

CSCI 141

Lecture 24 Reading and Writing Files

Announcements

- A5 Code and A5 Written are due today!
- **Reminder: No late submissions accepted after Tuesday 6/4 at 10pm**
- There will be a lab next week. It will be short.
- Sample final exam questions: same deal as the midterm:
 - Canvas announcement will go up this weekend
 - Sample Questions due by Wednesday's lecture.
 - Worth 2 points of extra credit on the final exam.

Announcements

- Monday's quiz is worth 12 points - double the usual. Please show up!
- Wednesday and Friday will be review, and attendance is not required.

Goals

- Know the basics of file input/output:
 - Reading and seeking - iterating over lines, `read`, `readlines`, `seek`
 - Writing - `write` method

Last time

- Know how to modify lists using the following:
`insert`, `remove`, `del`
- Know the basics of how to use dictionaries (dicts):
 - Creation, assignment, indexing
 - `in`, `del`, iterating over keys and values

Last time

- Know how to modify lists using the following:
`insert`, `remove`, `del`
- Know the basics of how to use dictionaries (dicts):
 - Creation, assignment, indexing
 - `in`, `del`, **iterating over keys and values**

Dictionaries: Iterating

```
d = {key1: value1, key2: value2, ...}
```

```
for key in d:  
    print(key)
```

```
for key in d.keys():  
    print(key)
```

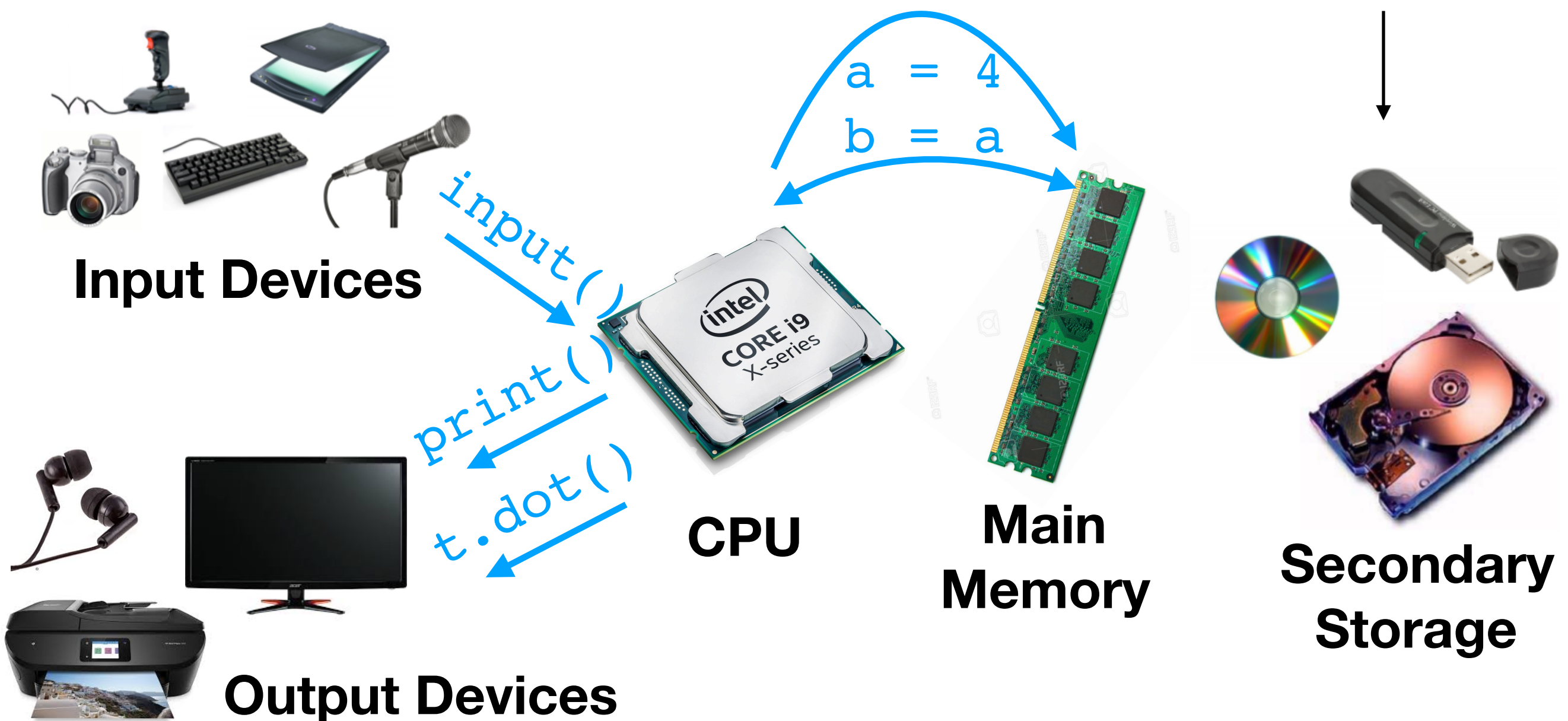
```
for val in d.values():  
    print(val)
```

```
for (key, val) in d.items():  
    print(key, val, sep=": ")
```

Note: Like range, these methods return sequences that are not lists.
To get a list of values use `list(d.values())`

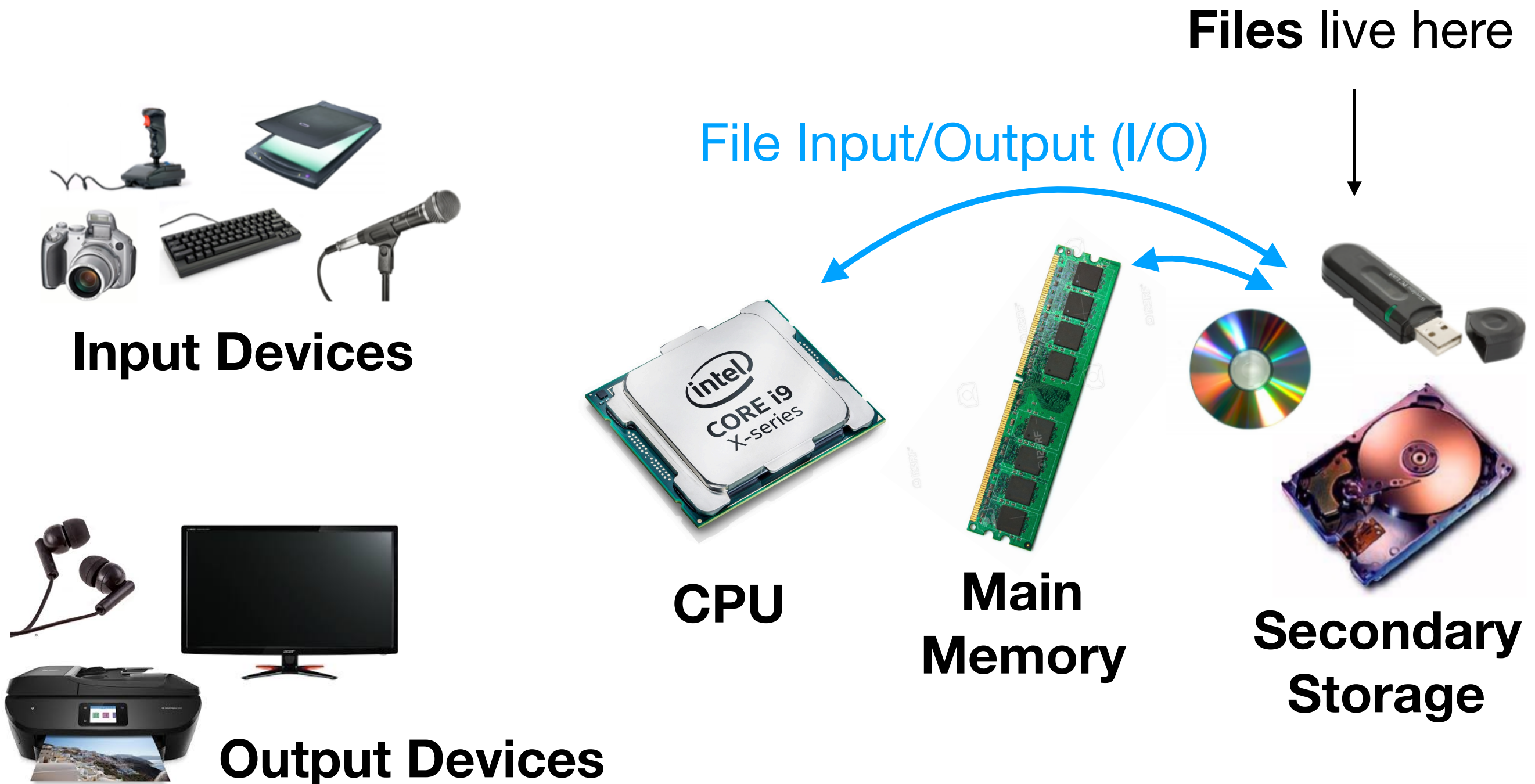
A blast from the past:

A simple model of a computer:



A blast from the past:

A simple model of a computer:



Files - Opening, Reading

In A5, I provided the code to read the training and test data from a file (lightly edited):

```
input_file = open(filename, "r")
for line in input_file:
    if "#" not in line:
        line = line.strip("\n")
        line_list = line.split(",")
```

File object

path to the file (str)

open file for reading (vs. writing)

you can **iterate** over a File object (!)

what does split do?

familiar string stuff

String's `split` method

Splits the string into a list on a given separator, or all whitespace by default.

(demo)

String's `split` method

Splits the string into a list on a given separator, or all whitespace by default.

(demo)

```
split()  
split(" ")  
split(",")
```

```
list.join (if time)
```

File objects



write

read

seek

Reading files: demo

Writing files

```
output_file = open(filename, "w")  
output_file.write("a string\n")
```

Write doesn't behave like print: it writes exactly the string you give it, with no implicit newlines or spacing

Reading files: why is this cool?

- You can now play with some big data:
 - A5, for example.
 - Another example in next week's lab:
Make this map showing earthquakes and their magnitudes:

