Operating Systems

CSCI 355

Syllabus, Grading and other policies

- Syllabus and schedule are posted in Moodle.
  - SyllabusSchedule.pdf
- Grading policy
  - GradingPolicy.pdf
- Cheating policy
  - CheatingPolicy.pdf

- These are posted as assignments, and you have to submit a simple line like “read and understood”.

Syllabus and Schedule

- ..\Policies\SyllabusSchedule.pptx

Grading policy

- ..\Policies\GradingPolicy.pptx

Cheating policy

- ..\Policies\CheatingPolicy.pptx

Go to moodle to acknowledge

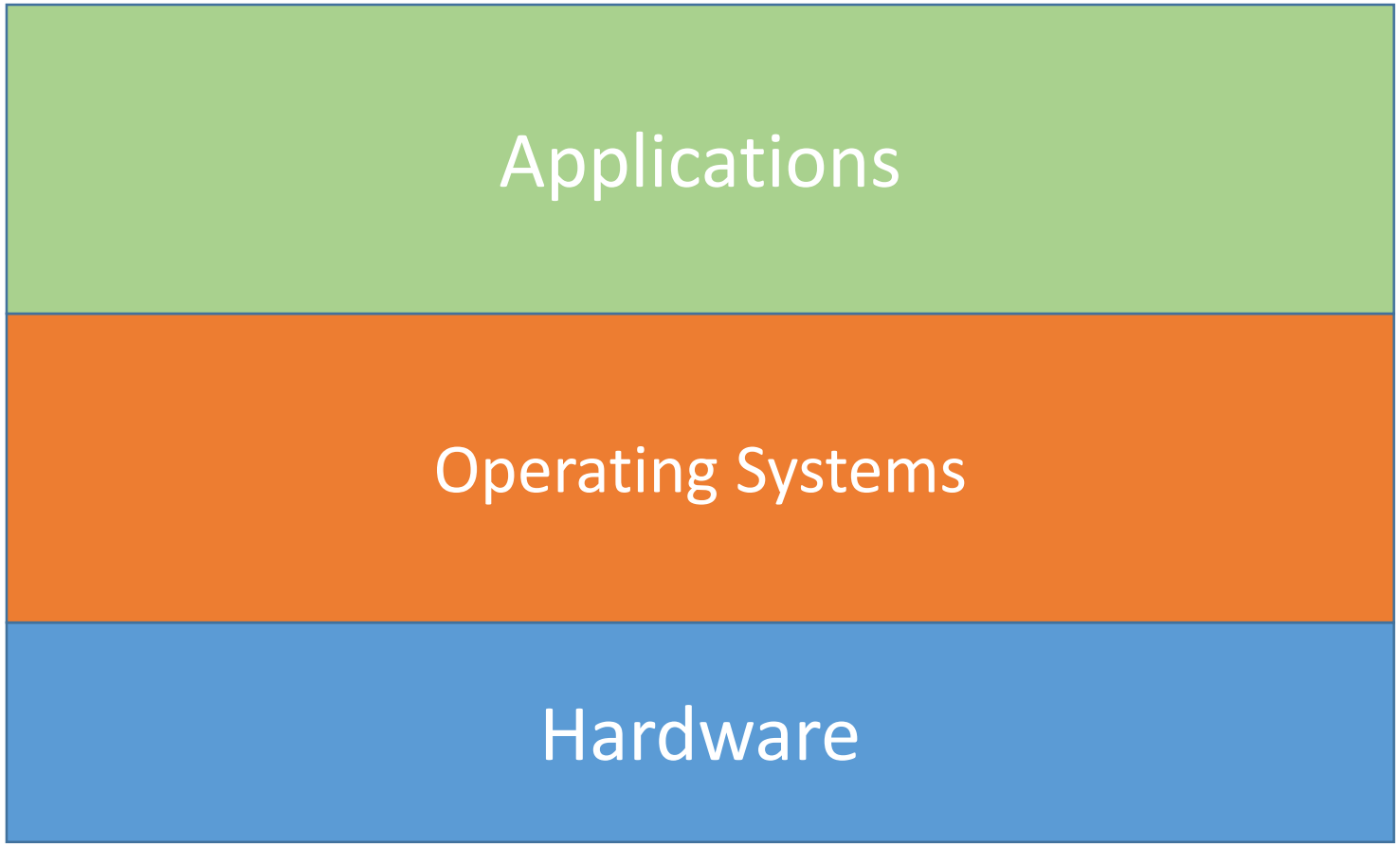
- Acknowledgement takes the form of submitting a statement like “read it”.

1. Read the syllabus
2. Read the cheating policy
3. Read the grading policy
4. Submission policy

Overview of Operating System

It’s a program that provide an environment for other programs to run

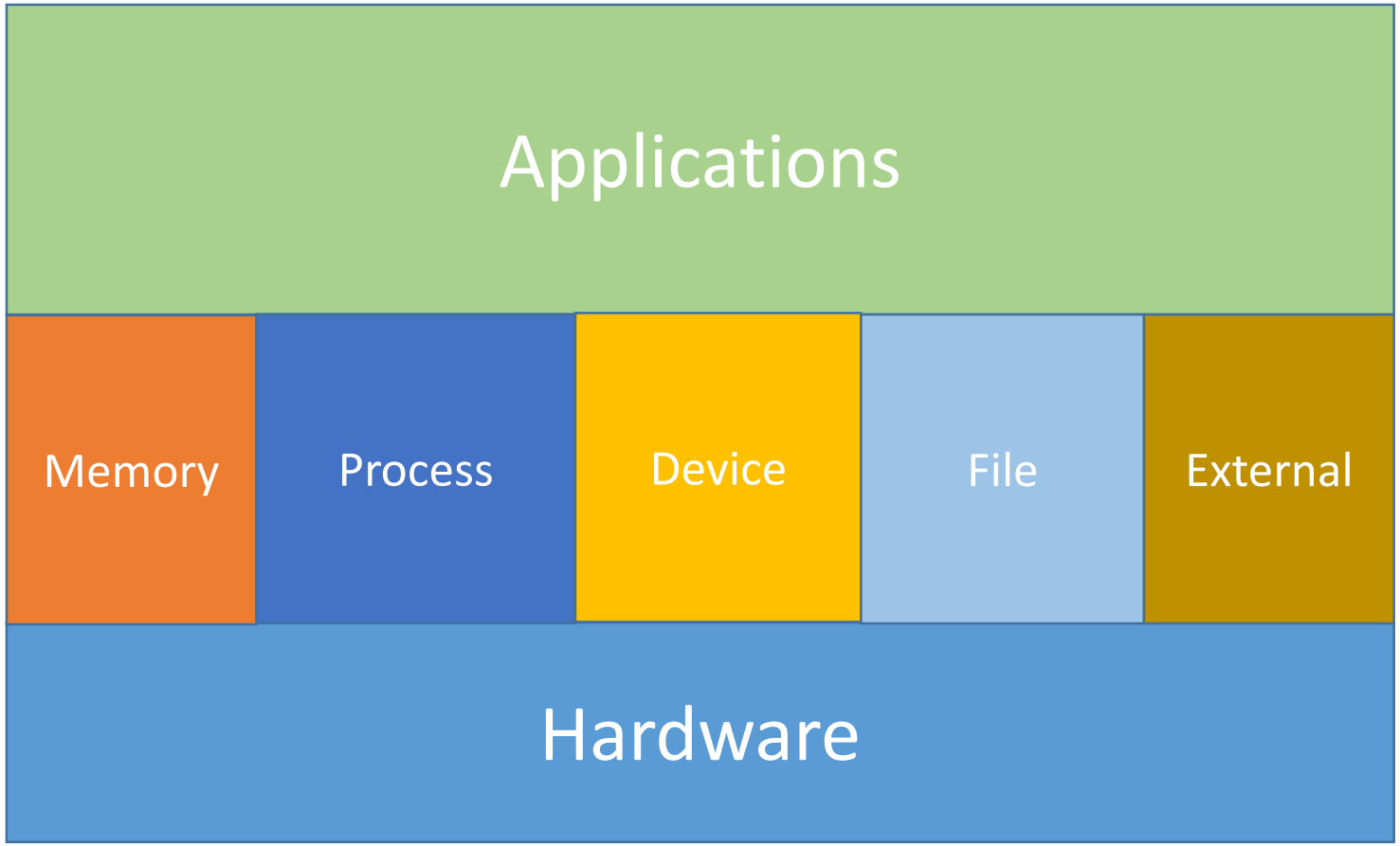
Operating Systems - three layer diagram



Manages hardware
And applications

Operating Systems Overview

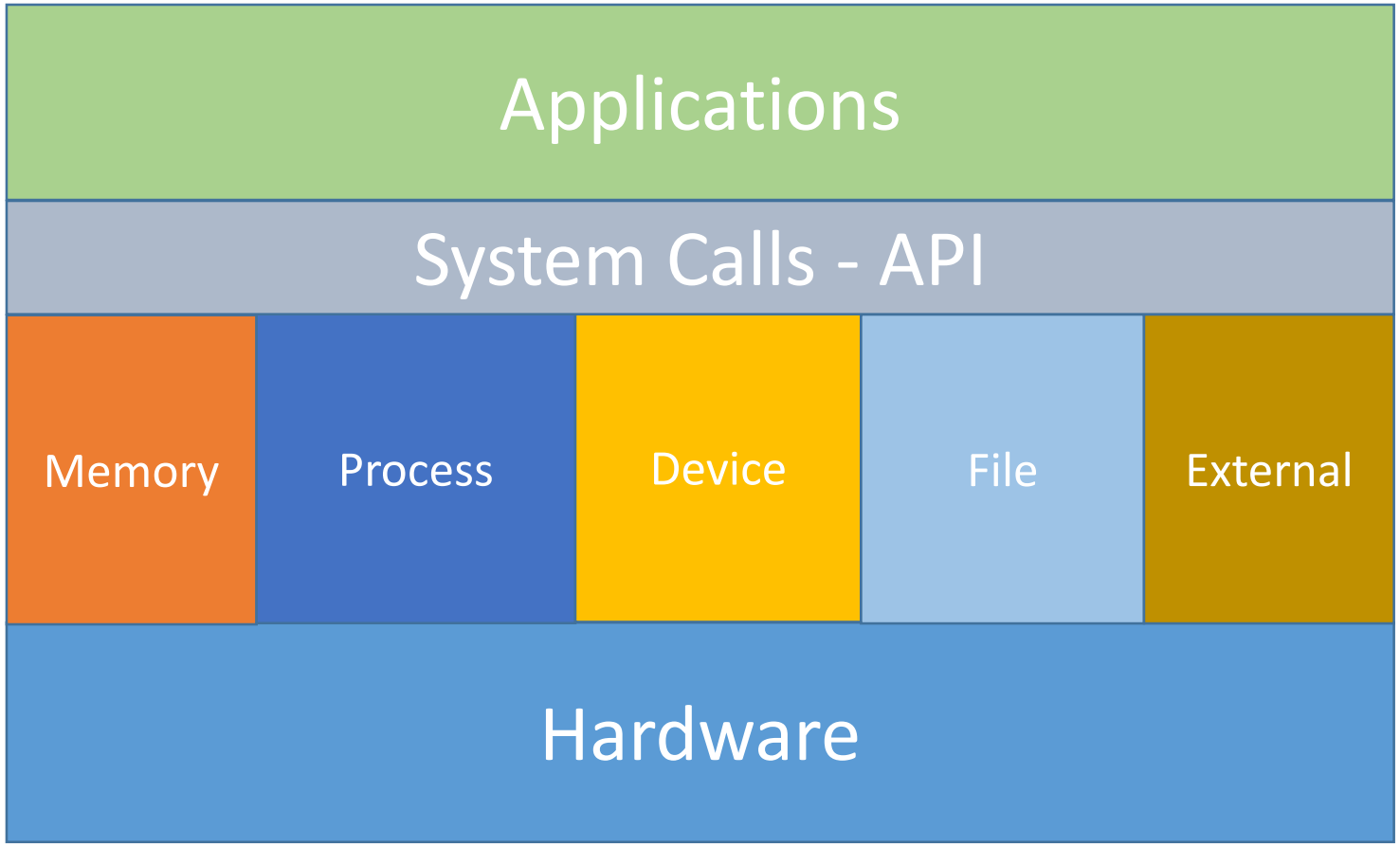
Operating Systems



Main management functions

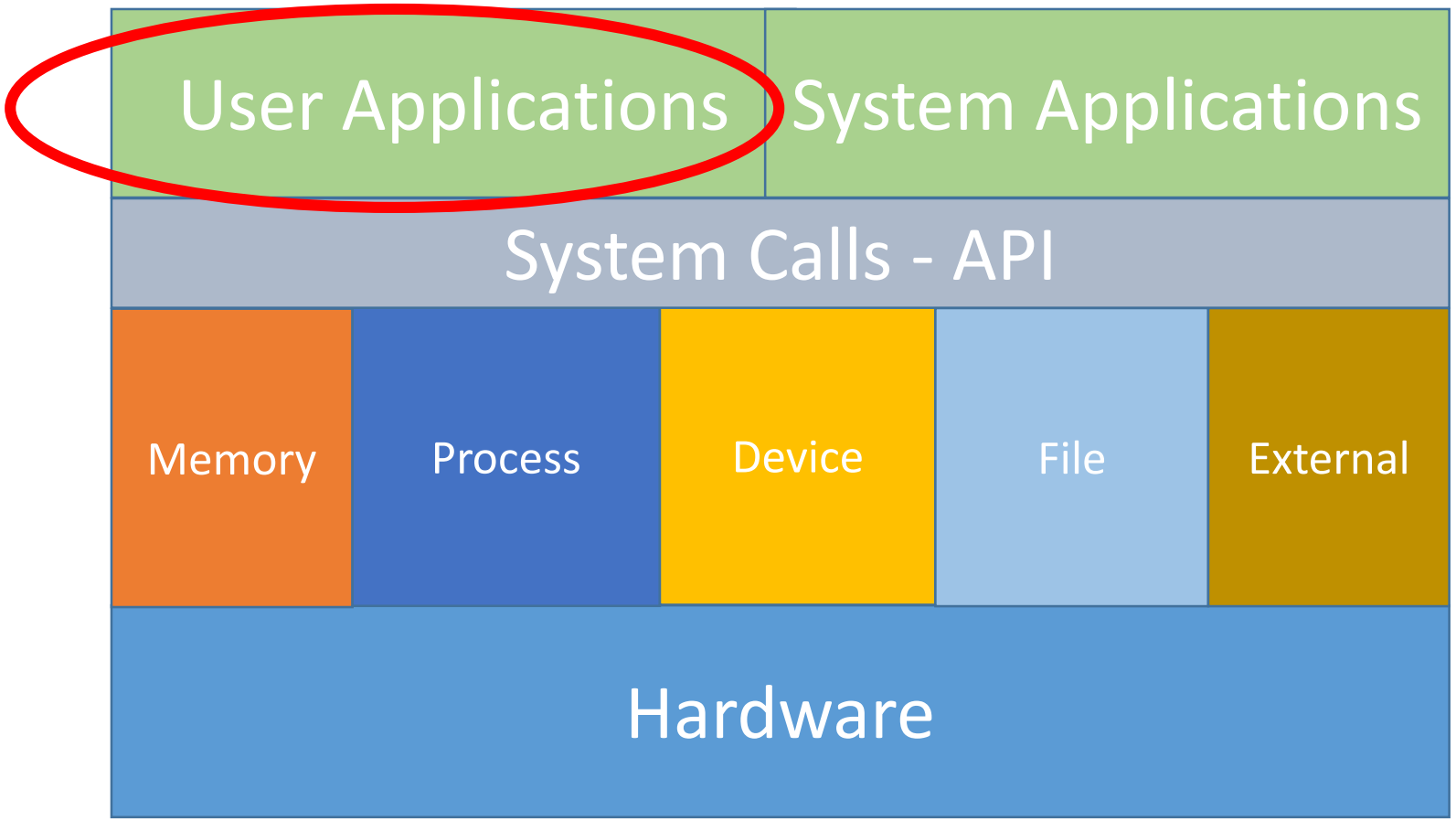
Operating Systems Overview

Operating Systems



Main management functions

Operating Systems Overview



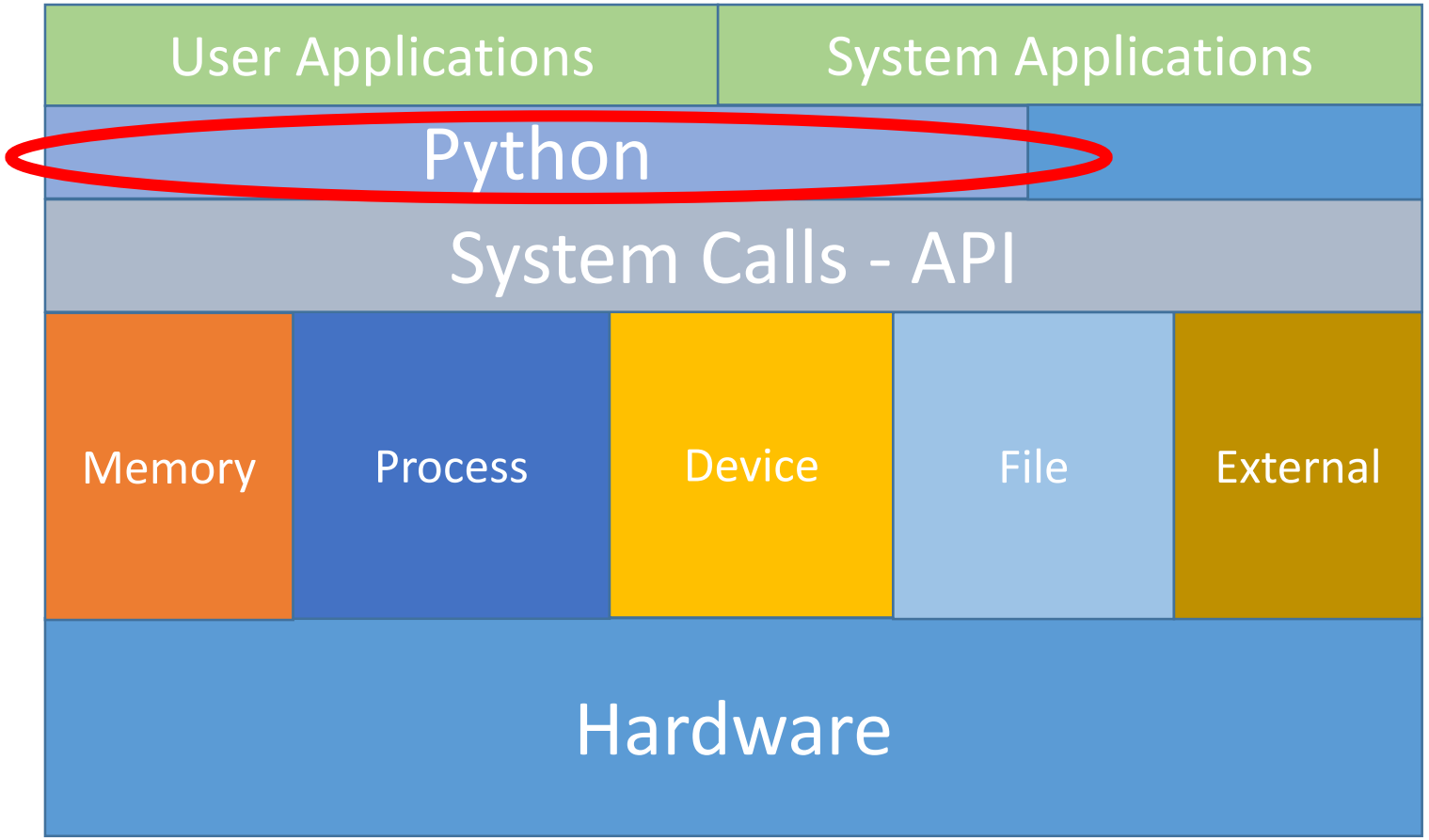
Eg. Terminal,
Disk management

Operating Systems

Main management functions

Operating Systems Overview

Python interface with OS



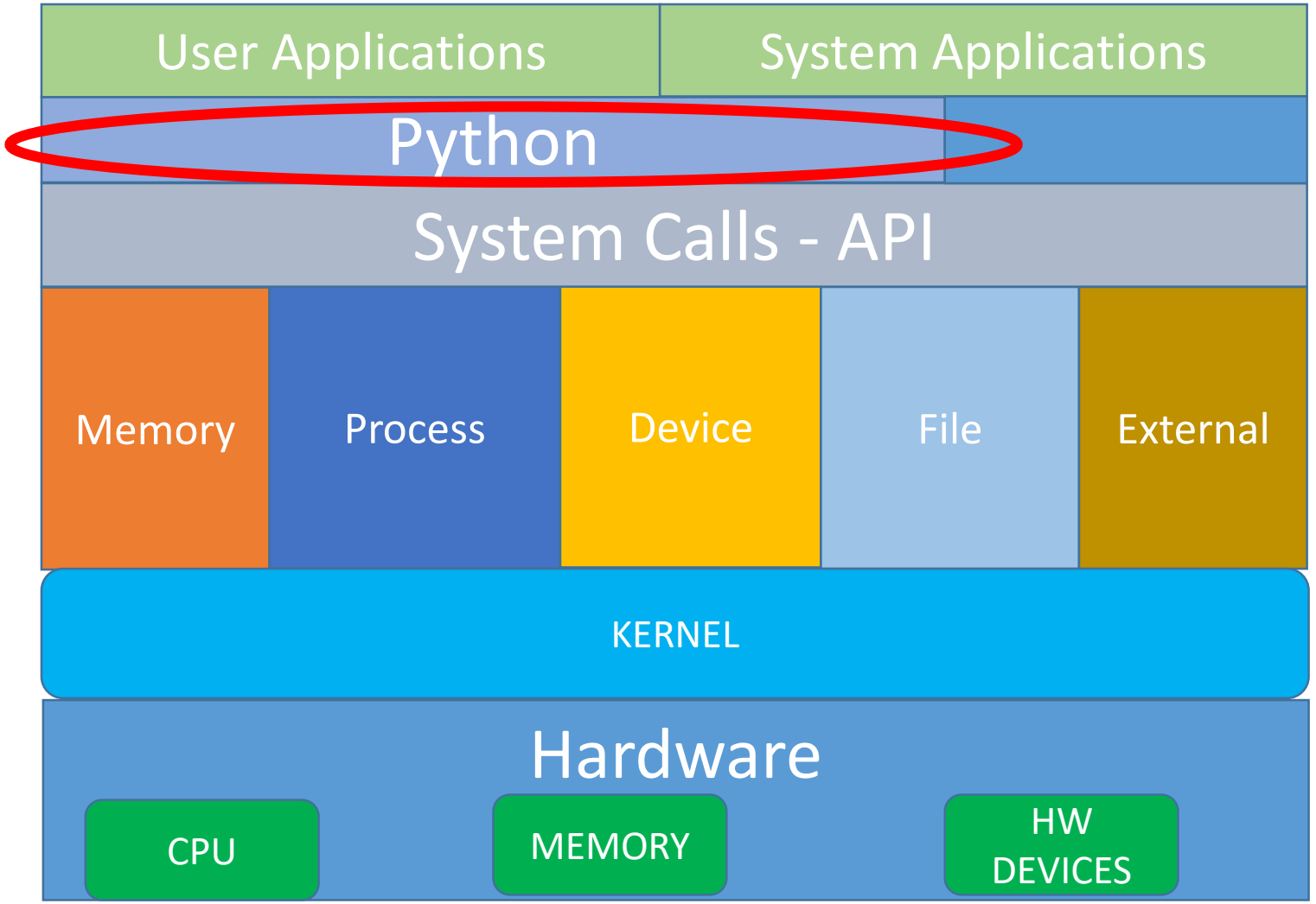
Eg. Terminal, Disk management

Operating Systems

Main management functions

Operating Systems Overview

Python interface with OS



Eg. Terminal, Disk management

Operating Systems

Main management functions

Software between applications and hardware
- **Interacts to hardware**
- **Call device drivers**

Some Operating System Definitions

- Resource Allocator – manages and allocates resources
- Control Program – controls the execution of user programs and operations of I/O devices
- Kernel – the one program running at all times

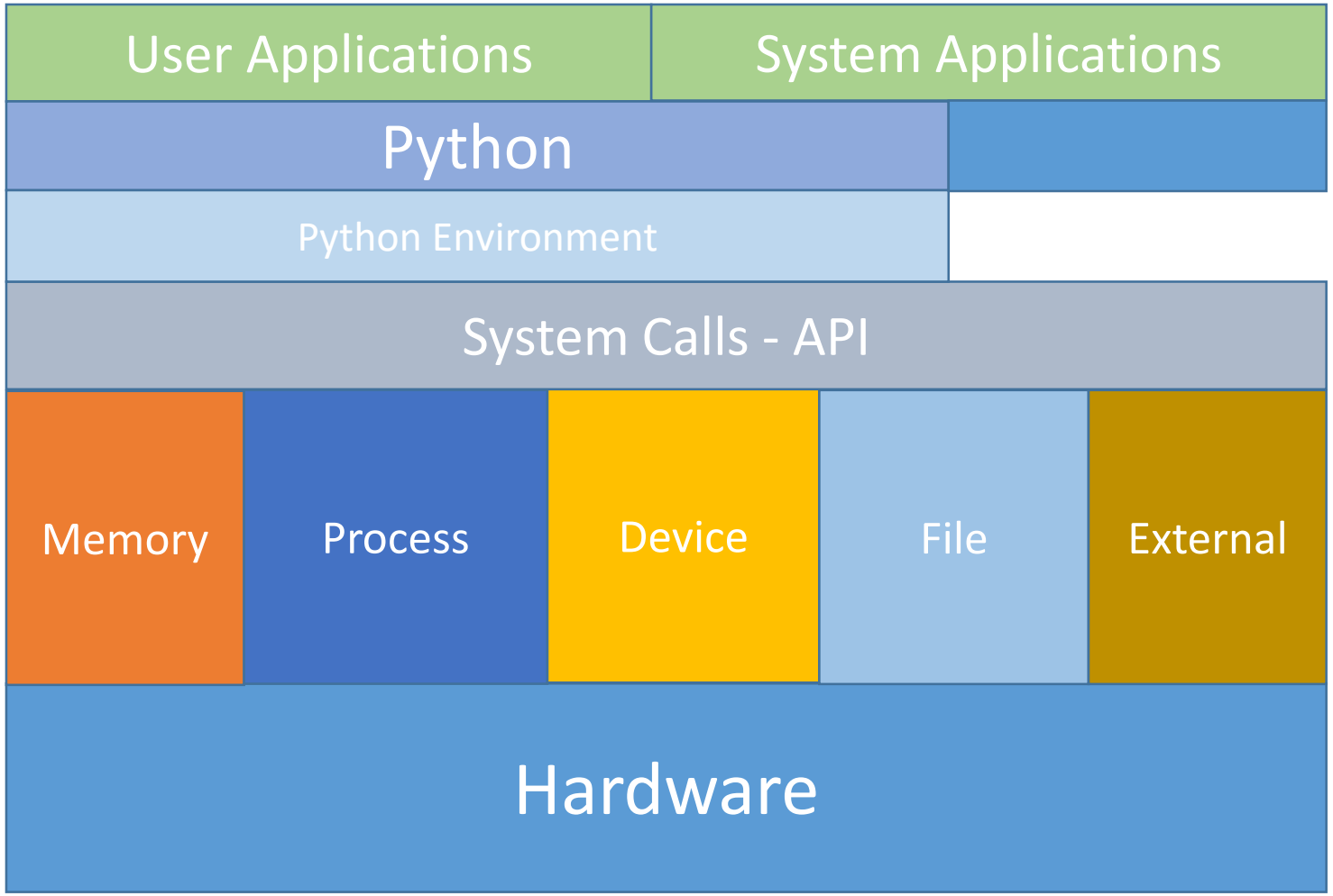
Before diving into OS management functionality

- Find out what the OS delivers
- We are going to make use of Python as a way to understand some of these OS functionality

- Want you to be proficient in Python
- BECAUSE being proficient in one language like Python
  - Will make it “much” easier to learn and work in other languages like
    - Java, C, JavaScript, etc

Python Environment

Python interface with OS



Eg. Terminal, Disk management

Operating Systems

Main management functions

MS VS Code for python

- Download:
  - https://code.visualstudio.com/
  - Do not download Python as you should make use of Python from Anaconda3

- Setup and tutorial:
  - https://code.visualstudio.com/docs/python/python-tutorial

Configuration (optional)

```
{
"name": "Python: Current
File",
"type": "python",
"request": "launch",
"program": "${file}",
"stopOnEntry": true
},
```

Test code and debugger

Hello.py

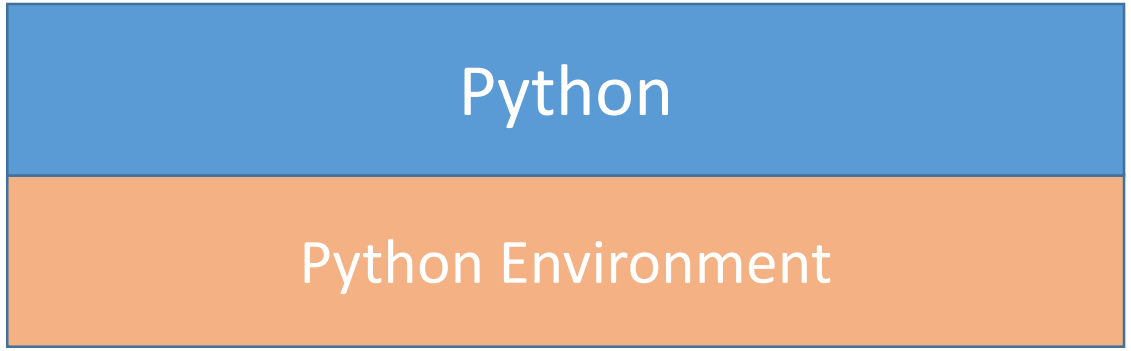
Sys-1.py

```
import sys

def get_platform():
platforms = {
'linux1' : 'Linux',
'linux2' : 'Linux',
'darwin' : 'OS X',
'win32' : 'Windows'
}
if sys.platform not in platforms:
print("not in platforms")
return sys.platform
return platforms[sys.platform]

print(get_platform())
```

Python Environment



Python seems to know how to import stuff,
and pip know how and where to install

Sounds like magic, and you need to know how this magic is done

Notes on written assignment

- A diagram is usually required
- Description should be specific [be very technical]
  - Say which files it is using
  - Which directories or folders it is using
  - Which environment variables
  - Enough details so that you can build an “import” feature – be “dumb”
  - For example:
    - Import
      - Which environment variables
      - Which directories or folders
      - Where does it look first, and where last
      - When it “import” where does it keep references
- Due Date: usually before the next class

Presentation of written assignments

- In the next class, two or three of you will present your answers to the class.
  - I will let you know in advance
  - You can show your diagrams or write on the whiteboard
  - Presentation time == 5 minutes

Moodle assignments naming convention

- Three types of assignments:
  - Written
  - Class
  - Code
- Moodle naming convention
  - <type><-><month-3 letter><-><day><-><a|b|c>
    - Written-Aug-29-b
    - Class-Aug-27-a
    - Code-Aug-31-a

Python Environment – Written Assignment

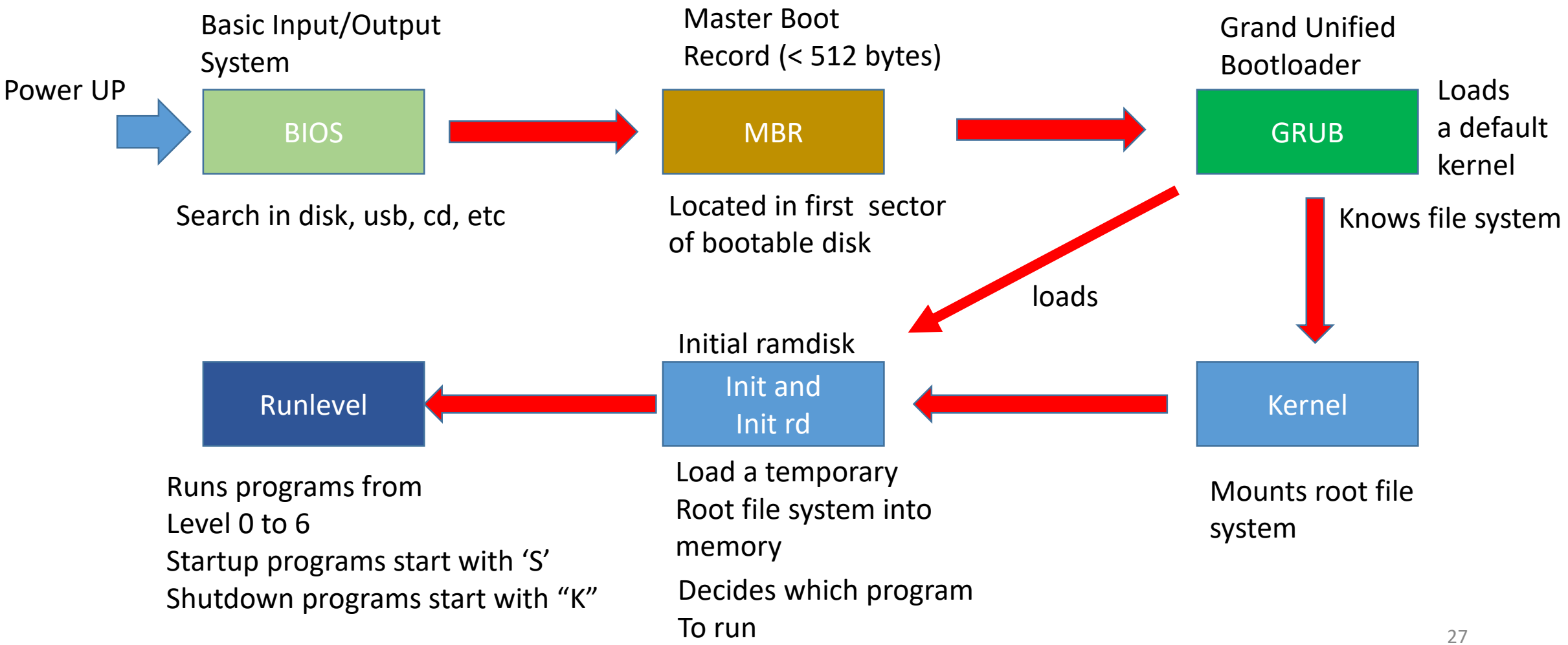
1. import module or package – how does it work [30 min]
   - Moodle: Written-8-27-a [due: 8/29]
2. Requirements: diagrams and description (grammar and spelling)
3. Submission: PDF

Written Assignments for Aug 29

1. pip – what is it and how does it work [30 min]
   - Moodle: Written-8-29-a [due: 8/31]
2. Virtualenv – what is it? [30 min]
   - Moodle: Written-8-29-b [due: 8/31]

What happens when you power up a computer

Linux Boot Process

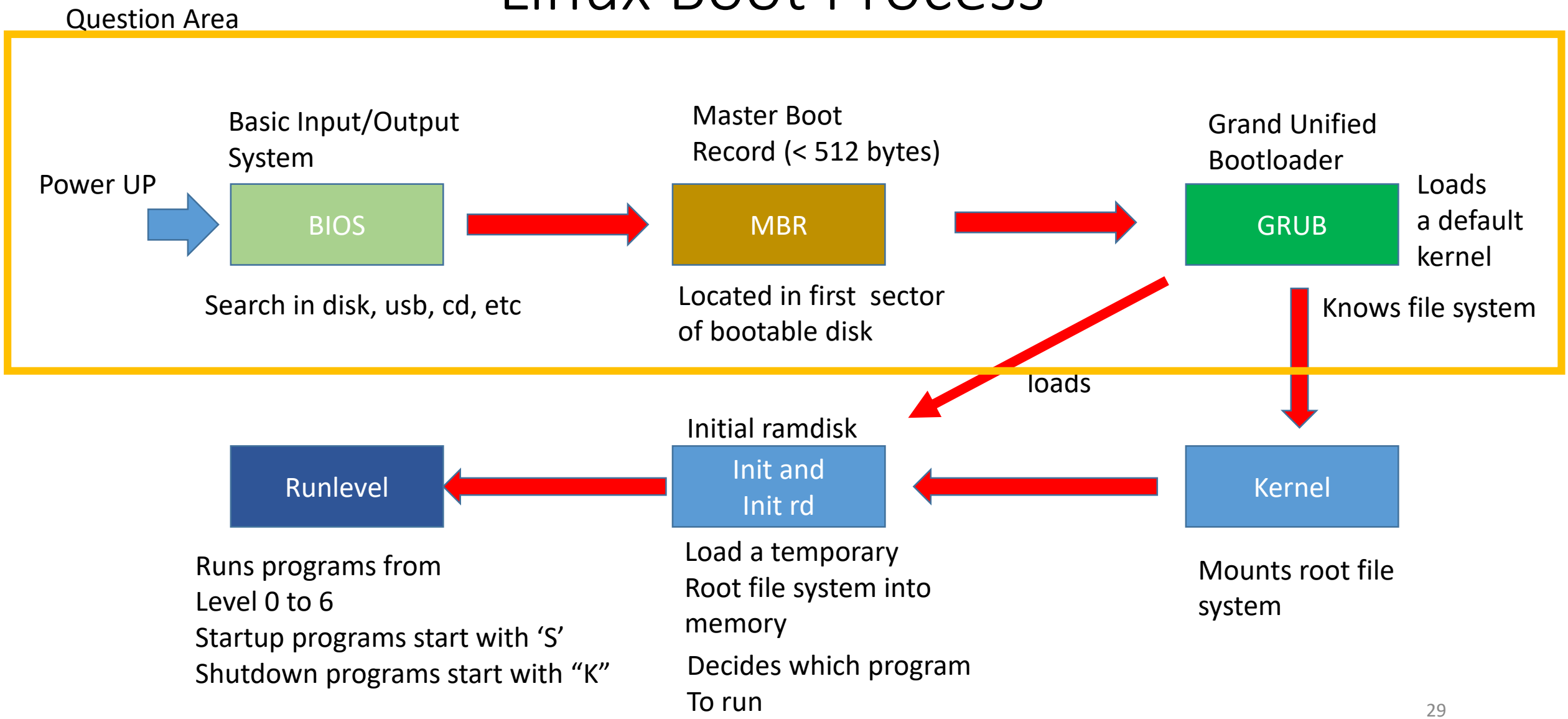


Boot process on Windows or MacOS

- My description of Linux Boot Process does not have enough technical details and description of each step
  - Examples.
    - How does the hardware know where BIOS is located and how does it know how to run it
    - Where does BIOS look for the MBR?
    - What does GRUB actually do?

- Your assignment
  - Write a detail boot process for Linux
  - Diagrams are required
  - Descriptions need to contain excruciating details
  - Moodle: Written-Aug-31-a [due in one week: Sep-7]
  - File format: PDF

Linux Boot Process



```
def fib(n):
    if n <= 2:
        return 1
    else:
        return fib(n-1) + fib(n-2)
```

Test it out.

Try fib(23), fib(45)

Increase the number until it dies. What number is that?

Did it die and why??

Class Exercise –

1. **Rewrite it so it will never run out of memory**

Moodle: Class-Aug-27-a

Written Assignment: Written-Aug-31-a

- Problem:
  - **Is there any way to increase the memory for a python program?**
  - **Use the coding example in class.**