COP 3402 Systems Software

Euripides Montagne
University of Central Florida

Outline

1. **Course Organization**

2. **What is Systems software?**

COP 3402 Systems Software

Who am I and where to find me?

Instructor: Euripides Montagne (HEC-216)

Tele.: (407) 823-2684

E-mail:euripides.montagne@ucf.edu

Lecture meetings: M and W 4:00 p.m. – 5:50 p.m. (CB1 104)

Office hours: M and W 2:00 p.m. - 3:45 p.m. (HEC 216)

COP 3402 Systems Software

Recitations (**there are no recitations the first week of classes**)

GTA : Sanan Hasanov
Email : sanan.hasanov@ucf.edu
Office hours: TBA

Lab. C011 Tuesdays from 4:30 p.m. to 5:20 p.m. in BA2 208

Lab. C012 Tuesdays from 5:30 p.m. to 6:20 p.m. in BA2 208

Lab. C013 Tuesdays from 6:30 p.m. to 7:20 p.m. in BA2 208

Lab. C014 Tuesdays from 7:30 p.m. to 8:20 p.m. in BA2 208

COP 3402 Systems Software
Who are your TAs?

GTA : Sanan Hasanov
Email : sanan.hasanov@ucf.edu
Office hours: TBA

TA: Beatriz Altimeyer Menezes
Email: be611804@ucf.edu
Office Hours: TBA

TA: Sami Najib
Email: sa129714@ucf.edu
Office Hours: Mondays from 6:00 p.m. to 8:00 p.m.
Here is the link: https://ucf.zoom.us/j/93159737966

COP 3402 Systems Software Course outline

Course Outline: This course is designed to provide a fundamental understanding of real and virtual machines as language processor. We will study the processor as an instruction interpreter. Compilers, assemblers, and virtual machines will be presented as systems software for program development. An introduction to Operating system will be given.

Course Topics: introduction to compilers and interpreters, virtual machines, computer architecture and assembler, loaders and linkers, macro-processors, run time environment and operating systems

Prerequisites:
COP3502 – Computer Science I
CDA3103 – Computer Organization

COP 3402 Systems Software Course outline (continued)

Material: Lecture notes: On website.

Required text:
Systems Software: *Essential Concepts,* First Edition, by Eurípides Montagne, Cognella, 2021.

Homework completion and textbook reading a critical determinant of your grade

Recommended
Compilers: Principles, Techniques, & Tools, Second Edition by Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman. Addison Wesley, 2007

Engineering a Compiler*,* Second Edition by Keith Cooper and Linda Torczon Morgan Kaufmann, 2012

Supplementary texts:
Compiler Construction: Principles and Practice by Kenneth C. Louden, PWS, 1997

Concepts of Programming Languages, 8th Edition by Robert W. Sebesta. Addison Wesley, 2010

COP 3402 Systems Software
Semester Plan

- System Software
- Computer System Structure
- Virtual machines
- Compilers and interpreters
- Syntax analysis
- Grammars and parsing
- Code generation
- Assemblers and ELF (Executable Linkable Files)
- Linkers and loaders
- Operating System Structure
- Interrupt Handling
- Processes and threads implementation
- Concurrency and process synchronization

COP 3402 Systems Software
Some Things to Remember

- We are all growed up
  - “85% of life is just showing up” – Woody Allen
  - Do the homework, read the textbook, go to recitation section
  - Learn to manage your time (it’s all about planning)
  - “A failure to plan on your part does not constitute an emergency on my part”
  - “Everyone *wants* to win. Winners *plan* to win.” – Coach K
  - “He who questions training only trains himself in asking questions” – The Sphinx (Mystery Man)

COP 3402 Systems Software
Please remember this statements

1) Webcourses is only used to post Lectures, assignments and exams grades. "The percentages on Webcourses are meaningless for this class".

2) To calculate your grade, You must use the syllabus grading policy.

3) Attendance is not mandatory but it is highly recommended to attend class because some times a student ask a question and that leads to exploring some concepts and enrich knowledge. Those discussions are not in the lectures.

COP 3402 Systems Software

An Introduction to Systems Software

What is Systems Software?

What is Systems Software?

Systems Software consists of a set of programs that support the operation of a computer system.

Systems software programs help, the programmer, to simplify the programming process.

Furthermore, systems software creates an environment to run application software efficiently.

Examples of systems software are:

- **Text editors**
- **Compilers**
- **Loaders**
- **Linker**
- **Debugger**
- **Assembler**
- **Operating system**

Systems software can be classified in two groups:

1.- Software to create a program development environment

- Text editor
- Compiler
- Assembler
- Linker
- Debugger (low-level)

2.- Software to create a run-time environment

- Operating system
- Loader
- Dynamic Linker
- Program libraries

stems Software: Program Development Environment

xt editor: Software that permits the creation and editing of text files (i.e. application programs).

mpiler: Translates programs written in a high level language to machine code(or assembly Language).

sembler: Translates programs written in assembly language to object code(binary).

atic Linker: Combines and resolves references between object programs and creates the executable code.

bugger It is used to debug executable programs and their **w-level)** related object code and source program.

Systems Software: Run-Time Environment

Loader: Loads an executable code and starts its execution.

Libraries:

Precompiled programs that create a set of functions to be used by other programs.

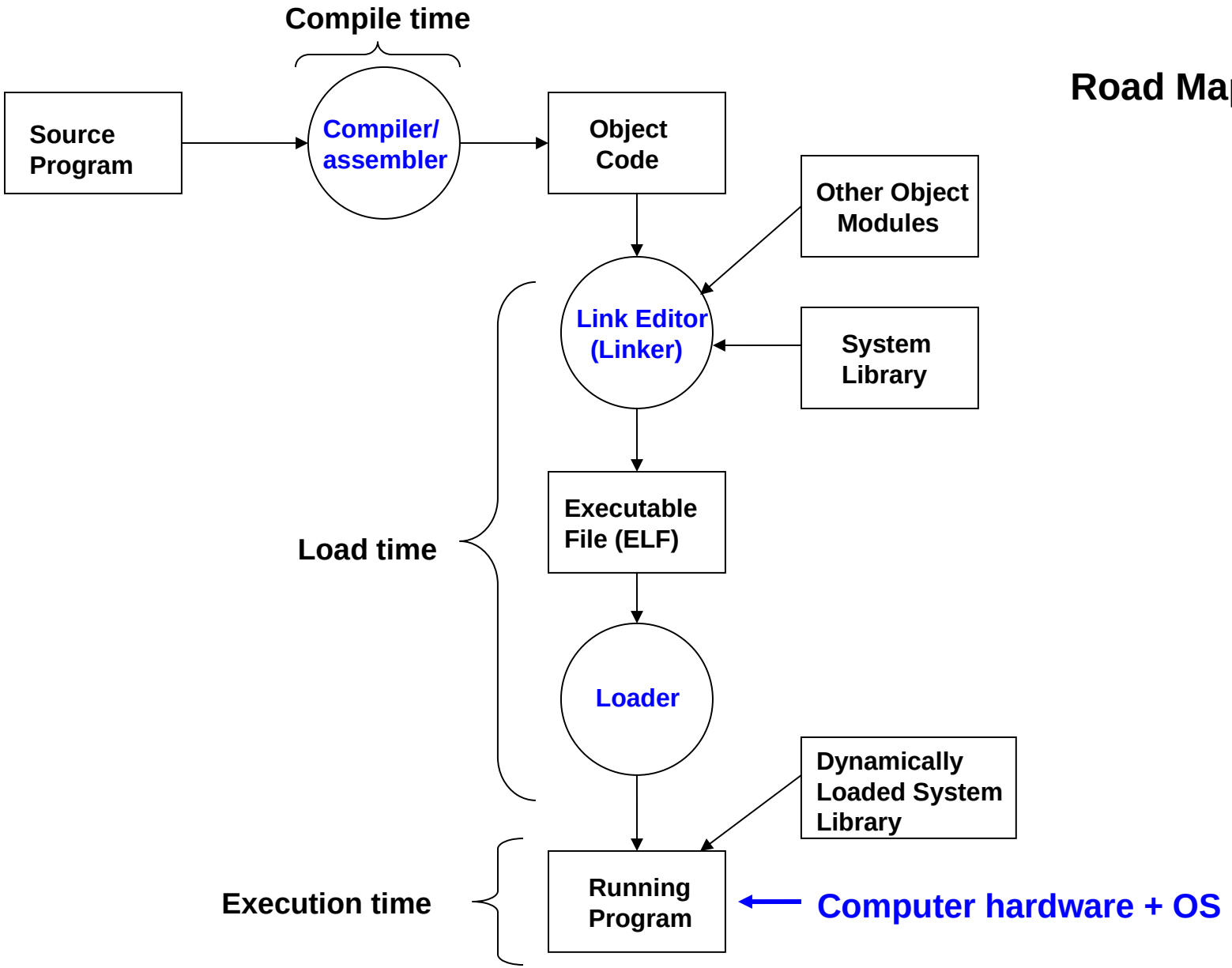
Dynamic Linker:

Loads and links shared libraries at run-time

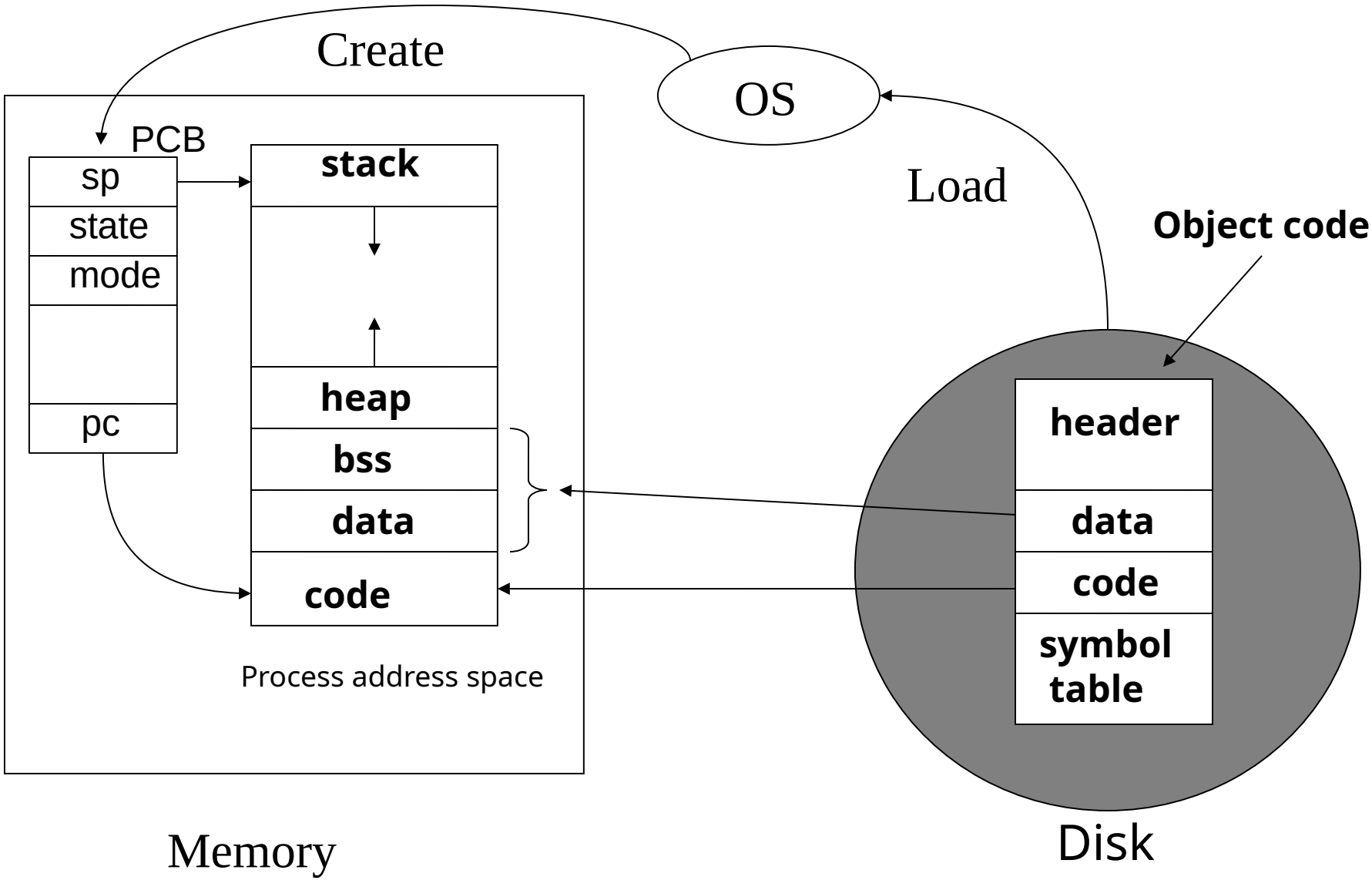
Operating system:

An event driven program that makes an abstraction of the computer system. The operating system handles all resources efficiently, creates an environment for application programs to run, and provides a friendly interface between the user and the computer system.

Road Map


Compile time
Source Program
Compiler/ assembler
Object Code
Other Object Modules
Link Editor (Linker)
System Library
Executable File (ELF)
Load time
Loader
Dynamically Loaded System Library
Running Program
Execution time
Computer hardware + OS

Environment for programs to run



COP 3402 Systems Software

Next class we will talk about

The processor
as an instruction interpreter