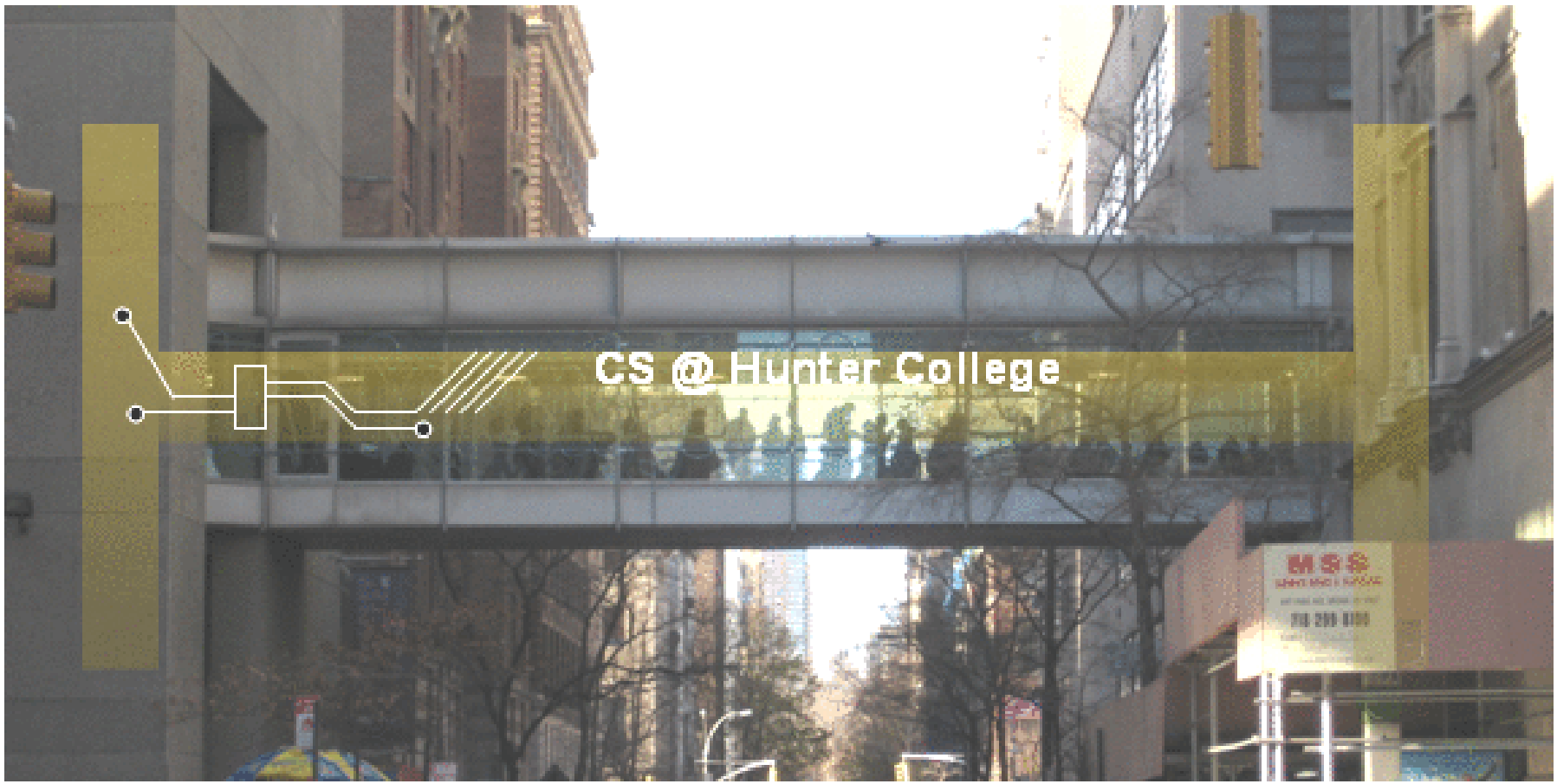
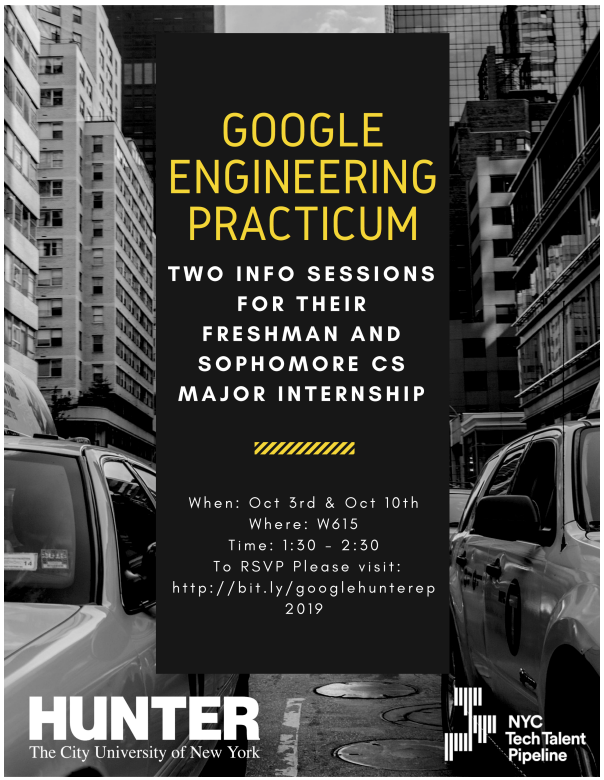
CSci 127: Introduction to Computer Science



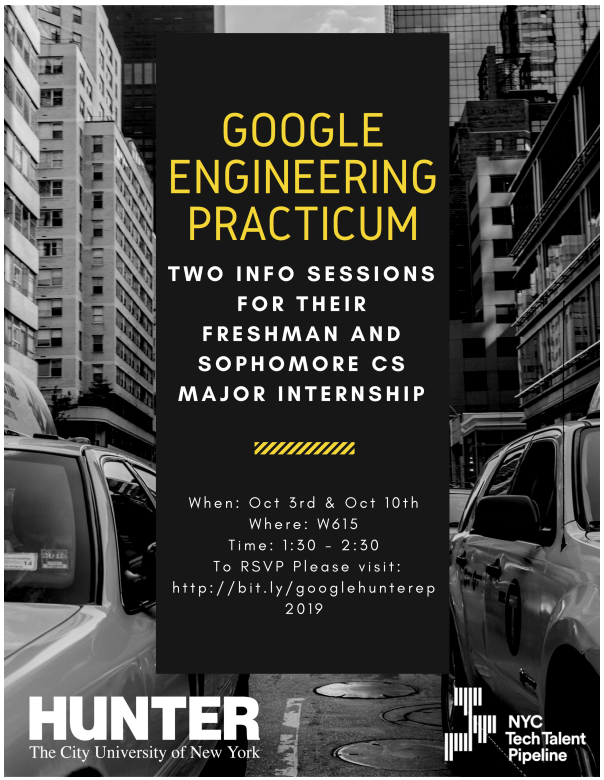
hunter.cuny.edu/csci

Announcements



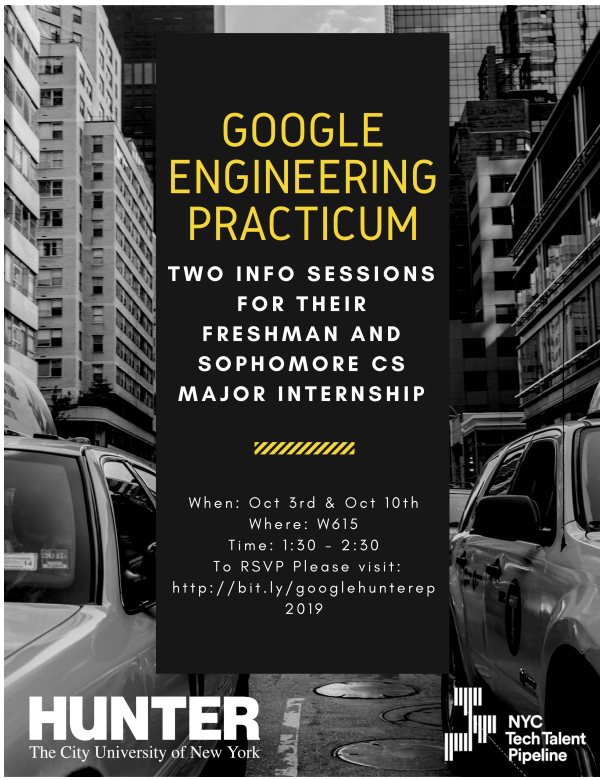
- Google Engineering Info Session: Today, 1:30pm, 614HW

Announcements



- Google Engineering Info Session: Today, 1:30pm, 614HW
- CS Survey:
  - Melina Diaconis, WiTNY

Announcements



- Google Engineering Info Session: Today, 1:30pm, 614HW
- CS Survey:
  - Melina Diaconis, WiTNY
  - Lily Caplan, president, Hunter ACM Student Chapter

Frequently Asked Questions

From lecture slips & recitation sections.

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus).

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus). Ex: `11 % 5` *is* `1`.

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus). Ex: `11 % 5` *is* `1`.
- What are types of variables?

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus). Ex: `11 % 5` *is* `1`.
- What are types of variables?
  Different kinds of information takes different amounts of space.
  Types we have seen so far: `int`, `float`, `str` *and objects (e.g. turtles).*

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus). Ex: `11 % 5` *is* `1`.
- What are types of variables?
  Different kinds of information takes different amounts of space.
  Types we have seen so far: `int`, `float`, `str` *and objects (e.g. turtles).*
- How can I tell strings from variables?

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus). Ex: `11 % 5` *is* `1`.
- What are types of variables?
  Different kinds of information takes different amounts of space.
  Types we have seen so far: `int`, `float`, `str` *and objects (e.g. turtles).*
- How can I tell strings from variables?
  Strings are surrounded by quotes (either single or double).

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus). Ex: `11 % 5` *is* `1`.
- What are types of variables?
  Different kinds of information takes different amounts of space.
  Types we have seen so far: `int`, `float`, `str` *and objects (e.g. turtles).*
- How can I tell strings from variables?
  Strings are surrounded by quotes (either single or double).
  Variables names (identifiers) for memory locations are not.

Frequently Asked Questions

From lecture slips & recitation sections.

- Where is the final? When are we taking it?
  Wednesday, 19 December, 9-11am, 118 North.
- Can we do more on colors, images, numpy & matplotlib?
  Yes, we will in Labs 6-9 & Lectures 6-9.
  Today, we'll focus on decisions, and logical expressions & circuits.
- What is pseudocode? Why do we use it?
  Pseudocode is the "informal high-level description of the operating principle of a computer program or other algorithm."
  We use it to write down the ideas, before getting deep into the details.
- What was that % symbol? Why is that math?
  It's the symbol for remainder (or modulus). Ex: `11 % 5` *is* `1`.
- What are types of variables?
  Different kinds of information takes different amounts of space.
  Types we have seen so far: `int`, `float`, `str` *and objects (e.g. turtles).*
- How can I tell strings from variables?
  Strings are surrounded by quotes (either single or double).
  Variables names (identifiers) for memory locations are not. Ex: `'num'` *vs.* `num`.

Frequently Asked Questions

From lecture slips & recitation sections.

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.

Frequently Asked Questions

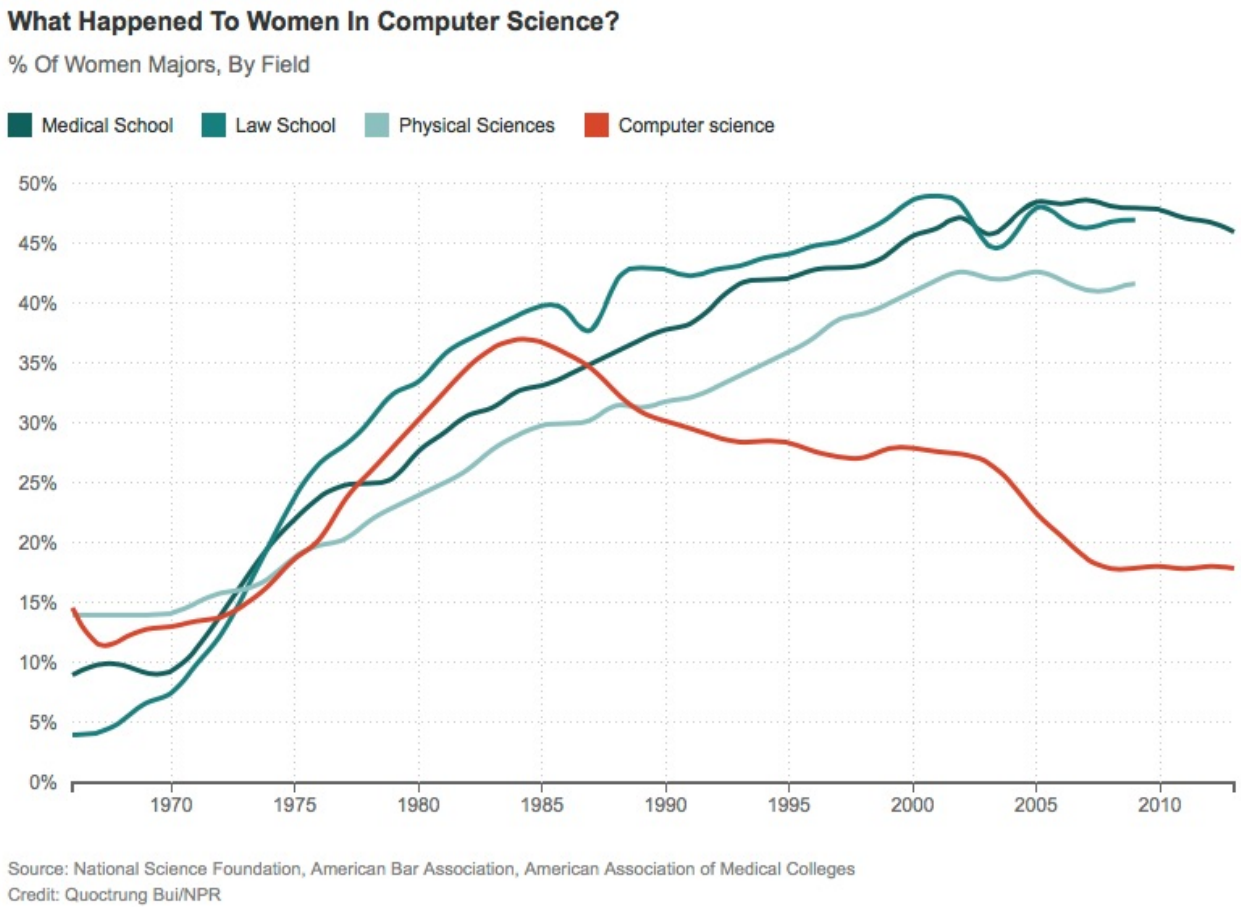
From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women. *Well, actually, there's not.*



(PlanetMoney)

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women. *Well, actually, there's not.*

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women. *Well, actually, there's not.*
- Yes, but I'm a guy. There's lots of guys.

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:
  - *Following Harvard: a rigorous & practical course to attract everyone.*

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:
  - *Following Harvard: a rigorous & practical course to attract everyone.*
  - *Automated grading to give immediate feedback.*

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:
  - *Following Harvard: a rigorous & practical course to attract everyone.*
  - *Automated grading to give immediate feedback.*
  - *Redirected resources to be student-focused (e.g. UTAs).*

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:
  - *Following Harvard: a rigorous & practical course to attract everyone.*
  - *Automated grading to give immediate feedback.*
  - *Redirected resources to be student-focused (e.g. UTAs).*
  - *Dedicated lab space & new computers.*

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:
  - *Following Harvard: a rigorous & practical course to attract everyone.*
  - *Automated grading to give immediate feedback.*
  - *Redirected resources to be student-focused (e.g. UTAs).*
  - *Dedicated lab space & new computers.*

  More students means more variety in upper division electives,

Frequently Asked Questions

From lecture slips & recitation sections.

- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:
  - *Following Harvard: a rigorous & practical course to attract everyone.*
  - *Automated grading to give immediate feedback.*
  - *Redirected resources to be student-focused (e.g. UTAs).*
  - *Dedicated lab space & new computers.*

  More students means more variety in upper division electives, more students with interests similar to yours,

Frequently Asked Questions

From lecture slips & recitation sections.

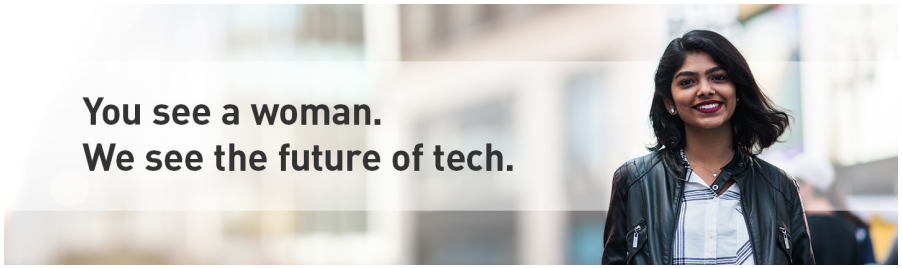
- Why all the fuss about women in computer science? There's lots of women.
  Well, actually, there's not.
- Yes, but I'm a guy. There's lots of guys.
  Not compared to other schools.
  Across CUNY there were < 1000 *students graduating with a tech degree (out of 24,000 annual graduates).*
  Versus Stanford where 90% take a CS course & half major in computer science.
- And how does this help me?
  Redesigned CSci 127:
  - *Following Harvard: a rigorous & practical course to attract everyone.*
  - *Automated grading to give immediate feedback.*
  - *Redirected resources to be student-focused (e.g. UTAs).*
  - *Dedicated lab space & new computers.*

  More students means more variety in upper division electives, more students with interests similar to yours, and more links to research and industry.

Today's Topics

- CS Survey
- Recap: Indexing, Slicing, & Decisions
- Logical Expressions
- Circuits

CS Survey



We Are Digital Enthusiasts

Jump start your future this January with a three-week Winternship.

What's a Winternship?
A Winternship is a PAID, three-week internship in NYC, open to first- and second-year women at CUNY during their January academic recess.

Why should I apply for a Winternship?
You'll learn more about job opportunities in tech and computing, build your resume, and expand your professional network.

Who can apply?
All first- and second-year women at CUNY who are interested in learning more about tech careers. You may be a computer science major, or you may not be. There are no academic requirements to apply. What are you waiting for?

Applications are now open for WiTNY Winternship!

APPLY TODAY!

witny.org/students

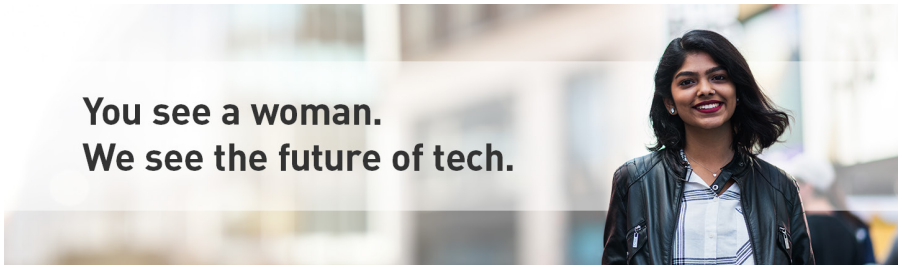
IMPORTANT DATES
- October 5, 2018: Applications due
- Mid-November: Placements announced
- January 7-24, 2019: Winternships take place in NYC and the surrounding tri-state area

Questions?
Maria DiKun, Program Coordinator, WiTNY | wit-ny@cornell.edu

WiTNY Women in Technology and Entrepreneurship in New York | CUNY The City University of New York | CORNELL TECH

- Maria Dikun & Melina Diaconis, WiTNY

CS Survey



We Are Digital Enthusiasts

Jump start your future this January with a three-week Winternship.

What's a Winternship?
A Winternship is a PAID, three-week internship in NYC, open to first- and second-year women at CUNY during their January academic recess.

Why should I apply for a Winternship?
You'll learn more about job opportunities in tech and computing, build your resume, and expand your professional network.

Who can apply?
All first- and second-year women at CUNY who are interested in learning more about tech careers. You may be a computer science major, or you may not be. There are no academic requirements to apply. What are you waiting for?

Applications are now open for WiTNY Winternship!

APPLY TODAY!

witny.org/students

IMPORTANT DATES
- October 5, 2018: Applications due
- Mid-November: Placements announced
- January 7-24, 2019: Winternships take place in NYC and the surrounding tri-state area

Questions?
Maria DiKun, Program Coordinator, WiTNY | wit-ny@cornell.edu

WiTNY Women in Technology and Entrepreneurship in New York | CUNY The City University of New York | CORNELL TECH

- Maria Dikun & Melina Diaconis, WiTNY
- Lily Caplan, president, Hunter ACM Student Chapter

Lecture Slip: In Pairs or Triples...

Some review:

1.
```
motto = "Mihi cura futuri"
print(motto[2:4])
print(motto[2:4].upper())
```

2.
```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

Recap: Indexing & Slicing

```
motto = "Mihi cura futuri"
print(motto[2:4])
print(motto[2:4].upper())
```

Recap: Indexing & Slicing

```
motto = "Mihi cura futuri"
print(motto[2:4])
print(motto[2:4].upper())
```

M	i	h	i		c	u	r	a		f	u	t	u	r	i

Recap: Indexing & Slicing

```
motto = "Mihi cura futuri"
print(motto[2:4])
print(motto[2:4].upper())
```

M	i	h	i		c	u	r	a		f	u	t	u	r	i
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Recap: Indexing & Slicing

```
motto = "Mihi cura futuri"
print(motto[2:4])
print(motto[2:4].upper())
```

M	i	**h**	**i**		c	u	r	a		f	u	t	u	r	i
0	1	**2**	**3**	4	5	6	7	8	9	10	11	12	13	14	15

Recap: Indexing & Slicing

```
motto = "Mihi cura futuri"
print(motto[2:4])
print(motto[2:4].upper())
```

M	i	**h**	**i**		c	u	r	a		f	u	t	u	r	i
0	1	**2**	**3**	4	5	6	7	8	9	10	11	12	13	14	15

Output:

```
hi
```

Recap: Indexing & Slicing

```
motto = "Mihi cura futuri"
print(motto[2:4])
print(motto[2:4].upper())
```

M	i	**h**	**i**		c	u	r	a		f	u	t	u	r	i
0	1	**2**	**3**	4	5	6	7	8	9	10	11	12	13	14	15

Output:

```
hi
HI
```

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

T	h	e		f	u	t	u	r	e		b	e	l	o	n	g	s
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

T	h	**e**		f	u	t	u	r	e		b	e	l	o	n	g	s
0	1	**2**	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

T	h	**e**		f	u	t	u	r	e		b	e	**l**	o	n	g	s
0	1	**2**	3	4	5	6	7	8	9	10	11	12	**13**	14	15	16	17

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

T	h	**e**		f	u	t	u	r	e		b	e	**l**	o	**n**	g	s
0	1	**2**	3	4	5	6	7	8	9	10	11	12	**13**	14	**15**	16	17

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

T	h	**e**		f	u	t	u	r	e		b	e	**l**	**o**	**n**	g	s
0	1	**2**	3	4	5	6	7	8	9	10	11	12	**13**	**14**	**15**	16	17

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

T	h	**e**		f	u	t	u	r	e		b	e	**l**	**o**	**n**	g	s
0	1	**2**	3	4	5	6	7	8	9	10	11	12	**13**	**14**	**15**	16	17

Output:

Recap: Indexing & Slicing

```
ER = "The future belongs to those who believe in the beauty of their dreams."
print(ER.upper()[2], ER[13], ER[2], "a", ER[15], ER[14], "r  R.")
```

T	h	**e**		f	u	t	u	r	e		b	e	**l**	**o**	**n**	g	s
0	1	**2**	3	4	5	6	7	8	9	10	11	12	**13**	**14**	**15**	16	17

Output:

E l e a n o r R.

In Pairs or Triples...

Some challenges with types & decisions:

```
#What are the types:
y1 = 2017
y2 = "2018"
print(type(y1))
print(type("y1"))
print(type(2017))
print(type("2017"))
print(type(y2))
print(type(y1/4.0))

x = int(y2) - y1
if x < 0:
    print(y2)
else:
    print(y1)
```

```
cents = 432
dollars = cents // 100
change = cents % 100
if dollars > 0:
    print('$'+str(dollars))
if change > 0:
    quarters = change // 25
    pennies = change % 25
    print(quarters, "quarters")
    print("and", pennies, "pennies")
```

Python Tutor

```
#What are the types:
y1 = 2017
y2 = "2018"
print(type(y1))
print(type("y1"))
print(type(2017))
print(type("2017"))
print(type(y2))
print(type(y1/4.0))

x = int(y2) - y1
if x < 0:
    print(y2)
else:
    print(y1)
```

(Demo with pythonTutor)

Decisions

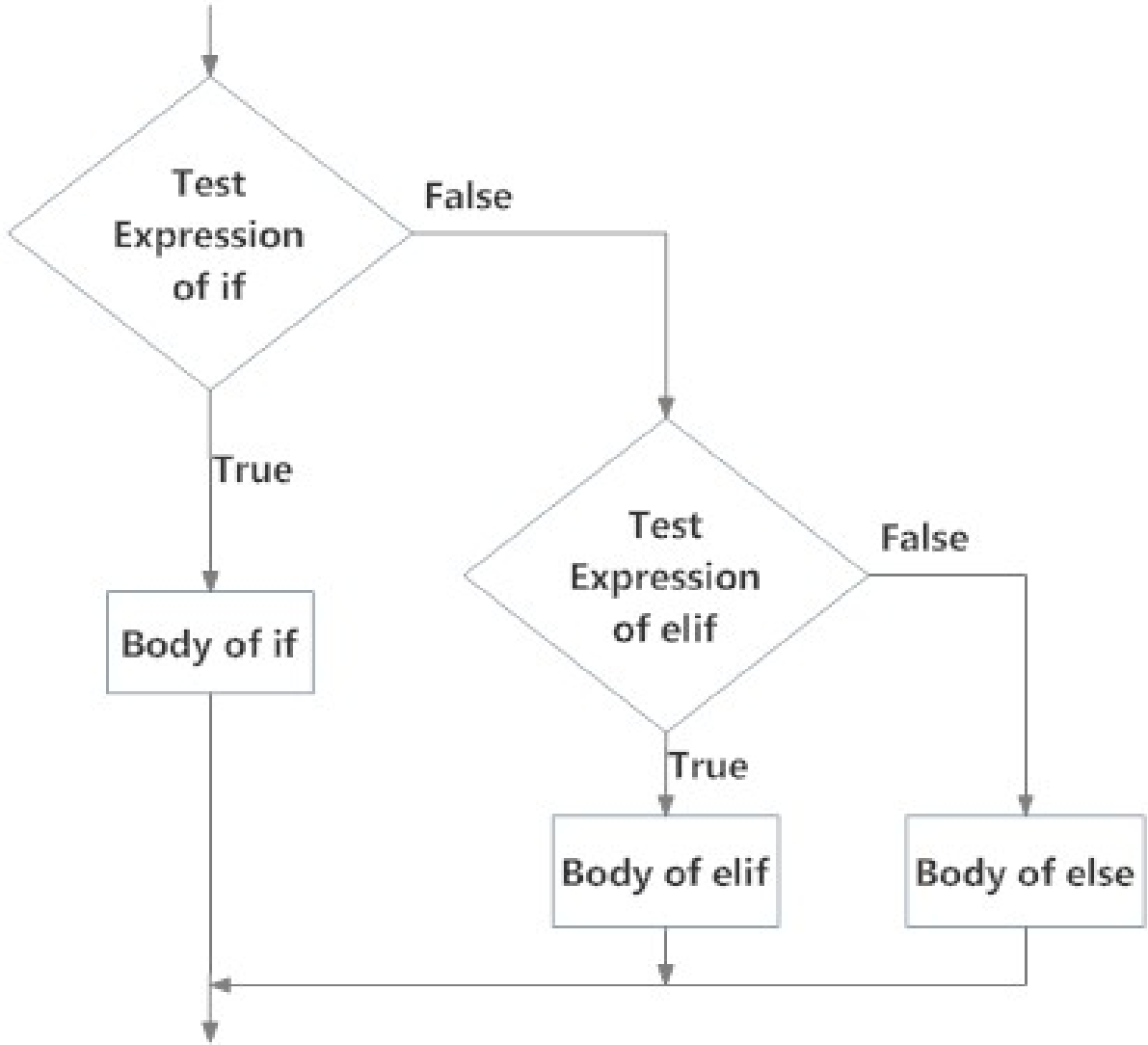
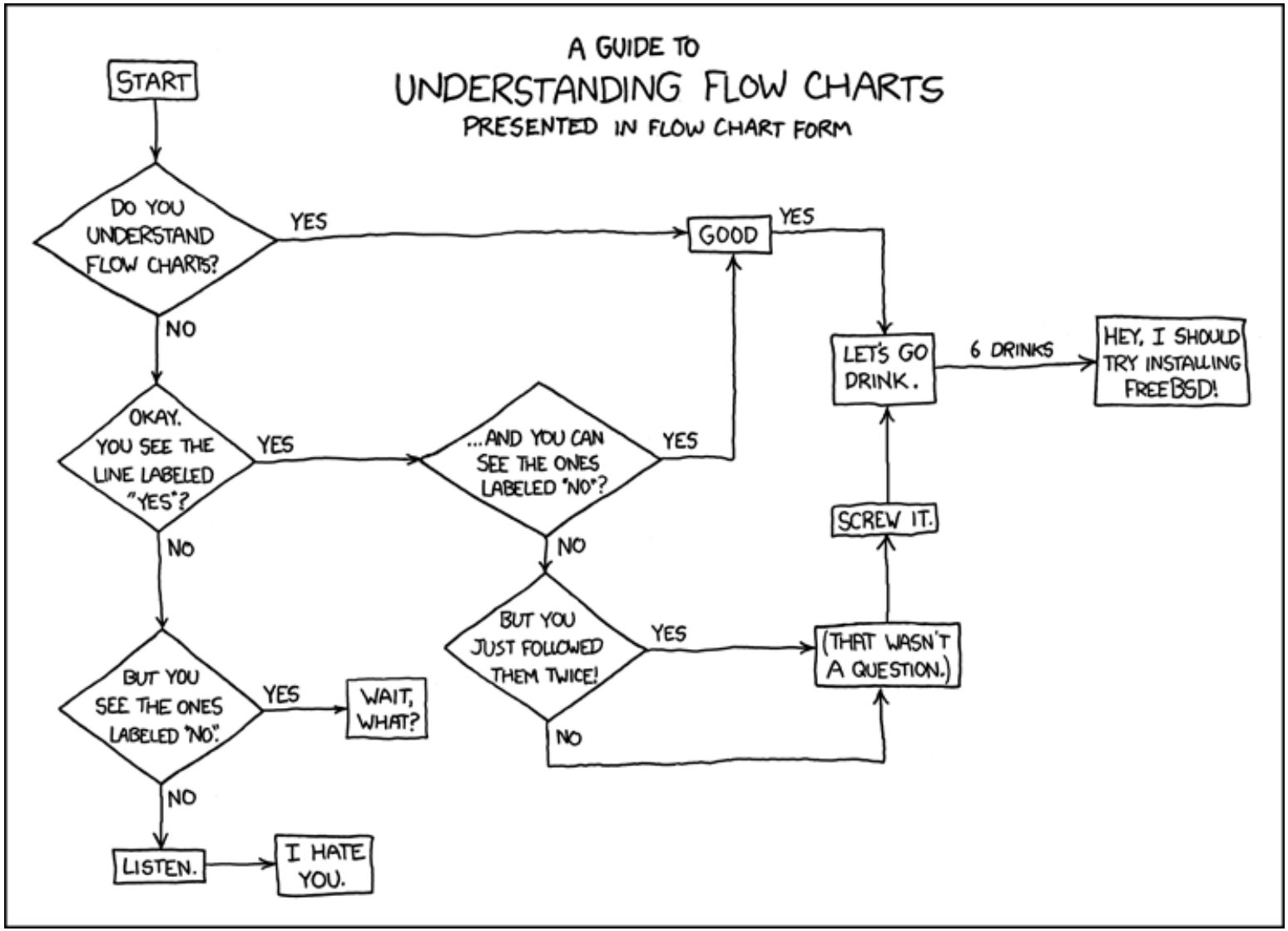


Fig: Operation of if...elif...else statement

(programiz)

Side Note: Reading Flow Charts



(xkcd/518)

In Pairs or Triples

Predict what the code will do:

```
origin = "Indian Ocean"
winds = 100
if (winds > 74):
    print("Major storm, called a ", end="")
    if origin == "Indian Ocean" or origin == "South Pacific":
        print("cyclone.")
    elif origin == "North Pacific":
        print("typhoon.")
    else:
        print("hurricane.")

visibility = 0.2
winds = 40
conditions = "blowing snow"
if (winds > 35) and (visibility < 0.25) and \
      (conditions == "blowing snow" or conditions == "heavy snow"):
    print("Blizzard!")
```

Python Tutor

```
origin = "Indian Ocean"
winds = 100
if (winds > 74):
    print("Major storm, called a ", end="")
    if origin == "Indian Ocean" or origin == "South Pacific":
        print("cyclone.")
    elif origin == "North Pacific":
        print("typhoon.")
    else:
        print("hurricane.")

visibility = 0.2
winds = 40
conditions = "blowing snow"
if (winds > 35) and (visibility < 0.25) and \
     (conditions == "blowing snow" or conditions == "heavy snow"):
    print("Blizzard!")
```

(Demo with `pythonTutor`)

Logical Operators

and

in1		in2	*returns:*
`False`	`and`	`False`	`False`
`False`	`and`	`True`	`False`
`True`	`and`	`False`	`False`
`True`	`and`	`True`	`True`

Logical Operators

and

in1		in2	*returns:*
`False`	`and`	`False`	`False`
`False`	`and`	`True`	`False`
`True`	`and`	`False`	`False`
`True`	`and`	`True`	`True`

or

in1		in2	*returns:*
`False`	`or`	`False`	`False`
`False`	`or`	`True`	`True`
`True`	`or`	`False`	`True`
`True`	`or`	`True`	`True`

Logical Operators

and

in1		in2	*returns:*
False	and	False	False
False	and	True	False
True	and	False	False
True	and	True	True

or

in1		in2	*returns:*
False	or	False	False
False	or	True	True
True	or	False	True
True	or	True	True

not

	in1	*returns:*
not	False	True
not	True	False

In Pairs or Triples

Predict what the code will do:

```
semHours = 18
reqHours = 120
if semHours >= 12:
    print('Full Time')
else:
    print('Part Time')

pace = reqHours // semHours
if reqHours % semHours != 0:
    pace = pace + 1
print('At this pace, you will graduate in', pace, 'semesters,')
yrs = pace / 2
print('(or', yrs, 'years).')


for i in range(1,20):
    if (i > 10) and (i % 2 == 1):
        print('oddly large')
    else:
        print(i)
```

Python Tutor

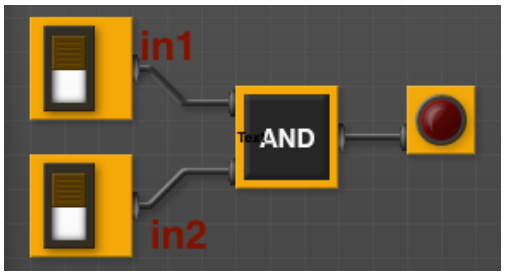
```
semHours = 18
reqHours = 120
if semHours >= 12:
     print('Full Time')
else:
     print('Part Time')

pace = reqHours // semHours
if reqHours % semHours != 0:
     pace = pace + 1
print('At this pace, you will graduate in', pace, 'semesters,')
yrs = pace / 2
print('(or', yrs, 'years).')


for i in range(1,20):
     if (i > 10) and (i % 2 == 1):
          print('oddly large')
     else:
          print(i)
```

(Demo with pythonTutor)

Circuit Demo

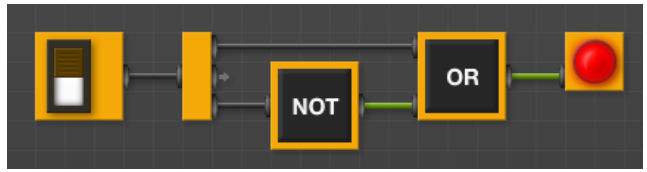


(Demo with `neuroproductions`)

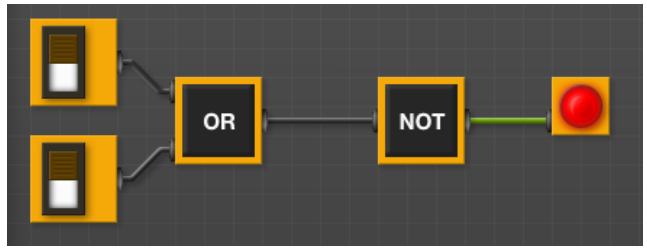
In Pairs or Triples

Predict when these expressions are true:

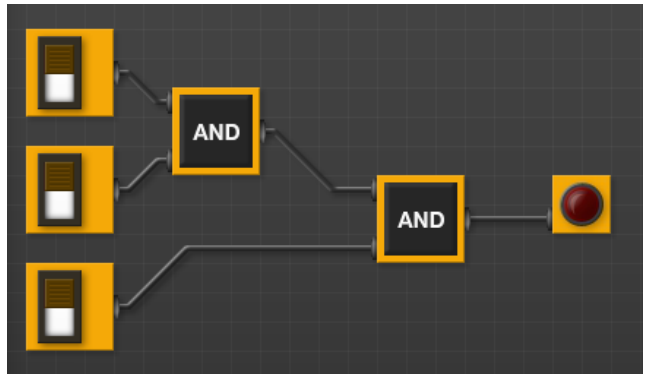
- `in1 or not in1`:



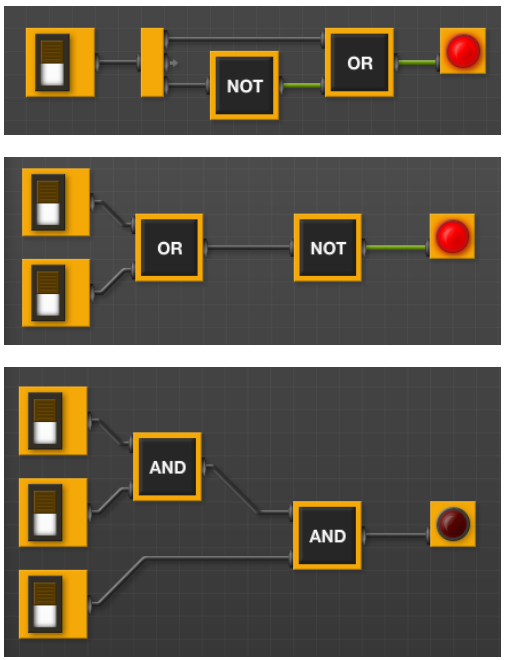
- `not(in1 or in2)`:



- `(in1 and in2) and in3`:



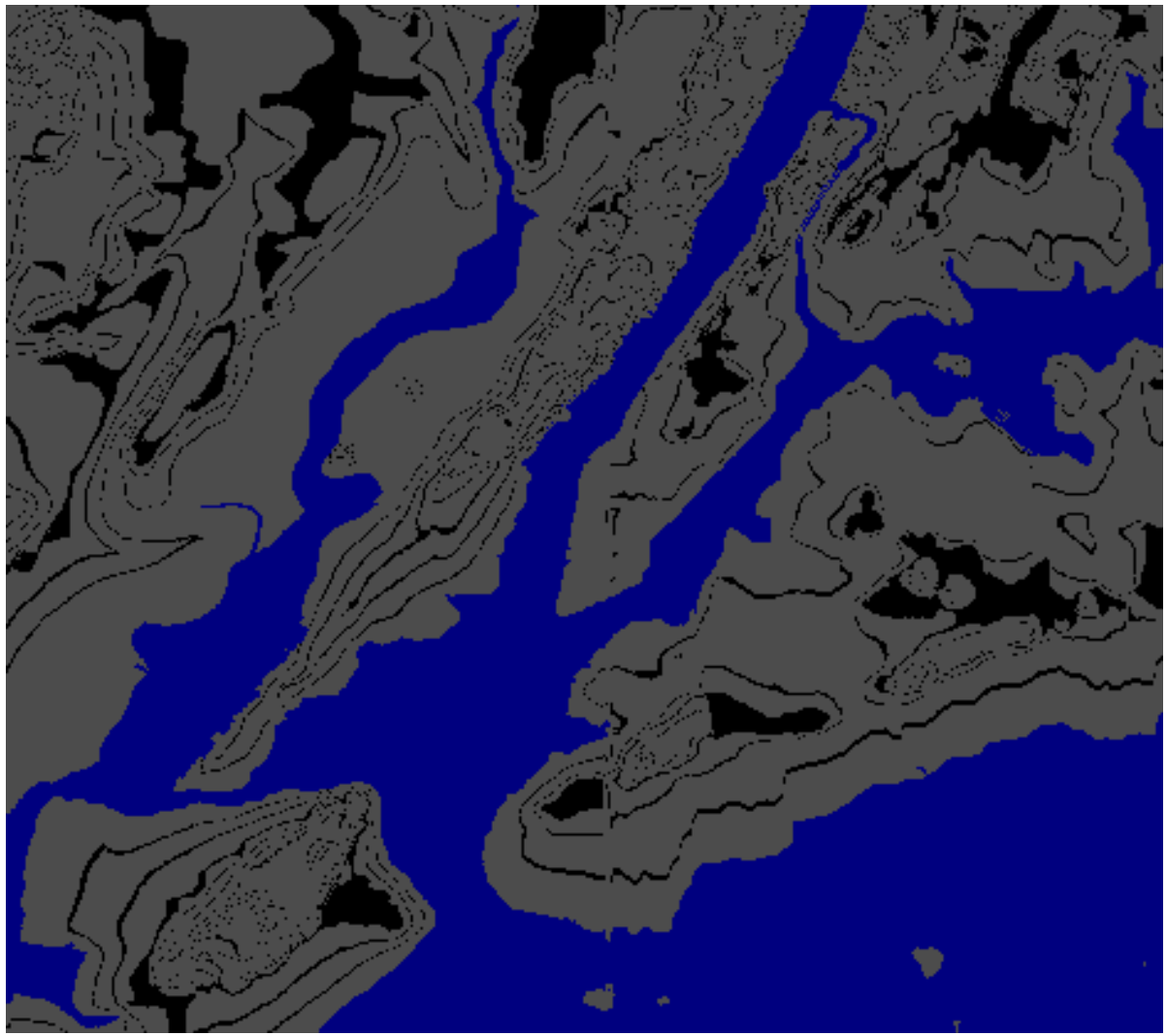
Circuit Demo



(Demo with `neuroproductions`)

Lecture Slip: In Pairs or Triples

Fill in the comments to describe the code (output below):



Recap

- On lecture slip, write down a topic you wish we had spent more time (and why).

Recap

- On lecture slip, write down a topic you wish we had spent more time (and why).
- In Python, we introduced:

Recap

- On lecture slip, write down a topic you wish we had spent more time (and why).
- In Python, we introduced:
  - Decisions
  - Logical Expressions
  - Circuits

Recap

- On lecture slip, write down a topic you wish we had spent more time (and why).
- In Python, we introduced:
  - Decisions
  - Logical Expressions
  - Circuits
- Pass your lecture slips to the aisles for the UTAs to collect.

Recap

- On lecture slip, write down a topic you wish we had spent more time (and why).
- In Python, we introduced:
  - Decisions
  - Logical Expressions
  - Circuits
- Pass your lecture slips to the aisles for the UTAs to collect.

Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.

Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.
- Pull out something to write on (not to be turned in).

Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.
- Pull out something to write on (not to be turned in).
- Lightning rounds:

Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.
- Pull out something to write on (not to be turned in).
- Lightning rounds:
  - write as much you can for 60 seconds;

Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.
- Pull out something to write on (not to be turned in).
- Lightning rounds:
  - write as much you can for 60 seconds;
  - followed by answer; and

Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.
- Pull out something to write on (not to be turned in).
- Lightning rounds:
  - write as much you can for 60 seconds;
  - followed by answer; and
  - repeat.

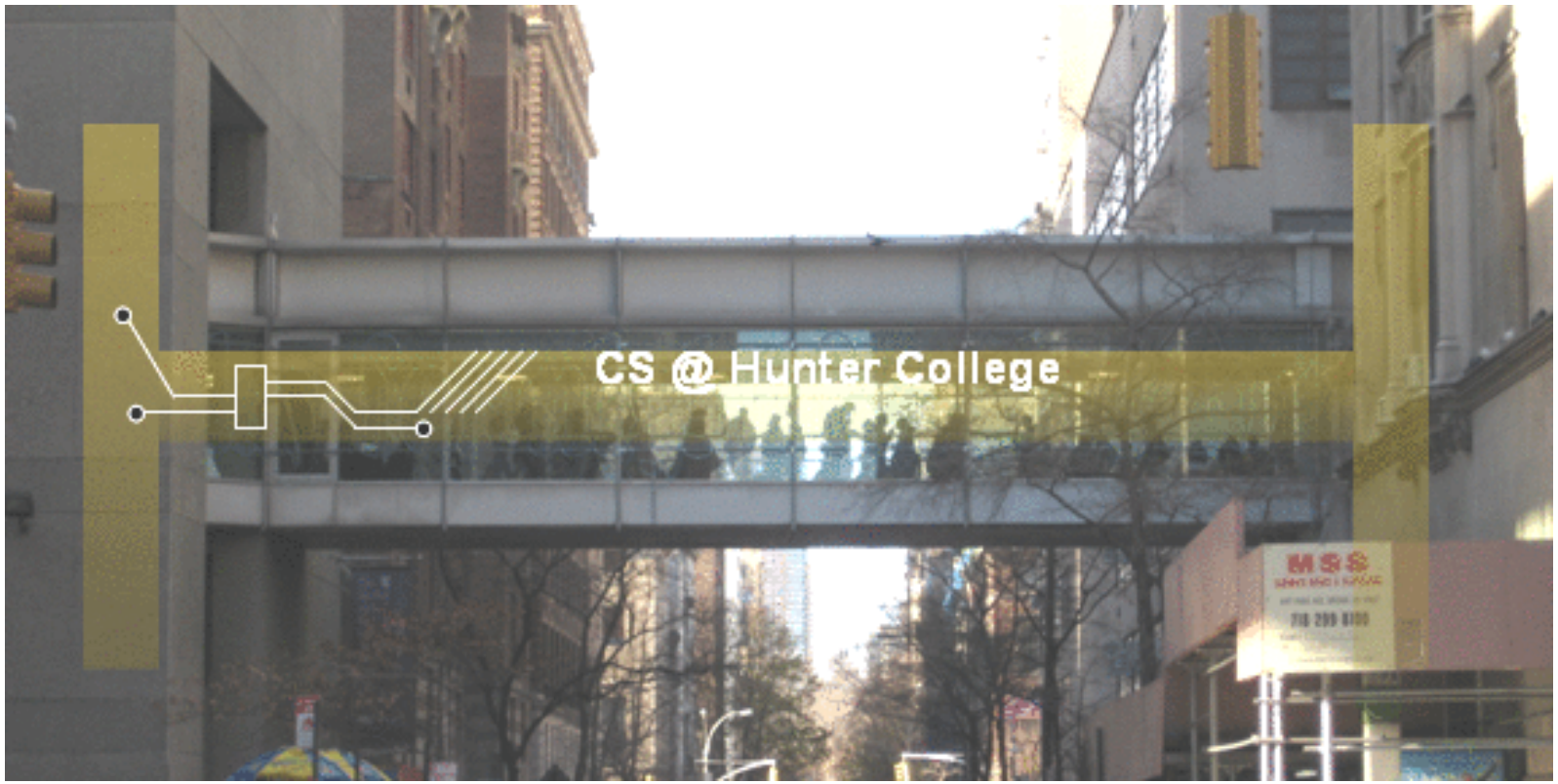
Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.
- Pull out something to write on (not to be turned in).
- Lightning rounds:
  - write as much you can for 60 seconds;
  - followed by answer; and
  - repeat.
- Past exams are on the webpage (under Final Exam Information).

Practice Quiz & Final Questions

- Since you must pass the final exam to pass the course, we end every lecture with final exam review.
- Pull out something to write on (not to be turned in).
- Lightning rounds:
  - write as much you can for 60 seconds;
  - followed by answer; and
  - repeat.
- Past exams are on the webpage (under Final Exam Information).
- We're starting with Fall 2018, Version 1.

Writing Boards



- Return writing boards as you leave...