CS21: INTRODUCTION TO COMPUTER SCIENCE

Prof. Mathieson
Fall 2017
Swarthmore College
Outline Oct 13:

- Return Lab 4
- Practice stack diagrams (pick up a handout)
- List accumulators and mutation in graphics
- Go over `max_circle.py` program
- Try/except in python: `try_except.py`

Notes

- Lab 5 due tomorrow night
- Office Hours 3-5pm today!!
- I’ll be around over break if you want to meet
Screenshots and Videos: please email!

- Windows Videos: [https://www.hongkiat.com/blog/window-screen-recording-softwares/](https://www.hongkiat.com/blog/window-screen-recording-softwares/)
- Mac Videos: Quicktime
- Linux screenshot:

![Linux screenshot](Image by Rich Wicentowski)
Stack Diagram Practice
(see Handout 2)
```python
def add(lst, x):
    n = len(lst)
    for i in range(n):
        lst[i] = lst[i] + x
    print("done adding")

def main():
    data = [7, 10, 4]
    z = 2
    add(data, z)
    print(data)

main()
```
Programs for today

- `inclass/week06/stack_handout.py`
- `inclass/week06/max_circle.py`
- `inclass/week06/try_except.py`
- `practice/snow_challenge.py`
Try/Except (start)
Try/Except

• Whenever an error message is produced, our code terminates and we can’t execute any further code
• Try/except is a way to navigate/circumvent errors
• Example: if a user should enter an integer but they don’t, allow them to try again
• A beginning of this example is shown below (add a while loop after break):

```python
try:
    n = int(input("Enter an integer: "))
except:
    print("not a valid number!")
print("continuing code...")
```