

COP

3502





Algorithm 1

1. Stand up and think of a number
2. Pair off with someone standing, add your numbers together, and adopt the sum as your new number
3. One of you should sit down; the other should go back to step 2, unless you are the sole person standing



Algorithm 2



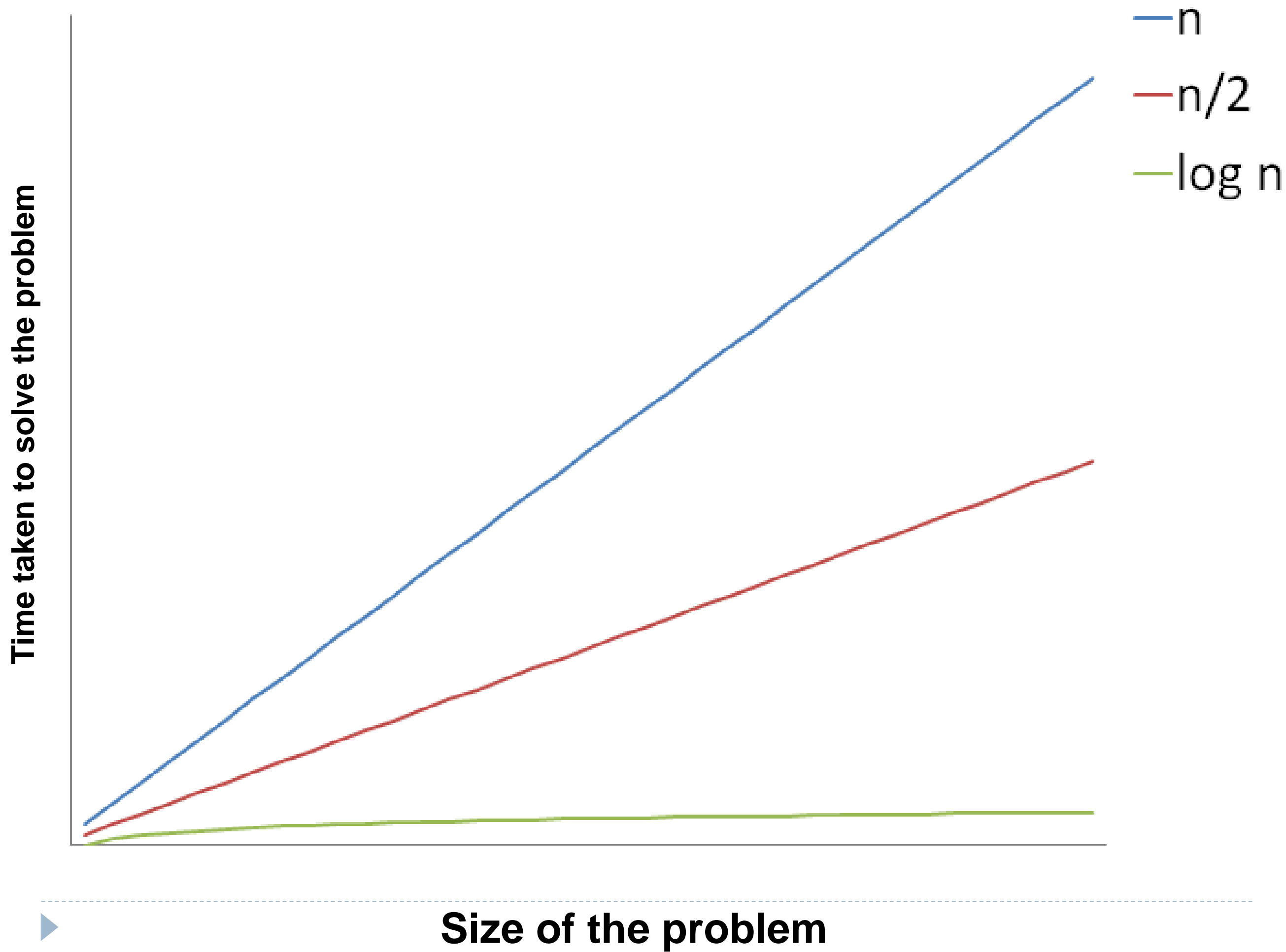
GAME

Alice: Guesses a random number < 20 .

Bob: Can only ask Binary questions.

What is the minimum number of questions that Bob needs to ask to get to Alice's number.





APIs



APIs

SYNOPSIS

```
#include <math.h>  
double sqrt(double X);
```

Do one thing, and do it well
As small as possible, but not smaller
Intuitive and easy

DESCRIPTION: sqrt computes the positive square root of the argument. ...

RETURNS: On success, the square root is returned. If X is real and positive...

```
Python  
import math  
math.sqrt(x)
```



Hello, world!

```
/* In C */  
#include <stdio.h>  
int main (void)  
{  
    printf("Hello, world!");  
}
```

In Python: `print("hello world")`



Basic knowledge about computers

- ▶ To utilize computers you need both hardware and software

Hardware



Hardware

Desktop, laptop.

Operating system



Operating systems

Windows XP, Vista,
Unix, Linux,
MaxOS

Software Applications

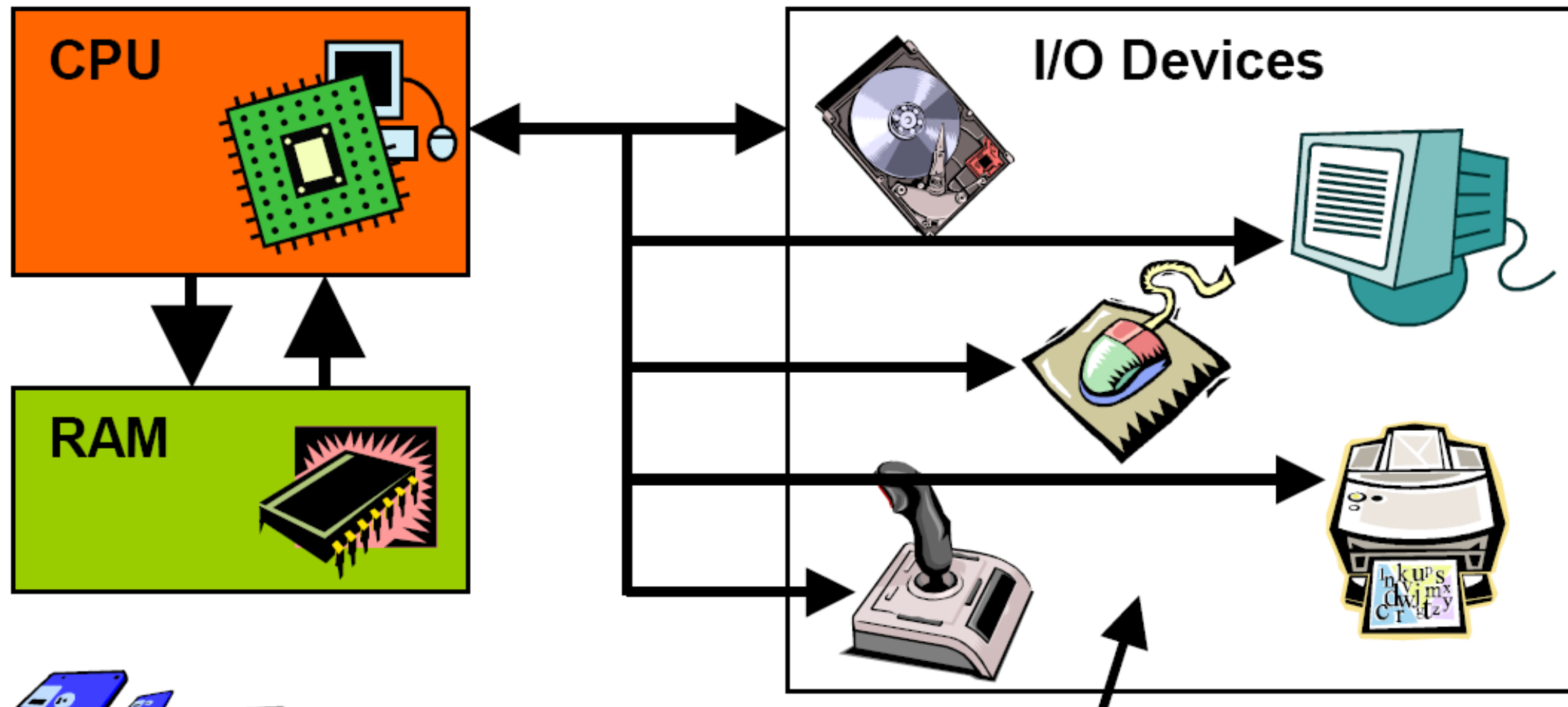


Software Applications

Office, Photoshop



Hardware

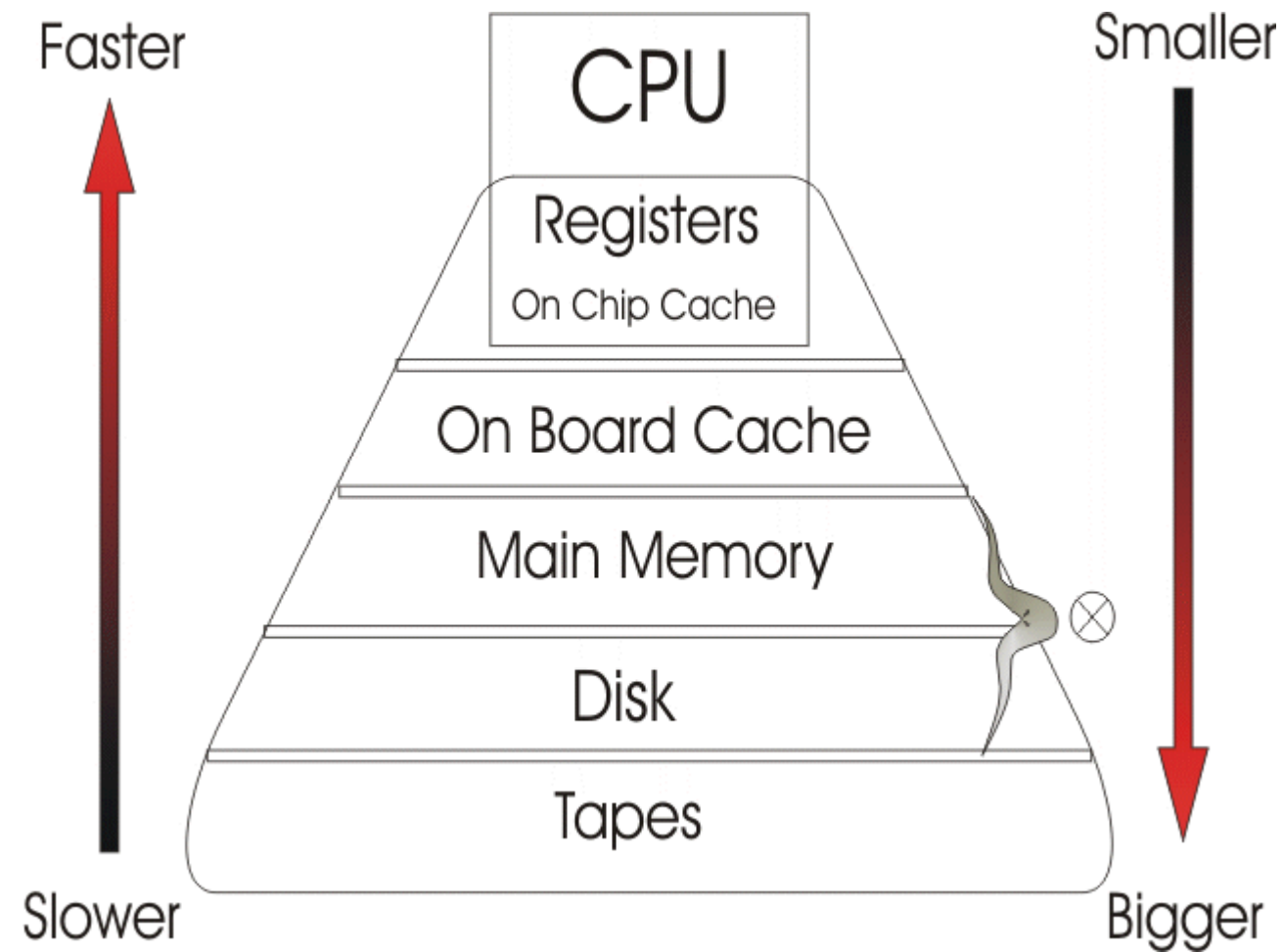


Central processing unit (CPU) → calculate, compute, etc.

Random access memory (RAM) → store temporary data

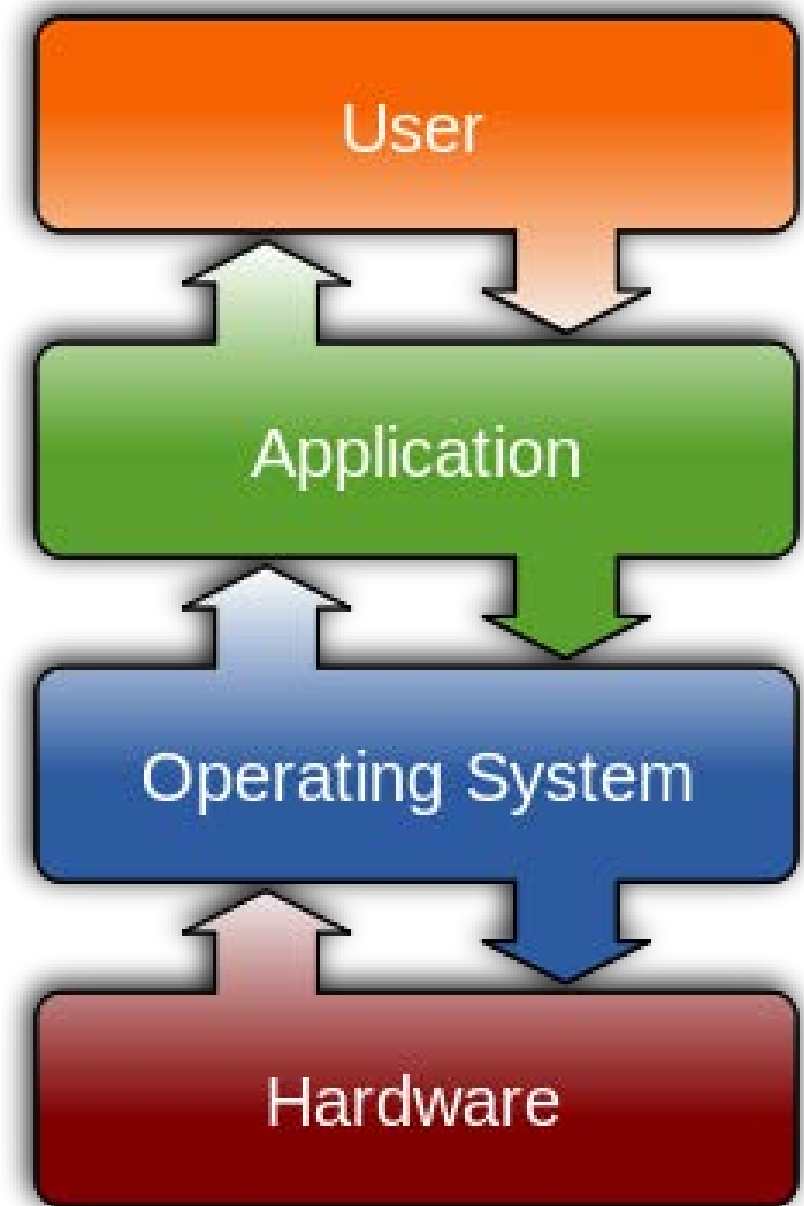
I/O device → input/output device

Hardware



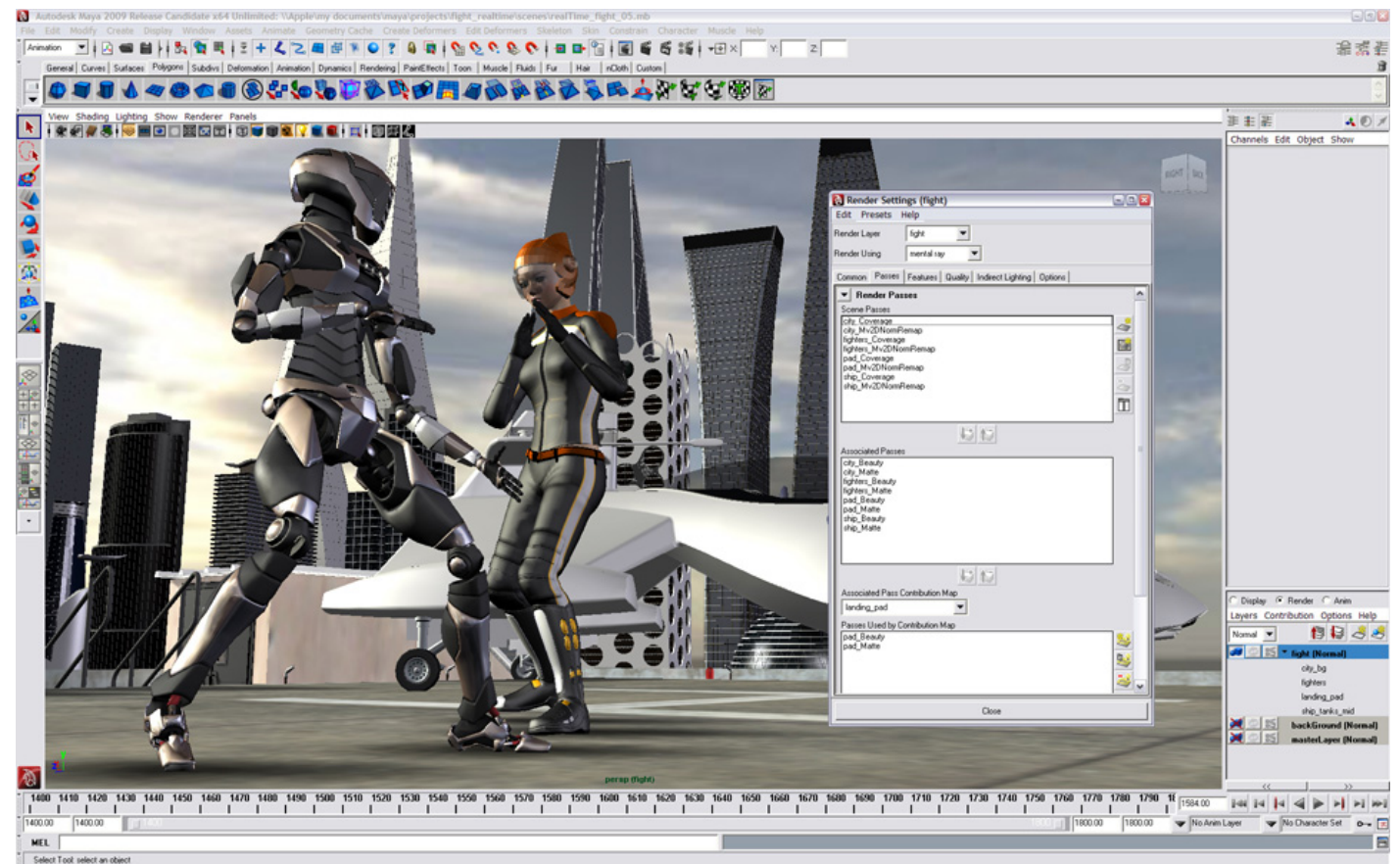
Operating System

- ▶ Examples of popular modern operating systems include [Android](#), [BSD](#), [iOS](#), [Linux](#), [Mac OS X](#), [Microsoft Windows](#),^[3] [Windows Phone](#), and [IBM z/OS](#).
- ▶ An **operating system (OS)** is a collection of software that manages [computer hardware](#) resources and provides common [services](#) for [computer programs](#).



Application Software

- ▶ **Application software**, also known as an **application** or an **app**, is computer software designed to help the user to perform specific tasks.
- ▶ Examples include enterprise software, accounting software, office suites, graphics software and media players.



Expectations

- ▶ attend all lectures
- ▶ class participation
mid term and final
- ▶ Quizzes and weekly assignments
- ▶ Practice programming/tools on your own
(4hrs+/week)



Expectations

- ▶ Own a laptop/desktop with root/administrator access
- ▶ Windows / Mac / Linux
- ▶ Preferably : Ubuntu with root.
- ▶ Get a CS Garnet account.
- ▶ And make sure you have ssh client installed on your computer.



Grades

Pass/Fail or letter grade

weekly assignments + quizzes (weight: 90)

attendance + class participation (weight: 10)

min

midterm (weight: 50)

final (weight: 50)



PASS



FALL



Office Hours

Mon: 4.15pm – 5.15pm, Love 161

Weekly Lectures

Mon: 5.15pm – 6.35pm, Love 103

Course Plan

- ▶ Subject to Change a bit as we progress
- ▶ The following slides will be updated to reflect this as the semester unfolds



Week 0

- ▶ Introduction.
- ▶ Bits.
- ▶ Binary.
- ▶ ASCII. Unicode
- ▶ Programming.
- ▶ Algorithms.
- ▶ Statements.
- ▶ Boolean expressions.



Week 1

- ▶ C. Source code.
- ▶ Compilers. Object code
- ▶ SSH. SFTP . GCC.
- ▶ Functions. Comments. Standard I/O.
- ▶ Arithmetic operators. Precedence. Associativity.
- ▶ Local variables. Types. Casting.
- ▶ Libraries.
- ▶ Boolean expressions continued.
- ▶ Conditions continued.
- ▶ Loops continued.



Week 2

- ▶ Functions, continued.
- ▶ Global variables.
- ▶ Parameters.
- ▶ Return values.
- ▶ Scopes. Arrays
- ▶ Strings.
- ▶ Command line arguments.
- ▶ Cryptography.



Week 3

- ▶ Linear search. Binary Search.
- ▶ Asymptotic notation.
- ▶ Recursion.
- ▶ Pseudo randomness.
- ▶ Bubble sort. Selection sort.
- ▶ Insertion sort. Merge sort.
- ▶ Debugging.



Week 4

- ▶ Structures.
- ▶ Debugging, continued.



Week 5

- ▶ File I/O.
- ▶ Forensics.
- ▶ Linked lists.
- ▶ Stacks.
- ▶ Queues.



Week 6

- ▶ Valgrind.
- ▶ Bitwise operators.
- ▶ Hash tables.
- ▶ Trees.
- ▶ Binary search trees.
- ▶ Tries.
- ▶ Huffman coding.



Week 7

- ▶ HTTP.
- ▶ HTML.
- ▶ SQL.



Week 8

- ▶ CSS.
- ▶ Inheritance.
- ▶ JavaScript.
- ▶ Events, continued.
- ▶ Ajax.



Week 9

► More Python



to be continued...

