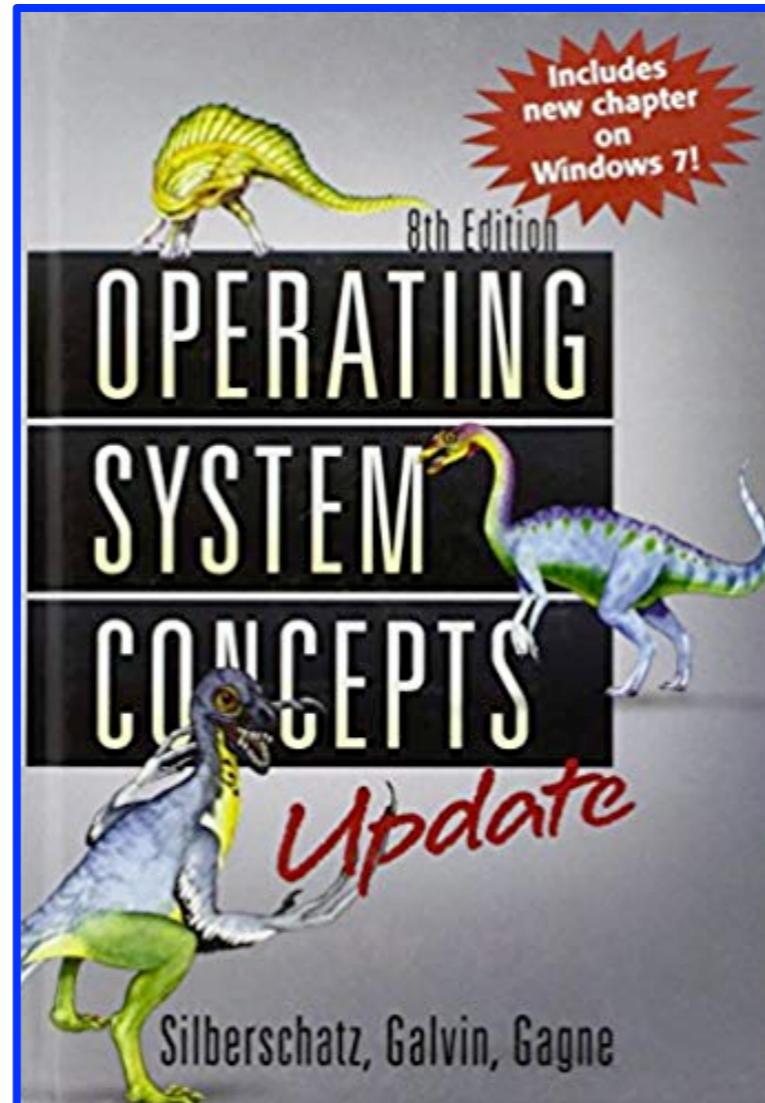
Peter Baer Galvin

Greg Gagne

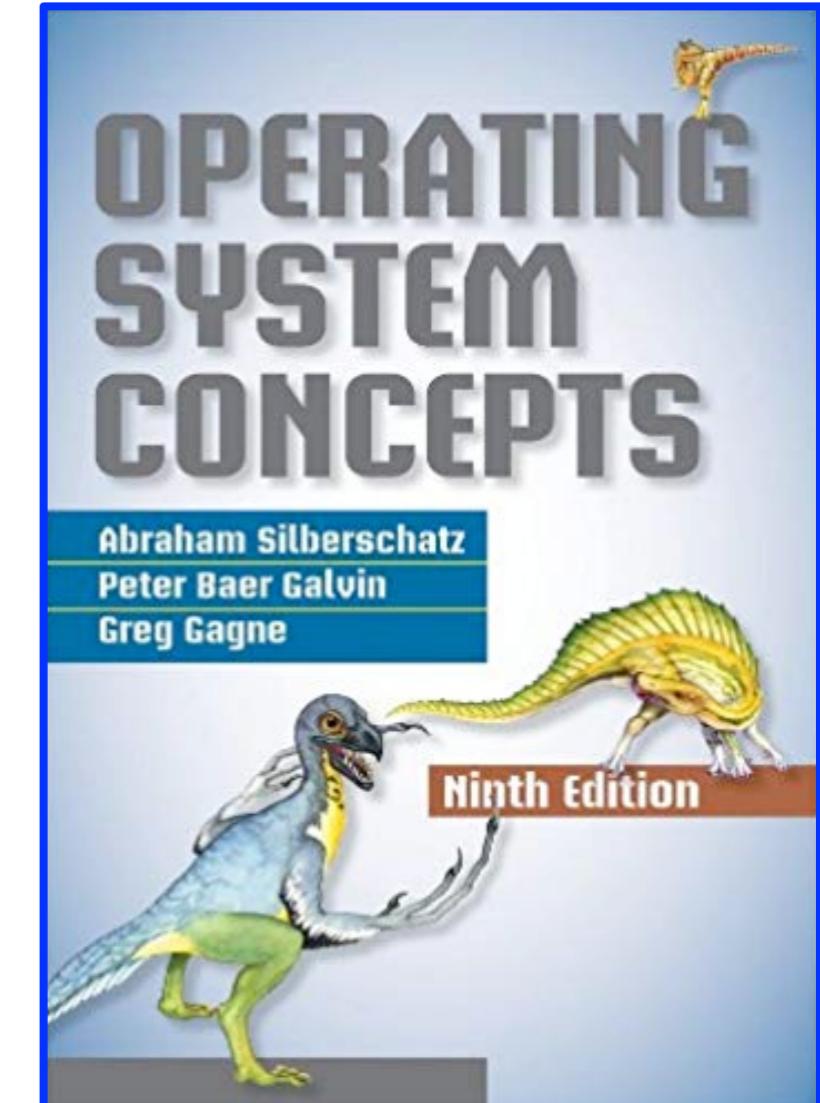
Older editions will do just as fine as the latest
10th edition.



7th edition



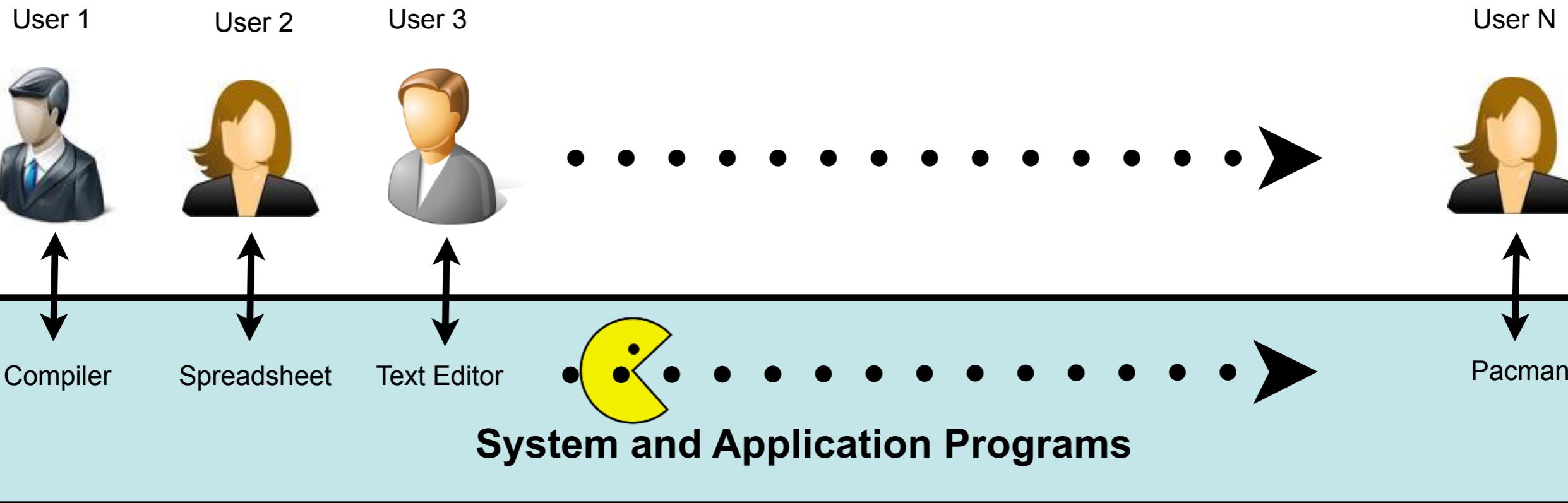
8th edition



9th edition

Operating Systems

- ★ What is an operating system?
- ★ Why is this interesting?
- ★ Why should we learn about this?



Operating System

Controls the hardware and coordinates its use among the various application programs for the various users.

Computer Hardware



Learning about operating systems is both useful and fun

You leverage what you learned in other courses

- ★ Computer architecture
- ★ Assembly programming
- ★ C programming

System and Application Programs

Operating System

You will learn how an operating system provides an **environment for programs to execute** (on a single core CPU) seemingly "*at the same time*" sharing the CPU without the programs knowing about each other or knowing the details of the actual hardware.

Computer Hardware

After this course you will have a better understanding of what is happening "under the hood" and thereby **become** a much **better engineer** and/or **programmer**.

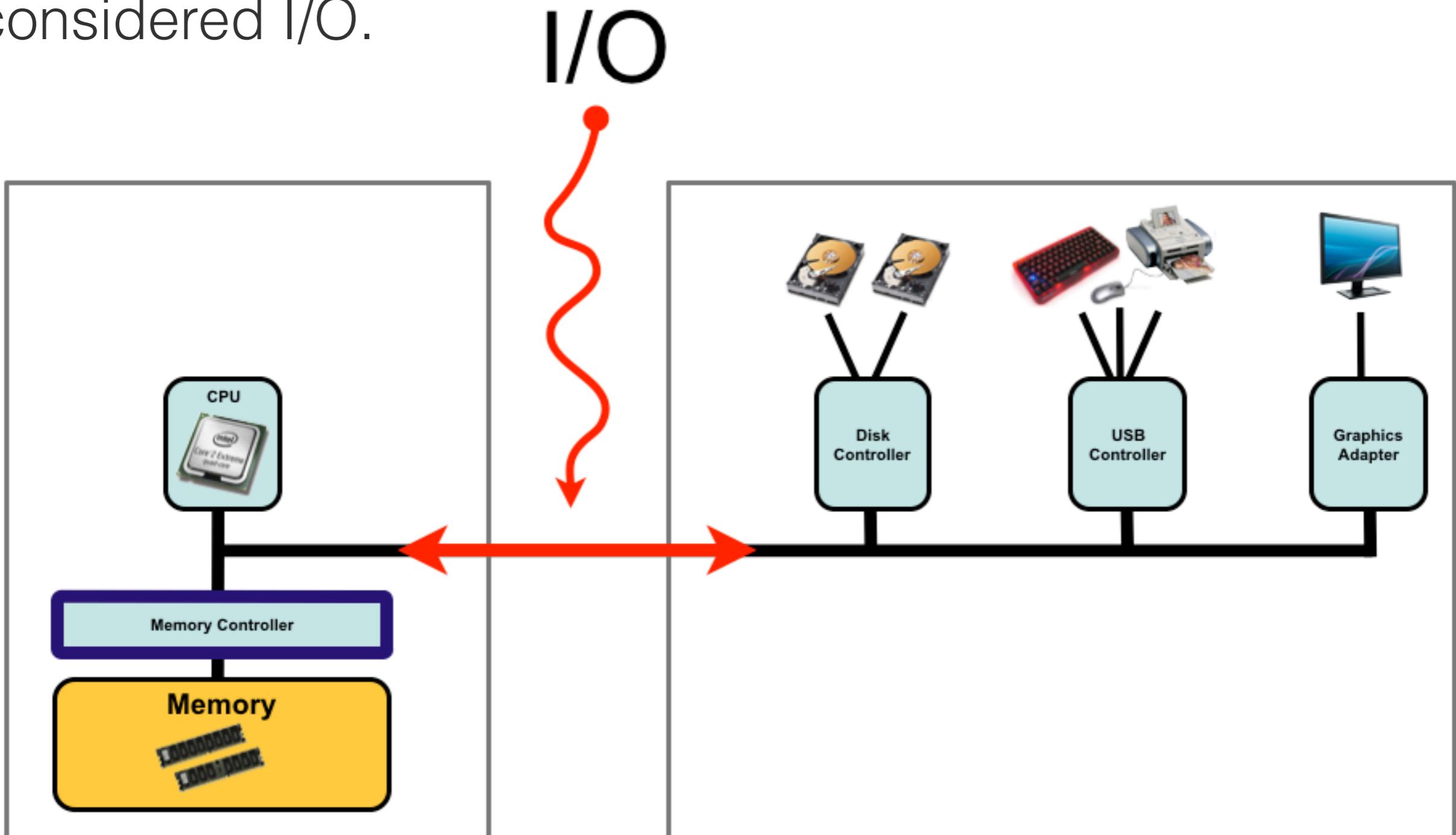
After completing this course you should have **enough knowledge** to **program** a small and **simple operating system** all by yourself if you put in some effort.

Input/Output (I/O)

In computing, input/output or I/O (or, informally, io or IO) is the communication between an information processing system, such as a computer, and the outside world, possibly a human or another information processing system.

Inputs are the signals or data received by the system and outputs are the signals or data sent from it.

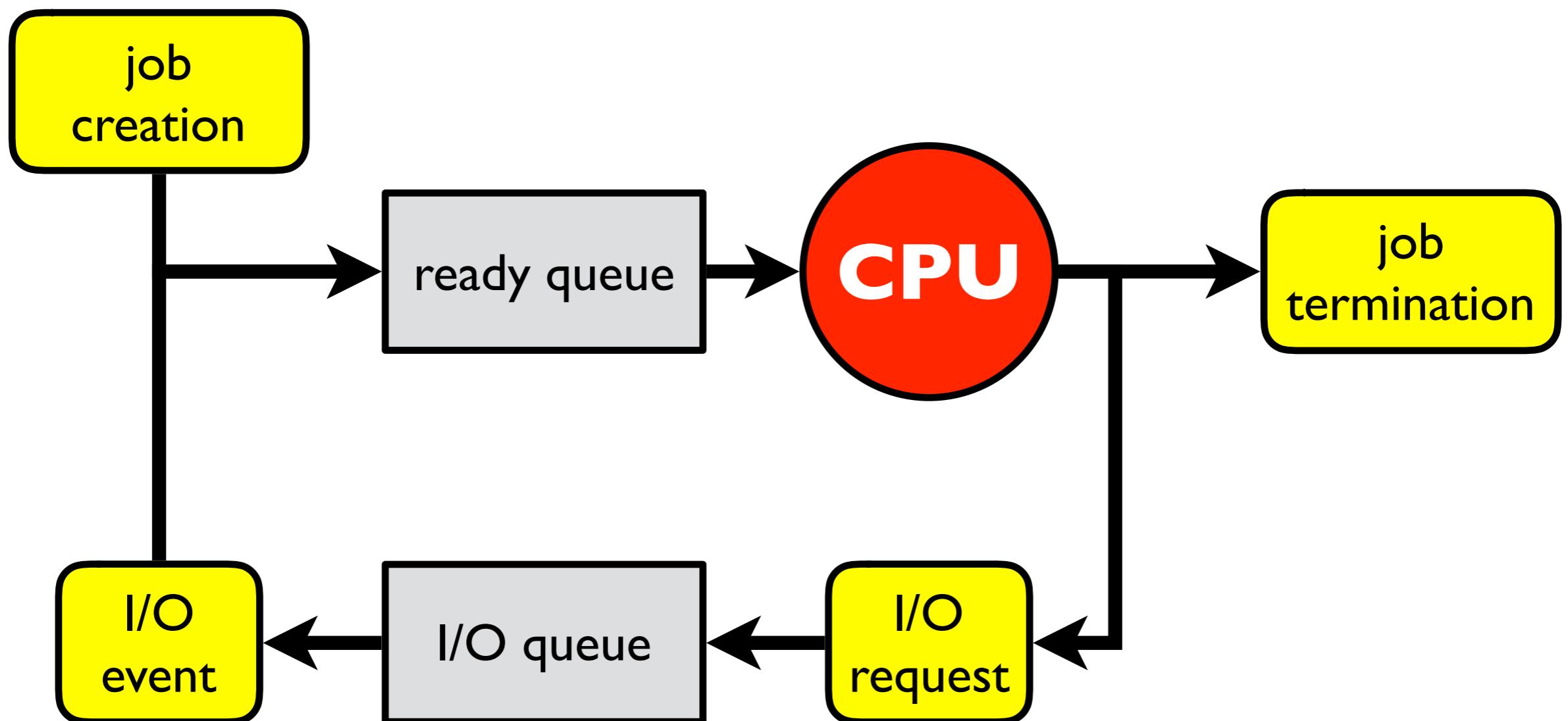
For our purposes, any transfer of information between the CPU/Memory and any of the external devices is considered I/O.



Compared to the speed of the CPU, I/O operations are very, very, **slow**.

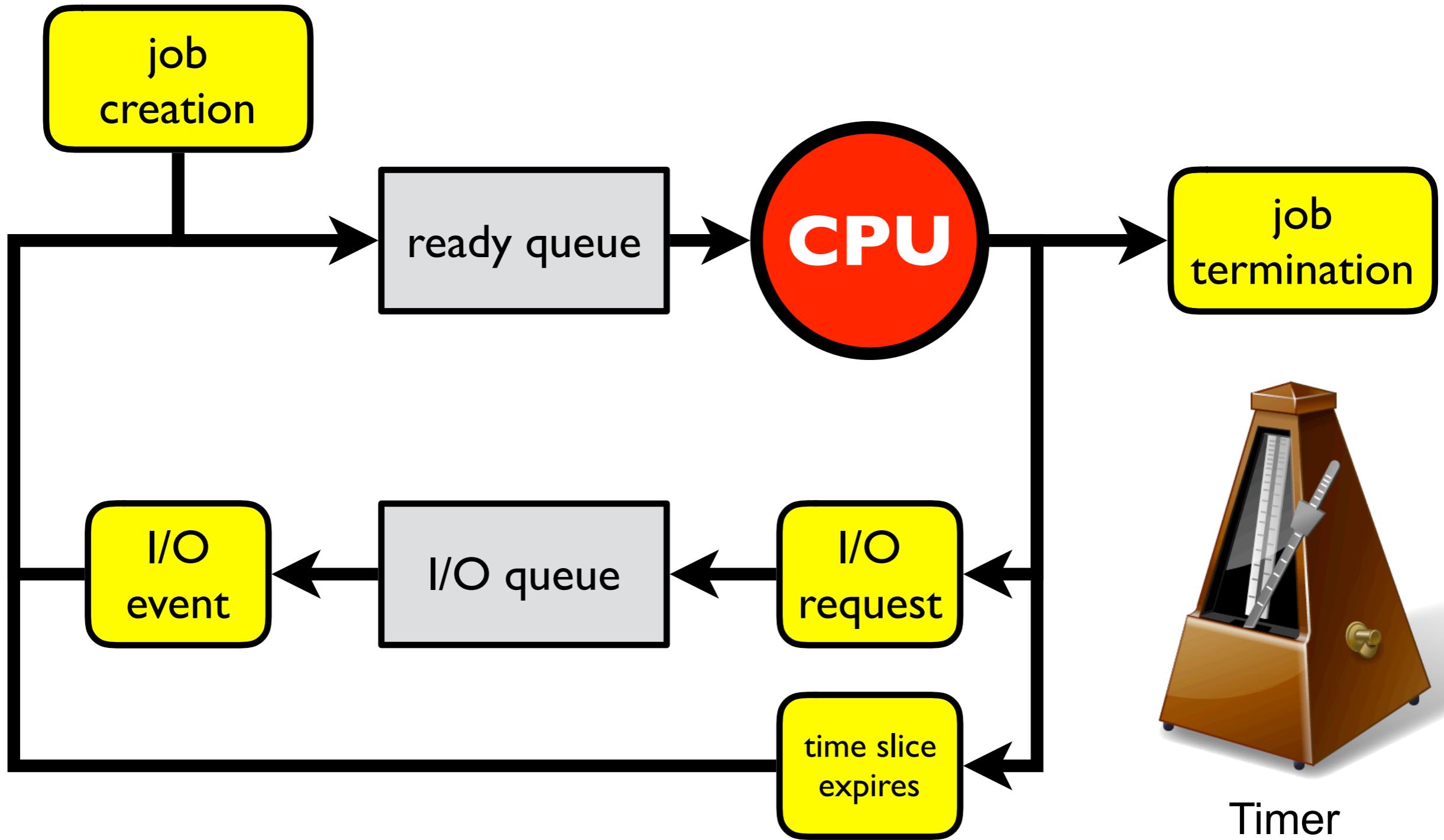
Multiprogramming

A schematic view of multiprogramming

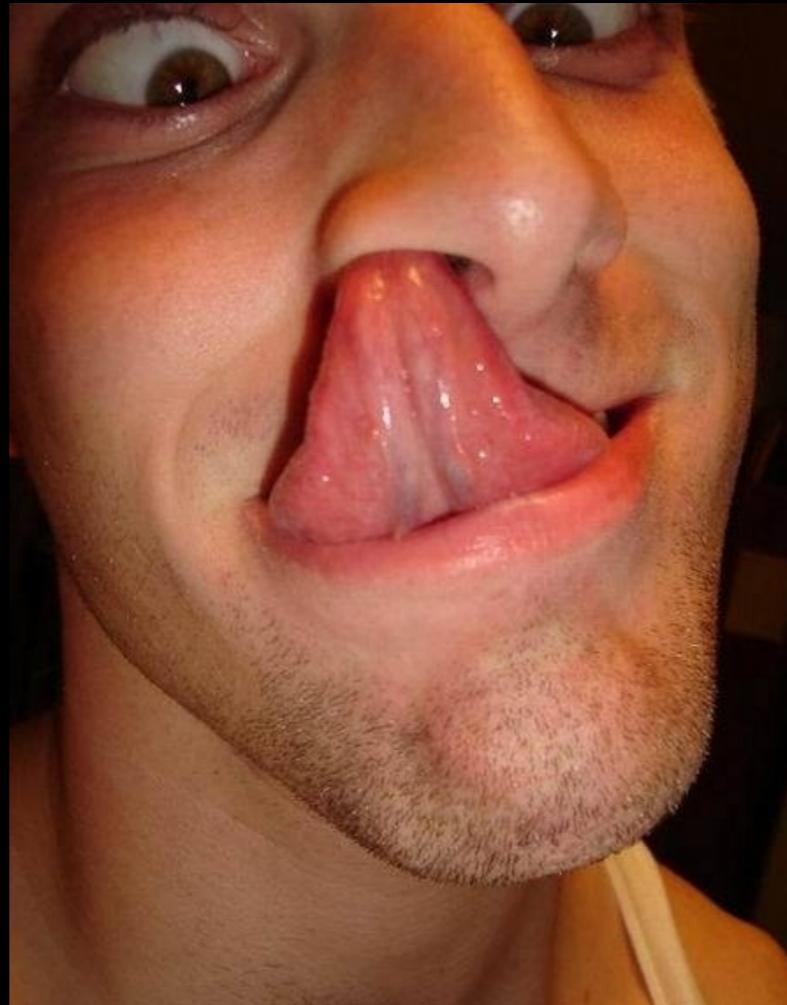


Multitasking

A schematic view of multitasking



What are you really good at?



How did you become good at this?



We learn in different ways

As a rough estimate the following conclusions can be made regarding how much you remember after a learning activity.



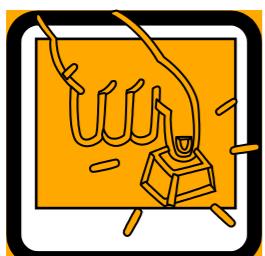
20 % after **listening**



30 % after **reading**



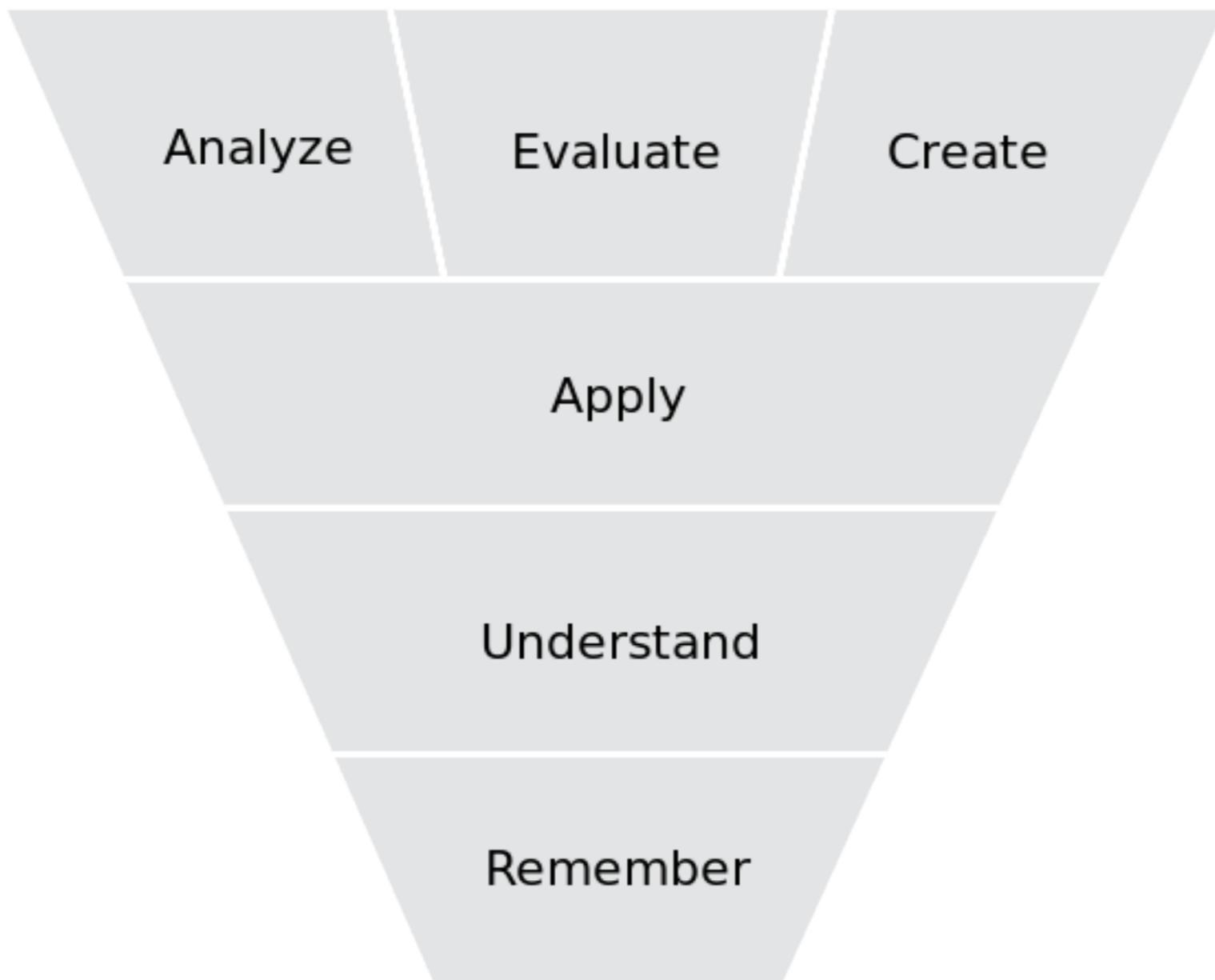
70 % after **talking**



90 % after **doing**

This tells us that listening to lectures and reading on your own may not be the most efficient way to learn ...

The cognitive domain



Thinking and learning

Seems like a good idea to combine these two models of learning and thinking.



20 % after listening



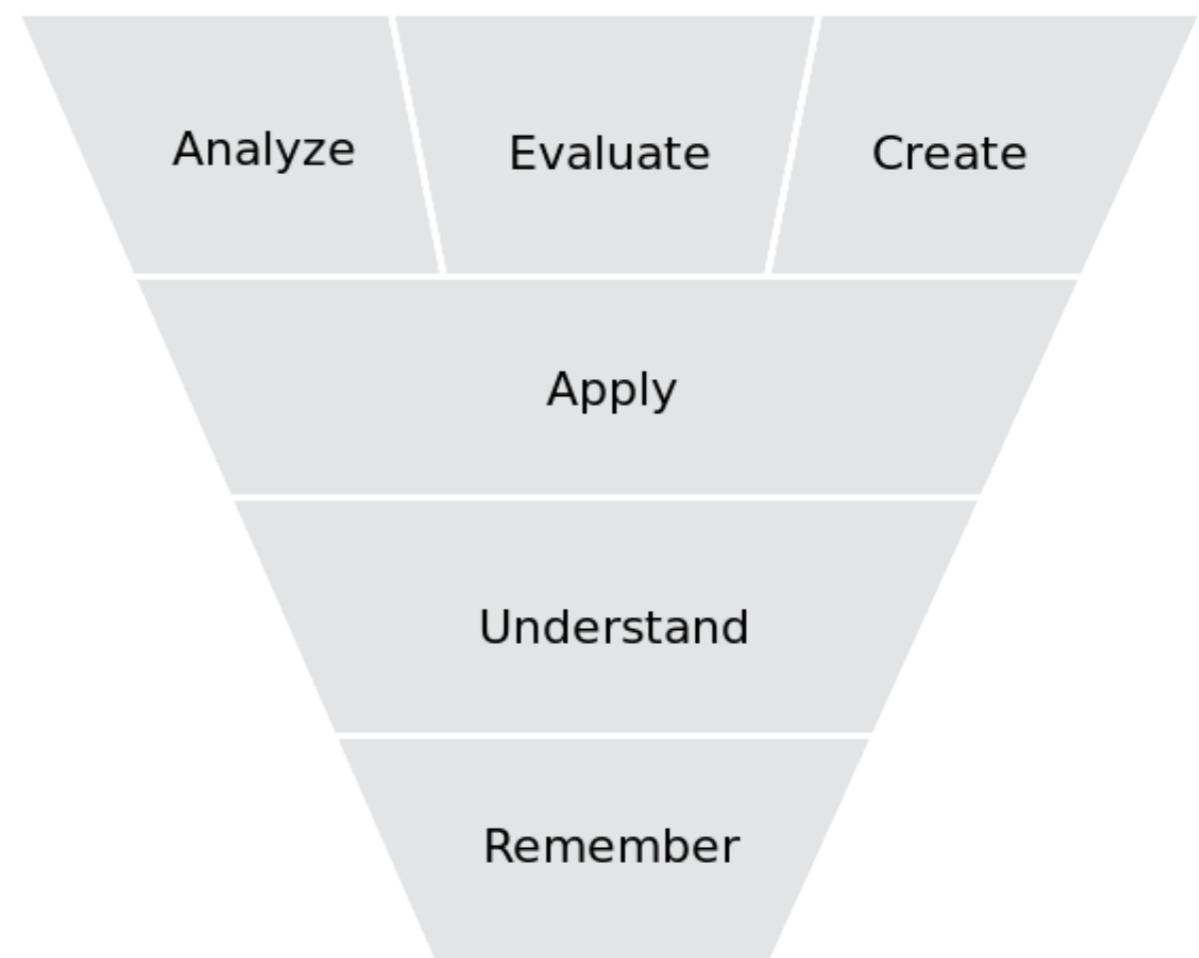
30 % after reading



70 % after talking



90 % after doing



Learning and examination activities

Teaching and examination is divided into four groups of activities: **learning, examination, higher grade and retake.**

- ★ **Participation in all learning activities** is highly recommended but **not mandatory**.
- ★ **Participation is mandatory** for all **examination activities**.
- ★ To pass the course you **must pass all examination activities**.

Learning

Type	Abbreviation	Activity
Learning	L	Lecture
Learning	W	Workshop
Learning	T	Tutoring
Learning	Tx	Extra tutoring

Examination

Type	Abbreviation	Activity	Grading scale
Examination	S	Seminar	Fail/Pass
Examination	C	Code grading	Fail, 3
Higher grade	Ch	Higher grade code grading	Fail, 3, 4, 5
Examination	P	Presentation	Fail/Pass
Examination	E	Final written exam	Fail, 3, 4, 5

Retake

Type	Abbreviation	Activity	Grading scale
Retake	Sr	Retake seminar	Fail/Pass
Retake	Cr	Retake code grading	Fail, 3
Retake	Pr	Retake presentation	Fail/Pass
Retake	Er	Retake written exam	Fail, 3, 4, 5

More information

More information about each activity can be found on the course home page.

Course homepages



OS (1DT044) and OSPP (1DT096)

<https://www.it.uu.se/education/course/homepage/os/vt19>

<http://bit.ly/os-ospp-2019>



DSP (1DT003)

<https://www.it.uu.se/education/course/homepage/dsp/vt19/>

<http://bit.ly/dsp-2019>

Course timelines



OS (1DT044) and OSPP (1DT096)

<http://bit.ly/os-ospp-2019-timeline>



DSP (1DT003)

<http://bit.ly/dsp-2019-timeline>

Module 0

Course overview and preparations.



A screenshot of a debugger interface showing assembly code. The code defines a string 'Hello World!' in the data section and implements a main function that prints it to the console and exits. The assembly code uses the AT&T syntax.

```
Run speed 20 inst/sec  
.  
.data  
msg: .asciiz "Hello World!"  
.text  
main:  
    li $v0, 4  
    la $a0, msg  
    syscall  
  
    li $v0, 1  
    syscall
```

```
#include <stdlib.h>  
#include <stdio.h>  
  
int main (void) {  
    printf("Hello, World!\n");  
    exit(0);  
}
```

To prepare for the tutorials and programming assignments you should make sure to go through the material in this section.

OS/OSPP

0 - Preparations

The shell and terminal

Linux

Git and GitHub

Mips and Mars

C programming

DSP

0 - Getting started

The shell and terminal

Linux

Git and GitHub

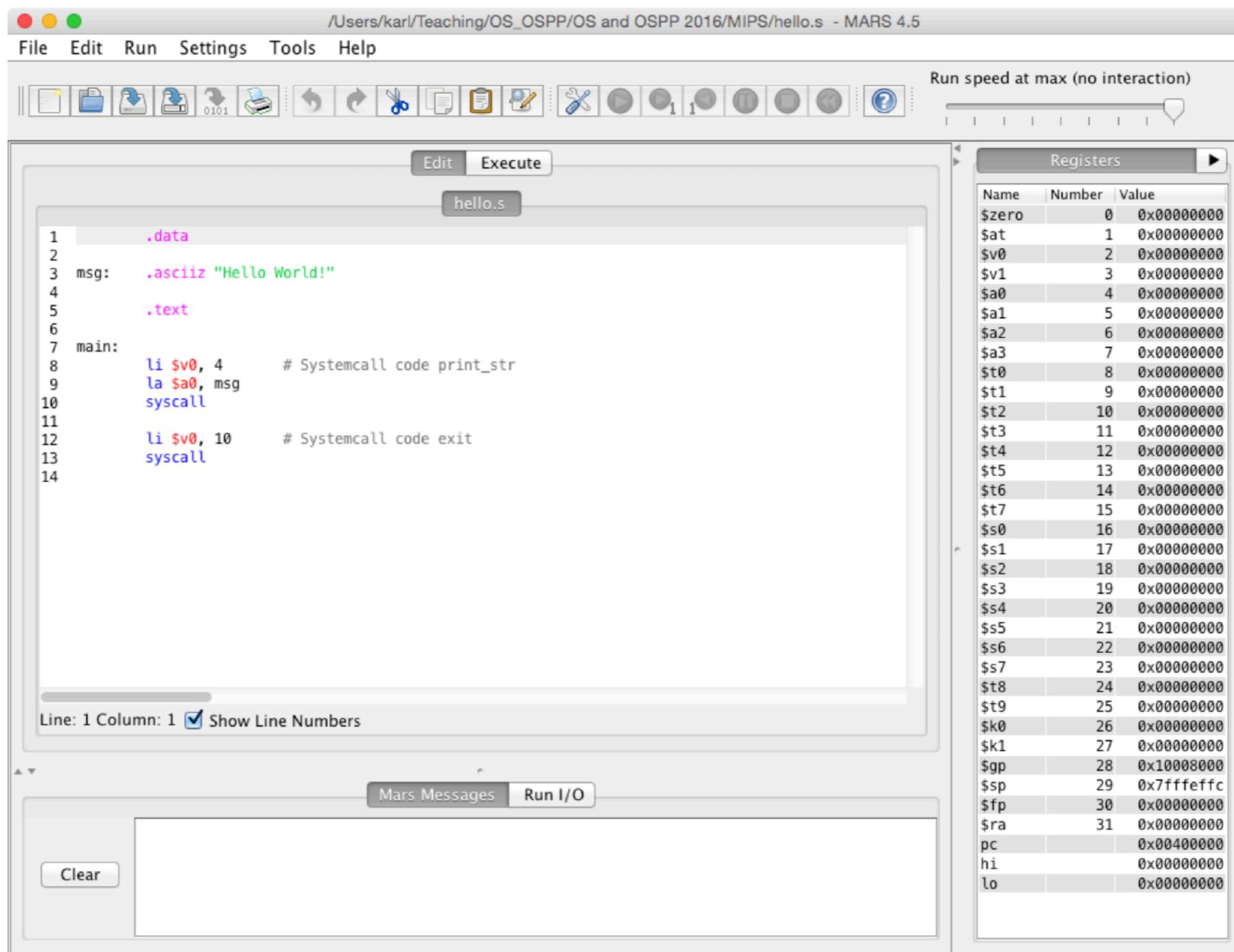
Mips and Mars

C programming

 Network paths

 Networking assignment

Mips and Mars



Mips and Mars

In order to study how the operating system interacts with the hardware, **Mips** assembly will be used.

To edit and execute Mips assembly programs we will use **Mars** (Mips Assembler and Runtime Simulator). Mars is available on the department Linux System.

Mars will run on any system (including Windows) as long as you have **Java installed**. If you prefer, you may **download** and install Mars on your private computer.



Mips memory layout



Clone repository



Introduction to Mars



Mips assembly examples

C programming

To study operating system concepts we will use the C programming language.

```
#include <stdlib.h>
#include <stdio.h>

int main (void) {
    printf("Hello, World! \n");
    exit(0);
}
```

C programming



Important concepts

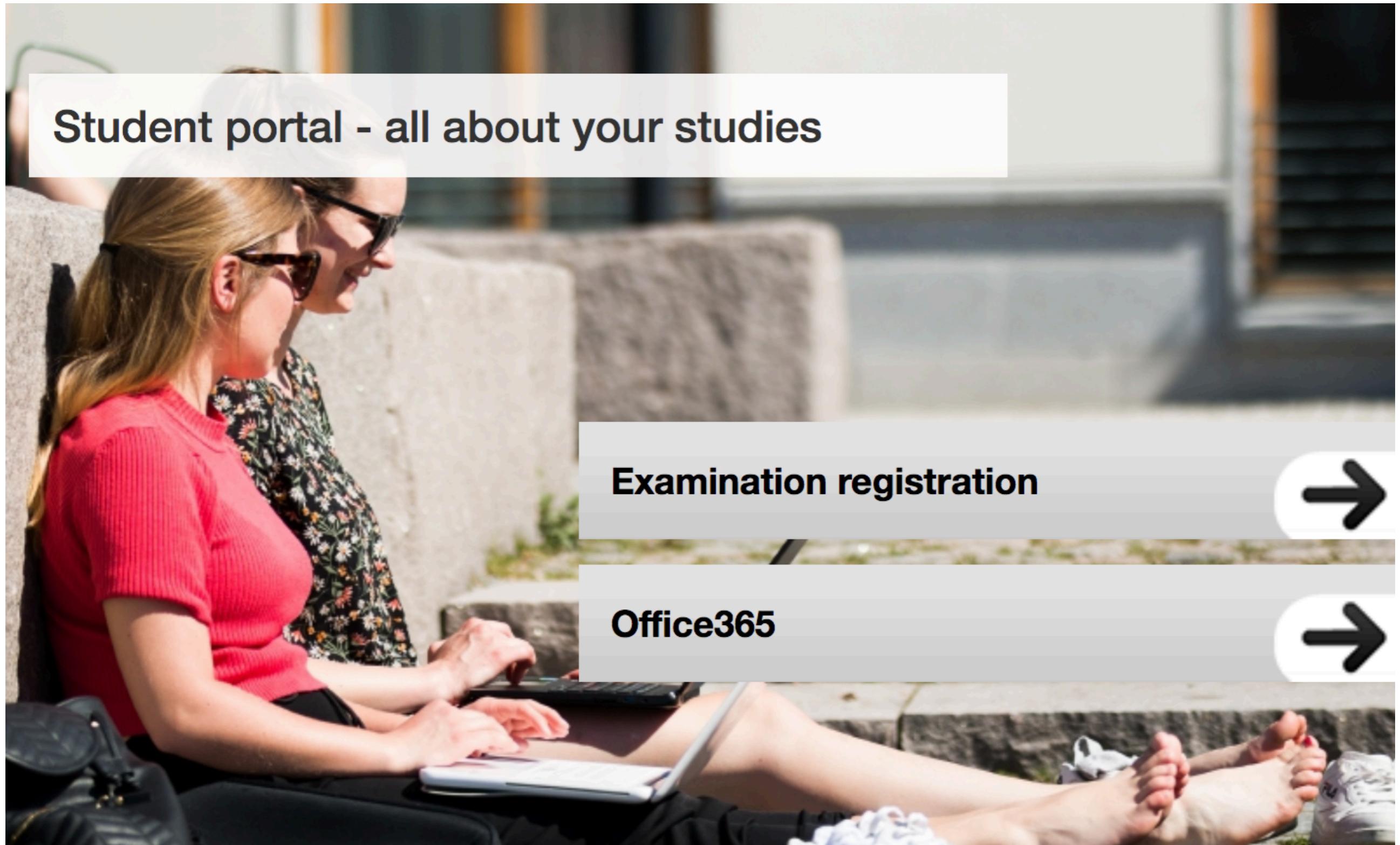


Learning resources



Programming exercise

www.studentportalen.uu.se



Student portal - all about your studies

Examination registration



Office365



Pairs and study groups

In the **Student portal** you find the section pairs and groups

- ▽ PAIRS AND GROUPS
 -  Pairs
 -  Study groups

Pairs and study groups

For **programming assignments** you will work in **pairs** of two students.

- You may choose to create a pair with another student by **signing up** for a pair - please pick the available pair with the lowest number.
- If you want to be assigned to a random pair by the teaching staff - sign up to the group **randomized**.
- If you **dropped** the course, **retake** the course but don't need to belong to appear etc - sign up to the group best describing your situation.
- **Deadline** 2019-01-23 08:01
- During **workshops, seminars** and the **case study** you will work in **study groups** of six students. The study groups will be formed by the teaching staff by randomly grouping 3 pairs together.

www.studentportalen.uu.se

HOW DO I REGISTER?

Information about when, where and how the student is to register is found on the course page in Student Portal.

As a new student, you must:

1. **Activate your student account** *

2. **Log in to the Student Portal**

Here you see information on what courses you are admitted to and how to register.

3. **Register for your course**

If you do not register, you may lose your place!

4. **Get Campus card** *

Tip! Information concerning how to register can be found by searching for the name or application code for the course via **Search** in the Student Portal.

**Are you
registered?**

Admitted students

Please web register in the student portal. Your registration will be transferred automatically to Ladok. The web registration will close one week after course start.

Students who are unable to web-register, should contact the Student Office (IT-kansliet): it-kansli@it.uu.se

Information for students admitted with conditions

Please web register in the student portal. Your registration will be transferred automatically to Ladok.

The student counsellors will handle all conditions in the students' admissions.

Non Admitted student

If you are on the waiting list we will contact you if there are seats available.
Please contact us for help with registration. it-kansli@it.uu.se

Exceptions:

- **Master students** should contact their programme counsellor (Liselott Dominicus): studievagledare@it.uu.se.
- **Exchange students** should contact Ulrika Jaresund, who is the exchange student coordinator at the IT Dept: ulrika.jaresund@it.uu.se.
- **Students with an older registration**, who want to re-register, should contact the Student Office (IT-kansliet): it-kansli@it.uu.se. You can only re-register if the course is not full.
- **Students who are placed on the waiting-list**, should wait for a possible answer (if any) via e-mail from the Student Office (IT-kansliet).

You are not able to take the exam, nor have any other results reported to Uppdok if you aren't admitted and registered for the course.

Sign up for final written exam

- You must sign up for the final written exam in the Student Portal.
- The exam signing-up system will not open until 2 weeks after the course start.
- The exam signing-up system in the Student Portal closes 12 days before the exam date.

Drop the course?

- Students who quit a course, must inform the Student Office (IT-Kansliet):
it-kansli@it.uu.se
- If less than 3 weeks have passed since the course started, the course registration will be removed.
- After 3 weeks a "course intermission" will be reported to UPPDOK instead.

The Student Office (IT-kansliet): ITC building 4, floor 2, room 4204, it-kansli@it.uu.se.

Exchange student coordinator: Ulrika Jaresund, ITC building 4, floor 2, room 4215, it-kansli@it.uu.se.

