As you walk in

Quiz will start at the beginning of lecture

- Have pencil/pen ready
- Don’t use your own scratch paper
  - We have some if you need it
- Silence phones

Quiz

Before we start

- Don’t open until we start
- Make sure your name and Andrew ID are on the front
- Read instruction page
- No questions (unless clarification on English)
- Typo in one of the versions an “i” should be “in”

Additional info

- 20 min

15-112 Lecture 2

Week 2 Tue Loops

Instructor: Pat Virtue

Announcements

Quiz

Grades

- Likely ready Wednesday
  - Superhero TAs!
- Very small impact on final grade

Fix-it Fridays!

- Combined with Friday pre-reading sections
- More information coming on Piazza

From Syllabus

Quizzes (about 8)	10%	Lowest 2 quiz grades are **half-weighted**.

Announcements

Weekly Rhythm Assignments/Quizzes

- Today, HW2 released; Week 3 Pre-reading published
- Wed, CP3 released
- Fri, 8 pm: CP3
- Sat, 8 pm: HW2
- Next Tue, in-lec: Quiz2

Loops

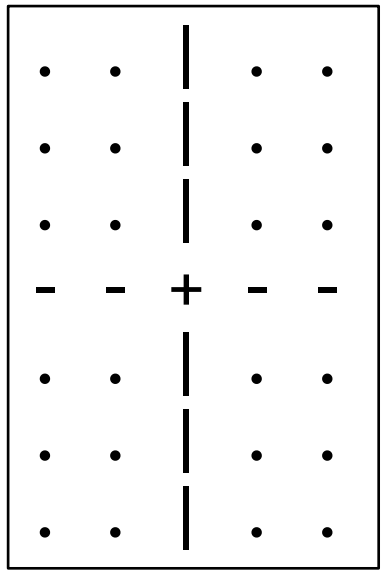
Poll 1

What does this code print?

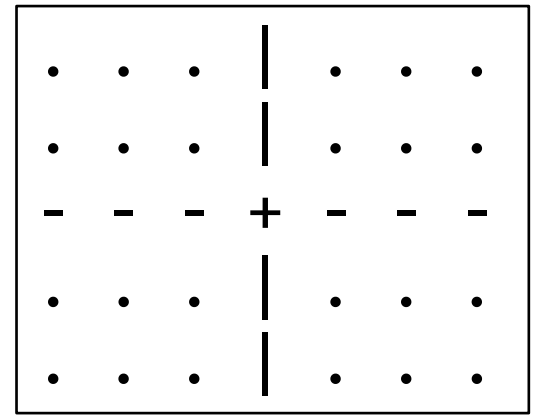
```
def printPlot(xMin, xMax, yMin, yMax):
    for xGrid in range(xMin, xMax+1):
        for yGrid in range(yMin, yMax+1):
            if xGrid == 0 and yGrid == 0:
                pixel = '+'
            elif xGrid == 0:
                pixel = '|'
            elif yGrid == 0:
                pixel = '-'
            else:
                pixel = '.'
            print(pixel, end=" ")
        print()
```

```
printPlot(-3, 3, -2, 2)
```

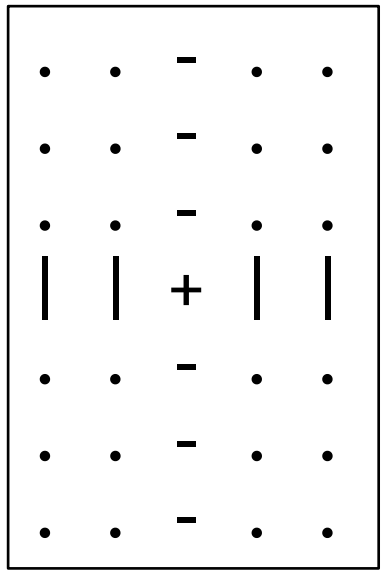
A)



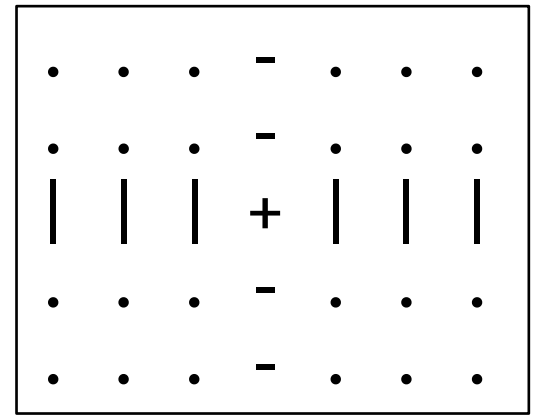
B)



C)



D)



E) I have no idea

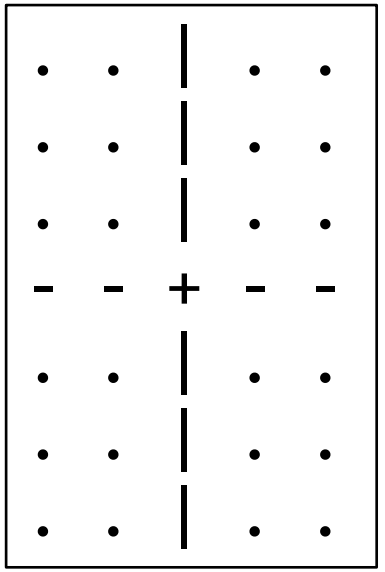
Poll 1

What does this code print?

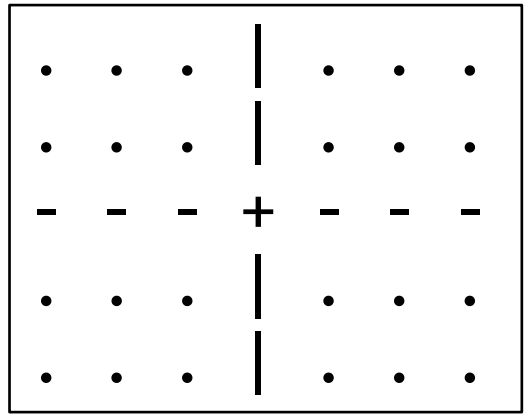
```
def printPlot(xMin, xMax, yMin, yMax):
    for xGrid in range(xMin, xMax+1):
        for yGrid in range(yMin, yMax+1):
            if xGrid == 0 and yGrid == 0:
                pixel = '+'
            elif xGrid == 0:
                pixel = '|'
            elif yGrid == 0:
                pixel = '-'
            else:
                pixel = '.'
            print(pixel, end=" ")
        print()
```

```
printPlot(-3, 3, -2, 2)
```

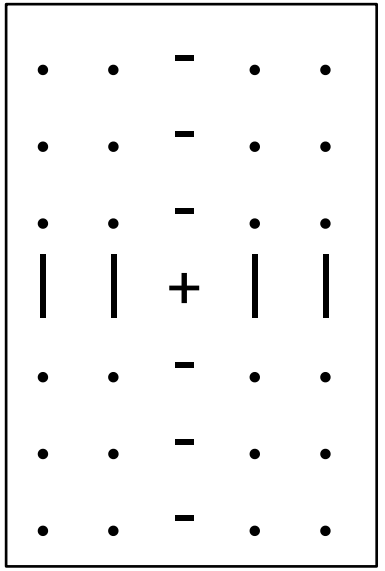
A)



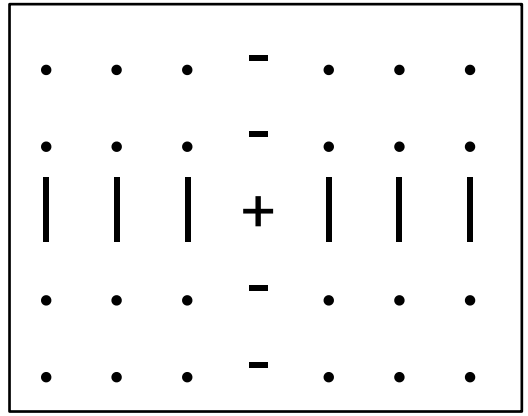
B)



C)



D)



E) I have no idea

Poll 2

Which code is better

A)

```
def sumFromMToN(m, n):
    total = 0
    for x in range(m, n+1):
        total += x
    return total
```

B)

```
def sumFromMToN(m, n):
    total = 0
    x = m
    while x <= n:
        total += x
        x += 1
    return total
```

For Loops vs While Loops

Often, we can write our code using either

How do we choose

- For loops are often easier to reason about, especially if were looping over a known sequence
- While loops work well when we don't know how many loops we need to do
- Easier to make mistakes with while loops
  - "Help! I run my code, but it doesn't do anything!!"
  - Infinite loop!!

    Tip: Use ctrl-C to interrupt program execution

    Tip: Include some print statements to see the loop in action

While Loops

Pick a number between 0 and 1000 (Unknown number of loops)

```
print("Enter first guess: ", end="")
guess = int(input())
numAttempts = 1

while guess != secret:
    if guess > secret:
        print("--- Too high!")
    else:
        print("--- Too low!")
    print("Enter new guess: ", end="")
    guess = int(input())
    numAttempts += 1

print(f"You got it in {numAttempts}! The secret number was {secret}!")
```

Poll 3

What is the n-th prime number when n=3?

A. 2

B. 3

C. 4

D. 5

E. 6

F. 7

G. 8

H. 9

I. 10

J. 11

Pattern: Find the n-th thing

Find the n-th dino

Pattern: Find the n-th thing

Need

- A way to get to the next guess
- A way to check it: isThing(guess)

Sketch

Loop from guess to guess until you've found n (well actually n+1) things

```
        if isThing(guess):
                numFound += 1
```

Pattern: Find the n-th thing

Find the n-th prime

- NEED: isPrime(number)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

Design: isPrime(n)

Use paper (or equivalent) to design your solutions!

Design: isPrime(n)

Then you can compare your code your paper examples

```
def isPrime(n):
    if n < 2:
        return False
    for factor in range(2,n):
        if n % factor == 0:
            return False
    return True
```

Pattern: Find the n-th thing

Find the n-th prime

- Assume we have isPrime(number)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

Pattern: Find the n-th thing

Find the n-th prime

- Assume we have isPrime(number)

```
def nthPrime(n):
    found = 0
    guess = 0
    while found <= n: # Note: Does one more loop when found == n !!
        guess += 1
        if isPrime(guess):
            found += 1
    return guess
```

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17