

# Comp 112

## Files and Exceptions

# What is a file?

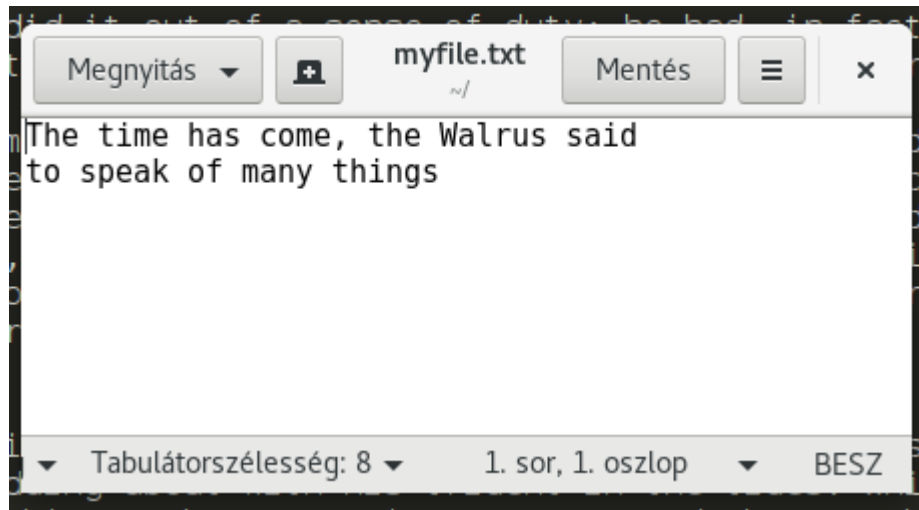
- A file has a name
- A file has a location
- A file has content
- A file is persistent
- A file is accessed by a program, often on behalf of a user

# Let's make a file

```
f = open("myfile.txt", "w")
f.write("The time has come, ")
f.write("the Walrus said\n")
f.write("to speak of many things\n")
f.close()
```

# Let's make a file

The file is now visible in your operating system.  
You can open it with your choice of program.



# Let's read a file

```
>>> f = open("myfile.txt", "r")
>>> f
>>> <_io.TextIOWrapper name='myfile.txt' mode='r' encoding='UTF-8'>
>>> line1 = f.readline()
>>> line1
'The time has come, the Walrus said\n'
>>> line2 = f.readline()
>>> line2
'to speak of many things\n'
>>> line3 = f.readline()
>>> line3
''
>>> f.close()
```

# Let's read a file

```
>>> f = open("myfile.txt", "r")
```

```
>>> everything = f.read()
```

```
>>> everything
```

```
'The time has come, the Walrus said\n\nto  
speak of many things\n'
```

```
>>> f.close()
```

# Let's read a file

```
>>> f = open("myfile.txt", "r")
>>> line1 = f.readline()
>>> line1
'The time has come, the Walrus said\n'
>>> line2 = f.readline()
>>> line2
'to speak of many things\n'
>>> line3 = f.readline()
>>> line3
''
>>> f.close()
```


# Let's read a file

```
f = open("myfile.txt", "r")
for line in f:
    print(line)
f.close()
```



# Idiom: read, modify, write

345  
56756  
23  
98  
23432  
677  
24324234234  
2323  
0  
-23434  
89823423  
34  
1001  
8008



Create a file named numbers.txt  
with this content

# Idiom: read, modify, write

```
fr = open("numbers.txt", "r")
numbers = []
for line in fr:
    numbers.append(int(line))
fr.close()
numbers.sort()
fw = open("numbers.txt", "w")
for number in numbers:
    fw.write(str(number)+"\n")
fw.close()
```

# Idiom: read, modify, write

-23434

0

23

34

98

345

677

1001

2323

8008

23432

56756

89823423

24324234234



Resulting file is sorted.txt

```
def add_score(score):
    scores = []
    try:
        with open("spaceinvader-scores.txt", "r") as f:
            for line in f:
                sline = line.split()
                sscore = int(sline[0])
                sname = " ".join(sline[1:])
                scores.append((sscore, sname))
    except IOError:
        scores = [(0, "")]*10
    i = 0
    while i < len(scores):
        sscore, sname = scores[i]
        if score >= sscore:
            newname = turtle.textinput("High score", "Congratulations. Enter your name.")
            if newname:
                scores.insert(i, (score, newname))
                scores = scores[0:10]
            try:
                with open("spaceinvader-scores.txt", "w") as f:
                    for (sscore, sname) in scores:
                        f.write(str(sscore)+" "+sname+"\n")
            except IOError:
                pass
            break
        i+=1
```

# Dealing with errors

```
>>> f = open("thisfiledoesnotexist.txt", "r")
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
FileNotFoundError: [Errno 2] No such file or  
directory: 'thisfiledoesnotexist.txt'
```

# Dealing with errors

```
try:
    f = open("somefile.txt", "r")
    print("I can read somefile.txt: "+f.read())
    f.close()
except FileNotFoundError:
    print("I can't find somefile.txt")
print("In either case, the program keeps going")
```

# Dealing with errors

```
>>> int("not an int")
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
ValueError: invalid literal for int() with base  
10: 'not an int'
```

# Dealing with errors

```
while True:
    try:
        num = int(input("Please type a number!"))
    except ValueError:
        print("Sorry, that's not a number")
        continue
    break
print("You entered a number! It is", num)
```



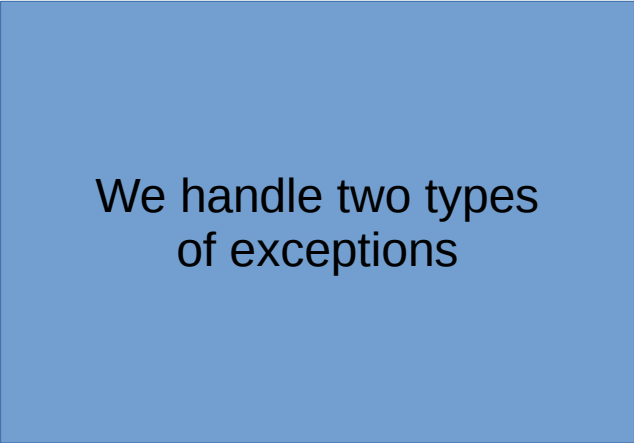
# Dealing with errors

```
fr = open("numbers.txt", "r")
numbers = []
for line in fr:
    numbers.append(int(line))
fr.close()
numbers.sort()
fw = open("sorted.txt", "w")
for number in numbers:
    fw.write(str(number)+"\n")
fw.close()
```

This code can produce (at least) two kinds of exceptions: `FileNotFoundError` and `ValueError`

# Dealing with errors

```
try:
    fr = open("numbers.txt", "r")
    numbers = []
    for line in fr:
        numbers.append(int(line))
    fr.close()
    # code omitted
except ValueError:
    print("The file contained a bad number!")
except FileNotFoundError:
    print("Couldn't access the file!")
```



We handle two types  
of exceptions

Exception name	Explanation
KeyboardInterrupt	The user pressed Ctrl-C, which terminates your program
ValueError	Raised, for example, when accessing a key that doesn't exist in a dict
IndexError	Raised when accessing a position in a list that doesn't exist
NameError	You tried to access a variable that doesn't exist. Usually a misspelling
TypeError	You tried to do an operation to a value of the wrong type, such as subtracting strings
ZeroDivisionError	You tried to divide by zero
FileNotFoundError	You tried to read a file, but it doesn't exist
IsADirectoryError	You tried to read a file, but it's a directory, not a file
NotADirectoryError	You tried to do something to a file that you can do only to a directory

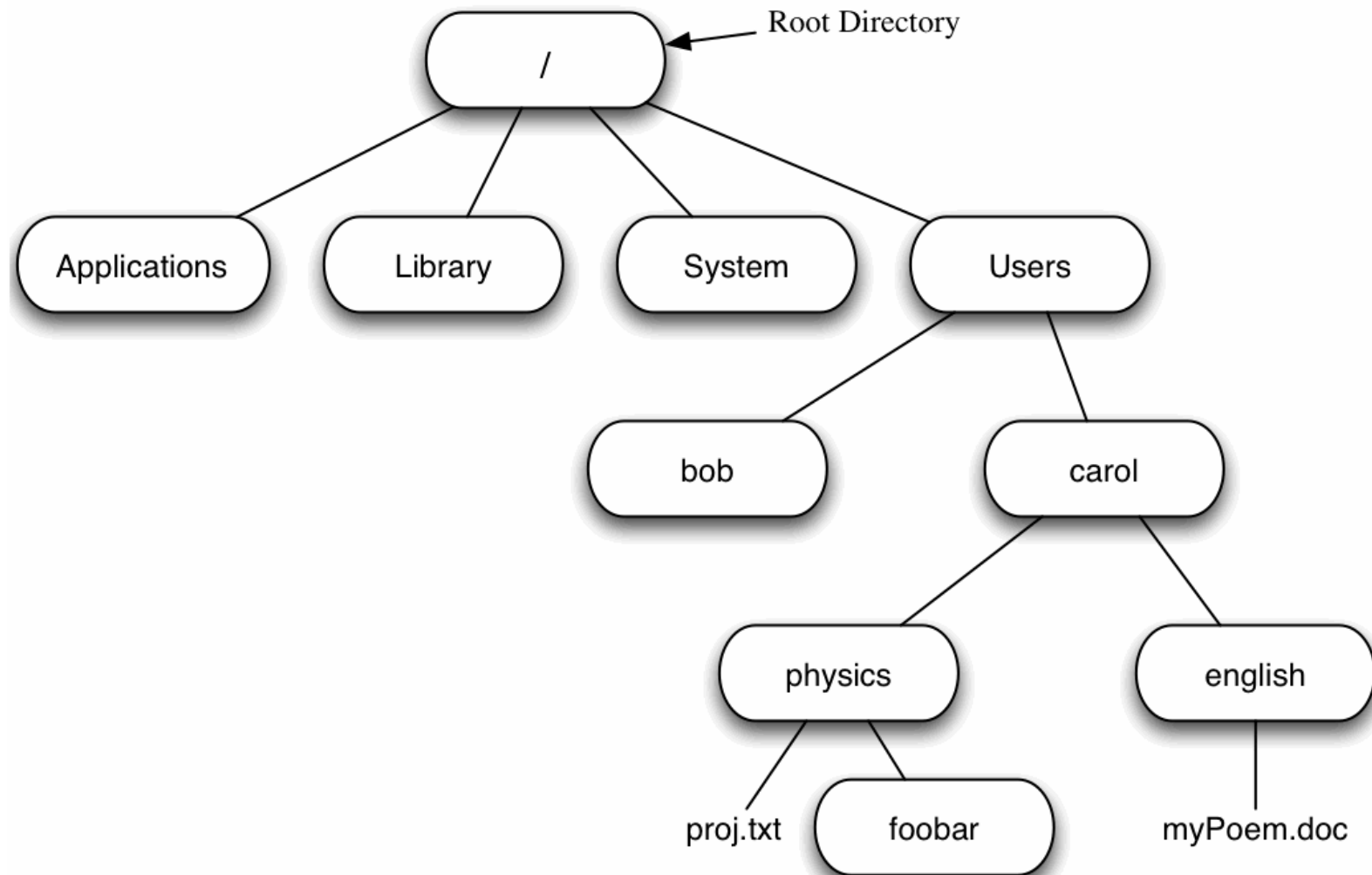
# Raising exceptions

```
def find_first_vowel(s):  
    i=0  
    while i < len(s):  
        if s[i].lower() in 'aeiou':  
            return i  
        i+=1  
    raise Exception("No vowels found!")
```

# Exception percolation

```
def divide_five_by(divisor):  
    return 5 / divisor  
  
try:  
    print(divide_five_by(3)) # OK  
    print(divide_five_by(0)) # Not OK  
    print("We never get here")  
except ZeroDivisionError:  
    print("Something went wrong")
```

# Directory hierarchy



/Users/carol/english/myPoem.doc

# Relative paths

Where is myfile.txt?

- `open("myfile.txt")`
  - It's in the "current directory," usually the same directory as the program
  - Path is relative
- `open("Documents/myfile.txt")`
  - It's in the directory Documents, which is in the same directory as the program
  - Path is relative
- `open("/Users/jEEPstein/Documents/myfile.txt")`
  - It's in the Documents folder, in the home directory of jeepstein
  - Path is absolute
  - Notice that path begins with slash

# Directory hierarchy

```
>>> import os
```

```
>>> os.listdir(".")
```

```
['Music', 'Documents', 'School',  
'Downloads', 'test.py', 'pong.py']
```

```
>>> os.listdir("Music")
```

```
['Billie Jean.mp3', 'Take On Me.mp3',  
'Testure.mp3', 'Rosa Decidua.mp3']
```



# Directory hierarchy

```
>>> import os
>>> os.listdir("School")
['comp112', 'comp360']
>>> os.listdir("School/comp112")
['hw1.txt', 'hw2.txt', 'hw3.txt', 'lectures']
>>> f = open("School/comp112/hw1.txt", "r")
>>> f.readline()
'1. Write a program to calculate baz bar\n'
```

# Directory hierarchy

```
import os

def dir_contents(path):
    contents = os.listdir(path)
    files = []
    for item in contents:
        item_path = path + "/" + item
        if os.path.isdir(item_path):
            sub_dir_contents = dir_contents(item_path)
            files.extend(sub_dir_contents)
        else:
            files.append(item_path)
    return files
```

# Directory hierarchy

```
>>> dir_contents(".")  
['Music', 'Documents', 'School',  
'Downloads', 'test.py', 'pong.py',  
'Music/Billie Jean.mp3',  
'Music/Testure.mp3', 'Music/Rosa  
Decidua.mp3', 'School/comp112/hw1.txt',  
School/comp112/hw2.txt',  
'School/comp112/hw3.txt',  
'School/comp360/funstuff.txt']
```

# Appending

```
>>> f = open("file.txt", "w")
>>> f.write("Line 1\n")
>>> f.close()
>>> f2 = open("file.txt", "a")
>>> f2.write("Line 2\n")
>>> f2.close()
```

# with syntax

```
f = open("file.txt", "w")
```

```
f.write("Something")
```

```
f.close()
```

-----

```
with open("file.txt", "w") as f:
```

```
    f.write("Something")
```

# A simple programming language

```
vals = {}
with open("file.txt", "r") as f:
    for line in f:
        words = line.split()
        if words[0] == "SET":
            var_name = words[1]
            var_val = int(words[2])
            vals[var_name] = var_val
        elif words[0] == "ADD":
            var_name1 = words[1]
            var_name2 = words[2]
            var_val1 = vals[var_name1]
            var_val2 = vals[var_name2]
            result = words[3]
            vals[result] = var_val1 + var_val2
        elif words[0] == "PRINT":
            var_name = words[1]
            print(vals[var_name])
```

# A simple programming language

```
SET a 4
```

```
SET b 23
```

```
SET c 45
```

```
ADD a b acc
```

```
ADD acc c acc
```

```
PRINT acc
```