

# CSci 127: Introduction to Computer Science



[hunter.cuny.edu/csci](http://hunter.cuny.edu/csci)

# Frequently Asked Questions

From lecture slips & recitation sections.

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

## Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid:*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN);*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN);*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

- Can I work ahead?

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

- Can I work ahead?

*Yes! All programs are available, on gradescope, 4 weeks before the deadline.*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

- Can I work ahead?

*Yes! All programs are available, on gradescope, 4 weeks before the deadline.*

- Last lecture didn't go into details of programming. Will you in the future?

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

- Can I work ahead?

*Yes! All programs are available, on gradescope, 4 weeks before the deadline.*

- Last lecture didn't go into details of programming. Will you in the future?

*No worries— today, we'll dive into the details of for, range and string methods.*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

- Can I work ahead?

*Yes! All programs are available, on gradescope, 4 weeks before the deadline.*

- Last lecture didn't go into details of programming. Will you in the future?

*No worries— today, we'll dive into the details of for, range and string methods.*

- You said "when you take second semester..." I just took this class for Pathways...

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

- Can I work ahead?

*Yes! All programs are available, on gradescope, 4 weeks before the deadline.*

- Last lecture didn't go into details of programming. Will you in the future?

*No worries— today, we'll dive into the details of for, range and string methods.*

- You said "when you take second semester..." I just took this class for Pathways...

*This is Pathways, but we hope that you will be a CS major/minor.*

# Frequently Asked Questions

From lecture slips & recitation sections.

- When is recitation? It's not on my schedule.

*The course is hybrid: 1.5 hours of lecture (T 9:45-11, 118 HN); 0.5 hours quizzes (self-scheduled, 1001E HN); 1.0 hours on-line lab exercises (do anywhere).*

- When is the midterm?

*There is no midterm. Instead there's required weekly quizzes.*

- When is the final?

*Monday, 16 December, 9-11am.*

- Can I submit late homework?

*No. Instead we drop the 5 lowest grades.*

- I missed class. Do you need documentation?

*No. Missing lecture & quiz grades are replaced by your final exam score.*

*If you will miss  $\geq 3$  weeks ( $> 20\%$ ), see us about taking this in a future term.*

- Can I work ahead?

*Yes! All programs are available, on gradescope, 4 weeks before the deadline.*

- Last lecture didn't go into details of programming. Will you in the future?

*No worries— today, we'll dive into the details of for, range and string methods.*

- You said "when you take second semester..." I just took this class for Pathways...

*This is Pathways, but we hope that you will be a CS major/minor.*

*We also hope: "Get your education don't forget whence you came..."*

# Today's Topics



- Research Survey
- For-loops
- `range()`
- Variables
- Characters
- Strings

# Today's Topics



- **Research Survey**
  - For-loops
  - range()
  - Variables
  - Characters
  - Strings

## Why All the Handouts Today?

|   |                              |  |
|---|------------------------------|--|
| 1. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 2. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 3. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 4. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 5. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 6. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 7. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 8. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 9. Who is present? Who can receive help for the category below  | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |
| 10. Who is present? Who can receive help for the category below | <input type="checkbox"/> Yes | <input checked="" type="checkbox"/> No |

## Lecture Slip

## Overview

|  |  |
|--|--|
| <b>STUDY INFORMATION</b><br><b>RESEARCHER'S NAME:</b><br><b>INSTITUTION:</b><br><p>If you have questions about your rights as a research participant, or you have concerns or complaints about this study, you may contact the Office of Human Research Protection (OHRP) or the Office of Compliance Administration at one of the following telephone numbers. We will provide you with a copy of the OHRP brochure "Your Rights as a Research Participant".</p> <p>OHRP office at the Vice Chancellor for Research<br/>         300 East 10th Street, Suite 1000<br/>         Box 10000<br/>         Tel: (404) 727-6000</p> <p>Office of Compliance Administration<br/>         Box 10000<br/>         Tel: (404) 727-6000</p> <p><b>Signature of PI/Principal Investigator:</b></p> <p>If you agree to participate in this research study, please sign and date below (you will be given a copy of the informed consent form).</p> <p><b>Consent Form of Participation:</b></p> <p><b>Signature of PI/Principal Investigator:</b> _____</p> <p><b>Signature of Institutional Review Board Chair:</b> _____</p> <p><b>Signature of Institutional Monitoring Committee:</b> _____</p> <p><b>Protection of Individual Monitoring Committee:</b></p> <p><b>Signature of Individual Monitoring Committee:</b> _____</p> | <b>STUDY INFORMATION</b><br><b>RESEARCHER'S NAME:</b><br><b>INSTITUTION:</b><br><p>If you have questions about your rights as a research participant, or you have concerns or complaints about this study, you may contact the Office of Human Research Protection (OHRP) or the Office of Compliance Administration at one of the following telephone numbers. We will provide you with a copy of the OHRP brochure "Your Rights as a Research Participant".</p> <p>OHRP office at the Vice Chancellor for Research<br/>         300 East 10th Street, Suite 1000<br/>         Box 10000<br/>         Tel: (404) 727-6000</p> <p>Office of Compliance Administration<br/>         Box 10000<br/>         Tel: (404) 727-6000</p> <p><b>Signature of PI/Principal Investigator:</b></p> <p>If you agree to participate in this research study, please sign and date below (you will be given a copy of the informed consent form).</p> <p><b>Consent Form of Participation:</b></p> <p><b>Signature of PI/Principal Investigator:</b> _____</p> <p><b>Signature of Institutional Review Board Chair:</b> _____</p> <p><b>Signature of Institutional Monitoring Committee:</b> _____</p> <p><b>Protection of Individual Monitoring Committee:</b></p> <p><b>Signature of Individual Monitoring Committee:</b> _____</p> |
|--|--|

## Consent Form

Survey

# Research Study

This study investigates students' emotions, cognitions, motivation, and learning related to computer science.



Part 1: Consists of two brief surveys completed in class.

Prof. John Ranellucci

Educational Psychology

# Research Study

This study investigates students' emotions, cognitions, motivation, and learning related to computer science.



Part 1: Consists of two brief surveys completed in class.

Part 2: I'm asking you to answer two extra questions at the end of your "lecture slips".

Prof. John Ranellucci

Educational Psychology

# Research Study

This study investigates students' emotions, cognitions, motivation, and learning related to computer science.



- Part 1: Consists of two brief surveys completed in class.
- Part 2: I'm asking you to answer two extra questions at the end of your "lecture slips".
- Part 3: Consists of two surveys available online.

Prof. John Ranellucci

Educational Psychology

# Research Study

This study investigates students' emotions, cognitions, motivation, and learning related to computer science.



- Part 1: Consists of two brief surveys completed in class.
- Part 2: I'm asking you to answer two extra questions at the end of your "lecture slips".
- Part 3: Consists of two surveys available online.  
(Little longer and participants will be compensated with a \$20 Amazon gift certificate for completing both surveys.)

Prof. John Ranellucci

Educational Psychology

# Research Study

This study investigates students' emotions, cognitions, motivation, and learning related to computer science.



Prof. John Ranellucci  
Educational Psychology

- Part 1: Consists of two brief surveys completed in class.
- Part 2: I'm asking you to answer two extra questions at the end of your "lecture slips".
- Part 3: Consists of two surveys available online.  
(Little longer and participants will be compensated with a \$20 Amazon gift certificate for completing both surveys.)

*This study is not part of the class, and no individual analyses will be shared with your instructor. Survey links for the online survey will be emailed to all of you, other surveys will be distributed in class.*

# Today's Topics



- Research Survey
- **For-loops**
- `range()`
- Variables
- Characters
- Strings

## In Pairs or Triples...

*Some review and some novel challenges:*

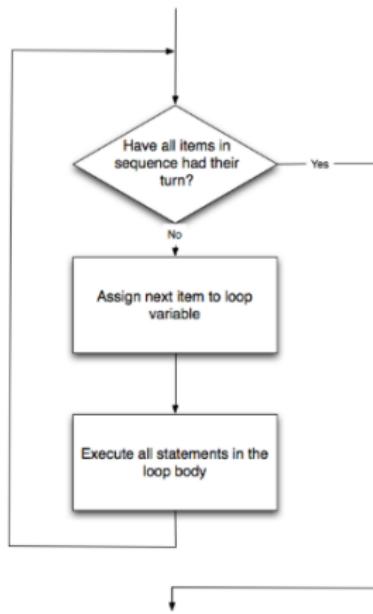
```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color) |  
10    for i in range(2):  
11        for j in range(2):  
12            print('Look around,')  
13            print('How lucky we are to be alive!')
```

# Python Tutor

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color) |  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

(Demo with pythonTutor)

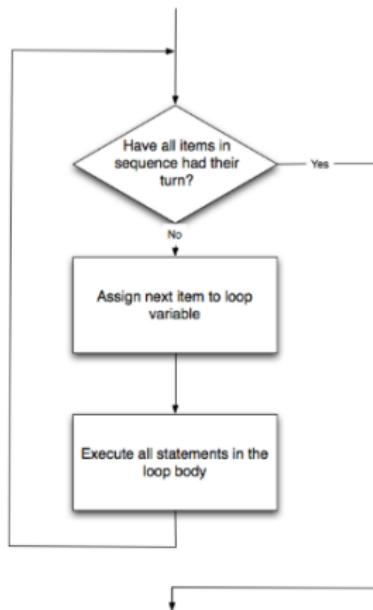
# for-loop



```
for i in list:  
    statement1  
    statement2  
    statement3
```

*How to Think Like CS, §4.5*

# for-loop



```
for i in list:  
    statement1  
    statement2  
    statement3
```

where `list` is a list of items:

- stated explicitly (e.g. `[1,2,3]`) or
- generated by a function,  
e.g. `range()`.

*How to Think Like CS, §4.5*

# In Pairs or Triples...

*Some review and some novel challenges:*

```
1 #Predict what will be printed:  
2  
3 for num in [2,4,6,8,10]:  
4     print(num)  
5  
6 sum = 0  
7 for x in range(0,12,2):  
8     print(x)  
9     sum = sum + x  
10  
11 print(x)  
12  
13 for c in "ABCD":  
14     print(c)
```

# Today's Topics



- Research Survey
- For-loops
- **range()**
- Variables
- Characters
- Strings

# Python Tutor

```
1 #Predict what will be printed:  
2  
3 for num in [2,4,6,8,10]:  
4     print(num)  
5  
6 sum = 0  
7 for x in range(0,12,2):  
8     print(x)  
9     sum = sum + x  
10  
11 print(x)  
12  
13 for c in "ABCD":  
14     print(c)
```

(Demo with pythonTutor)

# range()

Simplest version:

- `range(stop)`



# range()



Simplest version:

- `range(stop)`
- Produces a list: `[0,1,2,3,...,stop-1]`

# range()



Simplest version:

- `range(stop)`
- Produces a list:  $[0,1,2,3,\dots,stop-1]$
- For example, if you want the list  $[0,1,2,3,\dots,100]$ , you would write:

# range()



Simplest version:

- `range(stop)`
- Produces a list:  $[0,1,2,3,\dots,stop-1]$
- For example, if you want the list  $[0,1,2,3,\dots,100]$ , you would write:

```
range(101)
```

# `range()`

What if you wanted to start somewhere else:



# `range()`

What if you wanted to start somewhere else:

- `range(start, stop)`



# range()

What if you wanted to start somewhere else:

- `range(start, stop)`
- Produces a list:  
`[start,start+1,...,stop-1]`



# range()



What if you wanted to start somewhere else:

- `range(start, stop)`
- Produces a list:  
`[start,start+1,...,stop-1]`
- For example, if you want the list  
`[10,11,...,20]`  
you would write:

# range()



What if you wanted to start somewhere else:

- `range(start, stop)`
- Produces a list:  
`[start,start+1,...,stop-1]`
- For example, if you want the list  
`[10,11,...,20]`  
you would write:

```
range(10,21)
```

# `range()`

What if you wanted to count by twos, or some other number:



# range()

What if you wanted to count by twos, or some other number:

- `range(start, stop, step)`



# range()

What if you wanted to count by twos, or some other number:

- `range(start, stop, step)`
- Produces a list:  
`[start, start+step, start+2*step..., last]`  
(where last is the largest  $\text{start}+k*\text{step}$  less than stop)



# range()

What if you wanted to count by twos, or some other number:



- `range(start, stop, step)`
- Produces a list:  
 $[start, start+step, start+2*step\dots, last]$   
(where last is the largest  $start+k*step$  less than stop)
- For example, if you want the list  
[5,10,...,50]  
you would write:

# range()



What if you wanted to count by twos, or some other number:

- `range(start, stop, step)`
- Produces a list:  
 $[start, start+step, start+2*step\dots, last]$   
(where last is the largest  $start+k*step$  less than stop)
- For example, if you want the list  
[5,10,...,50]  
you would write:

```
range(5,51,5)
```

# In summary: range()



The three versions:

# In summary: range()



The three versions:

- `range(stop)`

## In summary: range()



The three versions:

- `range(stop)`
- `range(start, stop)`

## In summary: range()



The three versions:

- `range(stop)`
- `range(start, stop)`
- `range(start, stop, step)`

# Today's Topics



- Research Survey
- For-loops
- `range()`
- **Variables**
- Characters
- Strings

# Variables

- A **variable** is a reserved memory location for storing a value.



# Variables



- A **variable** is a reserved memory location for storing a value.
- Different kinds, or **types**, of values need different amounts of space:
  - ▶ **int**: integer or whole numbers

# Variables



- A **variable** is a reserved memory location for storing a value.
- Different kinds, or **types**, of values need different amounts of space:
  - ▶ **int**: integer or whole numbers
  - ▶ **float**: floating point or real numbers

# Variables



- A **variable** is a reserved memory location for storing a value.
- Different kinds, or **types**, of values need different amounts of space:
  - ▶ **int**: integer or whole numbers
  - ▶ **float**: floating point or real numbers
  - ▶ **string**: sequence of characters

# Variables



- A **variable** is a reserved memory location for storing a value.
- Different kinds, or **types**, of values need different amounts of space:
  - ▶ **int**: integer or whole numbers
  - ▶ **float**: floating point or real numbers
  - ▶ **string**: sequence of characters
  - ▶ **list**: a sequence of items

# Variables



- A **variable** is a reserved memory location for storing a value.
- Different kinds, or **types**, of values need different amounts of space:
  - ▶ **int**: integer or whole numbers
  - ▶ **float**: floating point or real numbers
  - ▶ **string**: sequence of characters
  - ▶ **list**: a sequence of items
    - e.g. [3, 1, 4, 5, 9] or
    - [‘violet’, ‘purple’, ‘indigo’]

# Variables



- A **variable** is a reserved memory location for storing a value.
- Different kinds, or **types**, of values need different amounts of space:
  - ▶ **int**: integer or whole numbers
  - ▶ **float**: floating point or real numbers
  - ▶ **string**: sequence of characters
  - ▶ **list**: a sequence of items
    - e.g. [3, 1, 4, 5, 9] or
    - [‘violet’, ‘purple’, ‘indigo’]
  - ▶ **class variables**: for complex objects, like turtles.

# Variable Names

- There's some rules about valid names for variables.



# Variable Names

- There's some rules about valid names for variables.
- Can use the underscore ('\_'), upper and lower case letters.



# Variable Names



- There's some rules about valid names for variables.
- Can use the underscore ('\_'), upper and lower case letters.
- Can also use numbers, just can't start a name with a number.

# Variable Names



- There's some rules about valid names for variables.
- Can use the underscore ('\_'), upper and lower case letters.
- Can also use numbers, just can't start a name with a number.
- Can't use symbols (like '+' or '\*') since used for arithmetic.

# Variable Names



- There's some rules about valid names for variables.
- Can use the underscore ('\_'), upper and lower case letters.
- Can also use numbers, just can't start a name with a number.
- Can't use symbols (like '+' or '\*') since used for arithmetic.
- Can't use some words that Python has reserved for itself (e.g. `for`).  
(List of reserved words in *Think CS*, §2.5.)

# Today's Topics



- Research Survey
- For-loops
- `range()`
- Variables
- **Characters**
- Strings

# Standardized Code for Characters

American Standard Code for Information Interchange (ASCII), 1960.

# Standardized Code for Characters

American Standard Code for Information Interchange (ASCII), 1960.  
(New version called: Unicode).

# Standardized Code for Characters

American Standard Code for Information Interchange (ASCII), 1960.  
(New version called: Unicode).

## ASCII TABLE

| Decimal | Hex | Char                   | Decimal | Hex | Char    | Decimal | Hex | Char | Decimal | Hex | Char  |
|---------|-----|------------------------|---------|-----|---------|---------|-----|------|---------|-----|-------|
| 0       | 0   | [NULL]                 | 32      | 20  | [SPACE] | 64      | 40  | @    | 96      | 60  | '     |
| 1       | 1   | [START OF HEADING]     | 33      | 21  | !       | 65      | 41  | A    | 97      | 61  | a     |
| 2       | 2   | [START OF TEXT]        | 34      | 22  | "       | 66      | 42  | B    | 98      | 62  | b     |
| 3       | 3   | [END OF TEXT]          | 35      | 23  | #       | 67      | 43  | C    | 99      | 63  | c     |
| 4       | 4   | [END OF TRANSMISSION]  | 36      | 24  | \$      | 68      | 44  | D    | 100     | 64  | d     |
| 5       | 5   | [ENQUIRY]              | 37      | 25  | %       | 69      | 45  | E    | 101     | 65  | e     |
| 6       | 6   | [ACKNOWLEDGE]          | 38      | 26  | &       | 70      | 46  | F    | 102     | 66  | f     |
| 7       | 7   | [BELL]                 | 39      | 27  | ,       | 71      | 47  | G    | 103     | 67  | g     |
| 8       | 8   | [BACKSPACE]            | 40      | 28  | (       | 72      | 48  | H    | 104     | 68  | h     |
| 9       | 9   | [HORIZONTAL TAB]       | 41      | 29  | )       | 73      | 49  | I    | 105     | 69  | i     |
| 10      | A   | [LINE FEED]            | 42      | 2A  | *       | 74      | 4A  | J    | 106     | 6A  | j     |
| 11      | B   | [VERTICAL TAB]         | 43      | 2B  | +       | 75      | 4B  | K    | 107     | 6B  | k     |
| 12      | C   | [FORM FEED]            | 44      | 2C  | ,       | 76      | 4C  | L    | 108     | 6C  | l     |
| 13      | D   | [CARRIAGE RETURN]      | 45      | 2D  | -       | 77      | 4D  | M    | 109     | 6D  | m     |
| 14      | E   | [SHIFT OUT]            | 46      | 2E  | .       | 78      | 4E  | N    | 110     | 6E  | n     |
| 15      | F   | [SHIFT IN]             | 47      | 2F  | /       | 79      | 4F  | O    | 111     | 6F  | o     |
| 16      | 10  | [DATA LINK ESCAPE]     | 48      | 30  | 0       | 80      | 50  | P    | 112     | 70  | p     |
| 17      | 11  | [DEVICE CONTROL 1]     | 49      | 31  | 1       | 81      | 51  | Q    | 113     | 71  | q     |
| 18      | 12  | [DEVICE CONTROL 2]     | 50      | 32  | 2       | 82      | 52  | R    | 114     | 72  | r     |
| 19      | 13  | [DEVICE CONTROL 3]     | 51      | 33  | 3       | 83      | 53  | S    | 115     | 73  | s     |
| 20      | 14  | [DEVICE CONTROL 4]     | 52      | 34  | 4       | 84      | 54  | T    | 116     | 74  | t     |
| 21      | 15  | [NEGATIVE ACKNOWLEDGE] | 53      | 35  | 5       | 85      | 55  | U    | 117     | 75  | u     |
| 22      | 16  | [SYNCHRONOUS IDLE]     | 54      | 36  | 6       | 86      | 56  | V    | 118     | 76  | v     |
| 23      | 17  | [END OF TRANS. BLOCK]  | 55      | 37  | 7       | 87      | 57  | W    | 119     | 77  | w     |
| 24      | 18  | [CANCEL]               | 56      | 38  | 8       | 88      | 58  | X    | 120     | 78  | x     |
| 25      | 19  | [END OF MEDIUM]        | 57      | 39  | 9       | 89      | 59  | Y    | 121     | 79  | y     |
| 26      | 1A  | [SUBSTITUTE]           | 58      | 3A  | :       | 90      | 5A  | Z    | 122     | 7A  | z     |
| 27      | 1B  | [ESCAPE]               | 59      | 3B  | ;       | 91      | 5B  | [    | 123     | 7B  | {     |
| 28      | 1C  | [FILE SEPARATOR]       | 60      | 3C  | <       | 92      | 5C  | \    | 124     | 7C  |       |
| 29      | 1D  | [GROUP SEPARATOR]      | 61      | 3D  | =       | 93      | 5D  | ]    | 125     | 7D  | }     |
| 30      | 1E  | [RECORD SEPARATOR]     | 62      | 3E  | >       | 94      | 5E  | ^    | 126     | 7E  | -     |
| 31      | 1F  | [UNIT SEPARATOR]       | 63      | 3F  | ?       | 95      | 5F  | -    | 127     | 7F  | [DEL] |

(wiki)

# Converting from Character to Code:

*(There is an ASCII table on the back of today's lecture slip.)*

**ASCII TABLE**

| Decimal Num Char | Octal Num Char | Hex Num Char | Decimal Num Char | Octal Num Char | Hex Num Char |
|------------------|----------------|--------------|------------------|----------------|--------------|
| 0                | 0              | 0            | 1                | 1              | 1            |
| 2                | 2              | 2            | 3                | 3              | 3            |
| 4                | 4              | 4            | 5                | 5              | 5            |
| 6                | 6              | 6            | 7                | 7              | 7            |
| 8                | 10             | 8            | 9                | 11             | 9            |
| 10               | 12             | A            | 11               | 13             | B            |
| 12               | 14             | C            | 13               | 15             | D            |
| 14               | 16             | E            | 15               | 17             | F            |
| 16               | 20             | 10           | 17               | 21             | 11           |
| 18               | 22             | 12           | 19               | 23             | 13           |
| 20               | 24             | 14           | 21               | 25             | 15           |
| 22               | 26             | 16           | 23               | 27             | 17           |
| 24               | 28             | 18           | 25               | 29             | 19           |
| 26               | 30             | 1A           | 27               | 31             | 1B           |
| 28               | 32             | 1C           | 29               | 33             | 1D           |
| 30               | 34             | 1E           | 31               | 35             | 1F           |
| 32               | 36             | 20           | 33               | 37             | 21           |
| 34               | 38             | 22           | 35               | 39             | 23           |
| 36               | 40             | 24           | 37               | 41             | 25           |
| 38               | 42             | 26           | 39               | 43             | 27           |
| 40               | 44             | 28           | 41               | 45             | 29           |
| 42               | 46             | 2A           | 43               | 47             | 2B           |
| 44               | 48             | 2C           | 45               | 49             | 2D           |
| 46               | 50             | 2E           | 47               | 51             | 2F           |
| 48               | 52             | 30           | 49               | 53             | 31           |
| 50               | 54             | 32           | 51               | 55             | 33           |
| 52               | 56             | 34           | 53               | 57             | 35           |
| 54               | 58             | 36           | 55               | 59             | 37           |
| 56               | 60             | 38           | 57               | 61             | 39           |
| 58               | 62             | 3A           | 59               | 63             | 3B           |
| 60               | 64             | 3C           | 61               | 65             | 3D           |
| 62               | 66             | 3E           | 63               | 67             | 3F           |
| 64               | 68             | 40           | 65               | 69             | 41           |
| 66               | 70             | 42           | 67               | 71             | 43           |
| 68               | 72             | 44           | 69               | 73             | 45           |
| 70               | 74             | 46           | 71               | 75             | 47           |
| 72               | 76             | 48           | 73               | 77             | 49           |
| 74               | 78             | 4A           | 75               | 79             | 4B           |
| 76               | 80             | 4C           | 77               | 81             | 4D           |
| 78               | 82             | 4E           | 79               | 83             | 4F           |
| 80               | 84             | 50           | 81               | 85             | 51           |
| 82               | 86             | 52           | 83               | 87             | 53           |
| 84               | 88             | 54           | 85               | 89             | 55           |
| 86               | 90             | 56           | 87               | 91             | 57           |
| 88               | 92             | 58           | 89               | 93             | 59           |
| 90               | 94             | 5A           | 91               | 95             | 5B           |
| 92               | 96             | 5C           | 93               | 97             | 5D           |
| 94               | 98             | 5E           | 95               | 99             | 5F           |
| 96               | 100            | 60           | 97               | 101            | 61           |
| 98               | 102            | 62           | 99               | 103            | 63           |
| 100              | 104            | 64           | 101              | 105            | 65           |
| 102              | 106            | 66           | 103              | 107            | 67           |
| 104              | 108            | 68           | 105              | 109            | 69           |
| 106              | 110            | 6A           | 107              | 111            | 6B           |
| 108              | 112            | 6C           | 109              | 113            | 6D           |
| 110              | 114            | 6E           | 111              | 115            | 6F           |
| 112              | 116            | 70           | 113              | 117            | 71           |
| 114              | 118            | 72           | 115              | 119            | 73           |
| 116              | 120            | 74           | 117              | 121            | 75           |
| 118              | 122            | 76           | 119              | 123            | 77           |
| 120              | 124            | 78           | 121              | 125            | 79           |
| 122              | 126            | 7A           | 123              | 127            | 7B           |
| 124              | 128            | 7C           | 125              | 129            | 7D           |
| 126              | 130            | 7E           | 127              | 131            | 7F           |
| 128              | 132            | 80           | 129              | 133            | 81           |
| 130              | 134            | 82           | 131              | 135            | 83           |
| 132              | 136            | 84           | 133              | 137            | 85           |
| 134              | 138            | 86           | 135              | 139            | 87           |
| 136              | 140            | 88           | 137              | 141            | 89           |
| 138              | 142            | 8A           | 139              | 143            | 8B           |
| 140              | 144            | 8C           | 141              | 145            | 8D           |
| 142              | 146            | 8E           | 143              | 147            | 8F           |
| 144              | 148            | 90           | 145              | 149            | 91           |
| 146              | 150            | 92           | 147              | 151            | 93           |
| 148              | 152            | 94           | 149              | 153            | 95           |
| 150              | 154            | 96           | 151              | 155            | 97           |
| 152              | 156            | 98           | 153              | 157            | 99           |
| 154              | 158            | 9A           | 155              | 159            | 9B           |
| 156              | 160            | 9C           | 157              | 161            | 9D           |
| 158              | 162            | 9E           | 159              | 163            | 9F           |
| 160              | 164            | 100          | 161              | 165            | 101          |
| 162              | 166            | 102          | 163              | 167            | 103          |
| 164              | 168            | 104          | 165              | 169            | 105          |
| 166              | 170            | 106          | 167              | 171            | 107          |
| 168              | 172            | 108          | 169              | 173            | 109          |
| 170              | 174            | 10A          | 171              | 175            | 10B          |
| 172              | 176            | 10C          | 173              | 177            | 10D          |
| 174              | 178            | 10E          | 175              | 179            | 10F          |
| 176              | 180            | 110          | 177              | 181            | 111          |
| 178              | 182            | 112          | 179              | 183            | 113          |
| 180              | 184            | 114          | 181              | 185            | 115          |
| 182              | 186            | 116          | 183              | 187            | 117          |
| 184              | 188            | 118          | 185              | 189            | 119          |
| 186              | 190            | 11A          | 187              | 191            | 11B          |
| 188              | 192            | 11C          | 189              | 193            | 11D          |
| 190              | 194            | 11E          | 191              | 195            | 11F          |
| 192              | 196            | 120          | 193              | 197            | 121          |
| 194              | 198            | 122          | 195              | 199            | 123          |
| 196              | 200            | 124          | 197              | 201            | 125          |
| 198              | 202            | 126          | 199              | 203            | 127          |
| 200              | 204            | 128          | 201              | 205            | 129          |
| 202              | 206            | 12A          | 203              | 207            | 12B          |
| 204              | 208            | 12C          | 205              | 209            | 12D          |
| 206              | 210            | 12E          | 207              | 211            | 12F          |
| 208              | 212            | 130          | 209              | 213            | 131          |
| 210              | 214            | 132          | 211              | 215            | 133          |
| 212              | 216            | 134          | 213              | 217            | 135          |
| 214              | 218            | 136          | 215              | 219            | 137          |
| 216              | 220            | 138          | 217              | 221            | 139          |
| 218              | 222            | 13A          | 219              | 223            | 13B          |
| 220              | 224            | 13C          | 221              | 225            | 13D          |
| 222              | 226            | 13E          | 223              | 227            | 13F          |
| 224              | 228            | 140          | 225              | 229            | 141          |
| 226              | 230            | 142          | 227              | 231            | 143          |
| 228              | 232            | 144          | 229              | 233            | 145          |
| 230              | 234            | 146          | 231              | 235            | 147          |
| 232              | 236            | 148          | 233              | 237            | 149          |
| 234              | 238            | 14A          | 235              | 239            | 14B          |
| 236              | 240            | 14C          | 237              | 241            | 14D          |
| 238              | 242            | 14E          | 239              | 243            | 14F          |
| 240              | 244            | 150          | 241              | 245            | 151          |
| 242              | 246            | 152          | 243              | 247            | 153          |
| 244              | 248            | 154          | 245              | 249            | 155          |
| 246              | 250            | 156          | 247              | 251            | 157          |
| 248              | 252            | 158          | 249              | 253            | 159          |
| 250              | 254            | 15A          | 251              | 255            | 15B          |
| 252              | 256            | 15C          | 253              | 257            | 15D          |
| 254              | 258            | 15E          | 255              | 259            | 15F          |
| 256              | 260            | 160          | 257              | 261            | 161          |
| 258              | 262            | 162          | 259              | 263            | 163          |
| 260              | 264            | 164          | 261              | 265            | 165          |
| 262              | 266            | 166          | 263              | 267            | 167          |
| 264              | 268            | 168          | 265              | 269            | 169          |
| 266              | 270            | 16A          | 267              | 271            | 16B          |
| 268              | 272            | 16C          | 269              | 273            | 16D          |
| 270              | 274            | 16E          | 271              | 275            | 16F          |
| 272              | 276            | 170          | 273              | 277            | 171          |
| 274              | 278            | 172          | 275              | 279            | 173          |
| 276              | 280            | 174          | 277              | 281            | 175          |
| 278              | 282            | 176          | 279              | 283            | 177          |
| 280              | 284            | 178          | 281              | 285            | 179          |
| 282              | 286            | 17A          | 283              | 287            | 17B          |
| 284              | 288            | 17C          | 285              | 289            | 17D          |
| 286              | 290            | 17E          | 287              | 291            | 17F          |
| 288              | 292            | 180          | 289              | 293            | 181          |
| 290              | 294            | 182          | 291              | 295            | 183          |
| 292              | 296            | 184          | 293              | 297            | 185          |
| 294              | 298            | 186          | 295              | 299            | 187          |
| 296              | 300            | 188          | 297              | 301            | 189          |
| 298              | 302            | 18A          | 299              | 303            | 18B          |
| 300              | 304            | 18C          | 301              | 305            | 18D          |
| 302              | 306            | 18E          | 303              | 307            | 18F          |
| 304              | 308            | 190          | 305              | 309            | 191          |
| 306              | 310            | 192          | 307              | 311            | 193          |
| 308              | 312            | 194          | 309              | 313            | 195          |
| 310              | 314            | 196          | 311              | 315            | 197          |
| 312              | 316            | 198          | 313              | 317            | 199          |
| 314              | 318            | 19A          | 315              | 319            | 19B          |
| 316              | 320            | 19C          | 317              | 321            | 19D          |
| 318              | 322            | 19E          | 319              | 323            | 19F          |
| 320              | 324            | 200          | 321              | 325            | 201          |
| 322              | 326            | 202          | 323              | 327            | 203          |
| 324              | 328            | 204          | 325              | 329            | 205          |
| 326              | 330            | 206          | 327              | 331            | 207          |
| 328              | 332            | 208          | 329              | 333            | 209          |
| 330              | 334            | 20A          | 331              | 335            | 20B          |
| 332              | 336            | 20C          | 333              | 337            | 20D          |
| 334              | 338            | 20E          | 335              | 339            | 20F          |
| 336              | 340            | 210          | 337              | 341            | 211          |
| 338              | 342            | 212          | 339              | 343            | 213          |
| 340              | 344            | 214          | 341              | 345            | 215          |
| 342              | 346            | 216          | 343              | 347            | 217          |
| 344              | 348            | 218          | 345              | 349            | 219          |
| 346              | 350            | 21A          | 347              | 351            | 21B          |
| 348              | 352            | 21C          | 349              | 353            | 21D          |
| 350              | 354            | 21E          | 351              | 355            | 21F          |
| 352              | 356            | 220          | 353              | 357            | 221          |
| 354              | 358            | 222          | 355              | 359            | 223          |
| 356              | 360            | 224          | 357              | 361            | 225          |
| 358              | 362            | 226          | 359              | 363            | 227          |
| 360              | 364            | 228          | 361              | 365            | 229          |
| 362              | 366            | 22A          | 363              | 367            | 22B          |
| 364              | 368            | 22C          | 365              | 369            | 22D          |
| 366              | 370            | 22E          | 367              | 371            | 22F          |
| 368              | 372            | 230          | 369              | 373            | 231          |
| 370              | 374            | 232          | 371              | 375            | 233          |
| 372              | 376            | 234          | 373              | 377            | 235          |
| 374              | 378            | 236          | 375              | 379            | 237          |
| 376              | 380            | 238          | 377              | 381            | 239          |
| 378              | 382            | 23A          | 379              | 383            | 23B          |
| 380              | 384            | 23C          | 381              | 385            | 23D          |
| 382              | 386            | 23E          | 383              | 387            | 23F          |
| 384              | 388            | 240          | 385              | 389            | 241          |
| 386              | 390            | 242          | 387              | 391            | 243          |
| 388              | 392            | 244          | 389              | 393            | 245          |
| 390              | 394            | 246          | 391              | 395            | 247          |
| 392              | 396            | 248          | 393              | 397            | 249          |
| 394              | 398            | 24A          | 395              | 399            | 24B          |
| 396              | 400            | 24C          | 397              | 401            | 24D          |
| 398              | 402            | 24E          | 399              | 403            | 24F          |
| 400              | 404            | 250          | 401              | 405            | 251          |
| 402              | 406            | 252          | 403              | 407            | 253          |
| 404              | 408            | 254          | 405              | 409            | 255          |
| 406              | 410            | 256          | 407              | 411            | 257          |
| 408              | 412            | 258          | 409              | 413            | 259          |
| 410              | 414            | 25A          | 411              | 415            | 25B          |
| 412              | 416            | 25C          | 413              | 417            | 25D          |
| 414              | 418            | 25E          | 415              | 419            | 25F          |
| 416              | 420            | 260          | 417              | 421            | 261          |
| 418              | 422            | 262          | 419              | 423            | 263          |
| 420              | 424            | 264          | 421              | 425            | 265          |
| 422              | 426            | 266          | 423              | 427            | 267          |
| 424              | 428            | 268          | 425              | 429            | 269          |
| 426              | 430            | 26A          | 427              | 431            | 26B          |
| 428              | 432            | 26C          | 429              | 433            | 26D          |
| 430              | 434            | 26E          | 431              | 435            | 26F          |
| 432              | 436            | 270          | 433              | 437            | 271          |
| 434              | 438            | 272          | 435              | 439            | 273          |
| 436              | 440            | 274          | 437              | 441            | 275          |
| 438              | 442            | 276          | 439              | 443            | 277          |
| 440              | 444            | 278          | 441              | 445            | 279          |
| 442              | 446            | 27A          | 443              | 447            | 27B          |
| 444              | 448            | 27C          | 445              | 449            | 27D          |
| 446              | 450            | 27E          | 447              | 451            | 27F          |
| 448              | 452            | 280          | 449              | 453            | 281          |
| 450              | 454            | 282          | 451              | 455            | 283          |
| 452              | 456            | 284          | 453              | 457            | 285          |
| 454              | 458            | 286          | 455              | 459            | 287          |
| 456              | 460            | 288          | 457              | 461            | 289          |
| 458              | 462            | 28A          | 459              | 463            | 28B          |
| 460              | 464            | 28C          | 461              | 465            | 28D          |
| 462              | 466            | 28E          | 463              | 467            | 28F          |
| 464              | 468            | 290          | 465              | 469            | 291          |
| 466              | 470            | 292          | 467              | 471            | 293          |
| 468              | 472            | 294          | 469              | 473            | 295          |
| 470              | 474            | 296          | 471              | 475            | 297          |
| 472              | 476            | 298          | 473              | 477            | 299          |
|                  |                |              |                  |                |              |

## Converting from Character to Code:

*(There is an ASCII table on the back of today's lecture slip.)*

| ASCII TABLE |      | Decimal |      | Hex Char |          | Octal Char |      | Binary |          | Decimal |      | Hex Char |          | Octal Char |      | Binary |              |
|-------------|------|---------|------|----------|----------|------------|------|--------|----------|---------|------|----------|----------|------------|------|--------|--------------|
| Decimal     | Char | Dec     | Hex  | Oct      | Bin      | Dec        | Hex  | Oct    | Bin      | Dec     | Hex  | Oct      | Bin      | Dec        | Hex  | Oct    | Bin          |
| 0           | \0   | 0       | 0000 | 0        | 00000000 | 0          | 0000 | 0      | 00000000 | 0       | 0000 | 0        | 00000000 | 0          | 0000 | 0      | 00000000     |
| 1           | \1   | 1       | 0001 | 1        | 00000001 | 1          | 0001 | 1      | 00000001 | 1       | 0001 | 1        | 00000001 | 1          | 0001 | 1      | 00000001     |
| 2           | \2   | 2       | 0010 | 2        | 00000010 | 2          | 0010 | 2      | 00000010 | 2       | 0010 | 2        | 00000010 | 2          | 0010 | 2      | 00000010     |
| 3           | \3   | 3       | 0011 | 3        | 00000011 | 3          | 0011 | 3      | 00000011 | 3       | 0011 | 3        | 00000011 | 3          | 0011 | 3      | 00000011     |
| 4           | \4   | 4       | 0100 | 4        | 00000100 | 4          | 0100 | 4      | 00000100 | 4       | 0100 | 4        | 00000100 | 4          | 0100 | 4      | 00000100     |
| 5           | \5   | 5       | 0101 | 5        | 00000101 | 5          | 0101 | 5      | 00000101 | 5       | 0101 | 5        | 00000101 | 5          | 0101 | 5      | 00000101     |
| 6           | \6   | 6       | 0110 | 6        | 00000110 | 6          | 0110 | 6      | 00000110 | 6       | 0110 | 6        | 00000110 | 6          | 0110 | 6      | 00000110     |
| 7           | \7   | 7       | 0111 | 7        | 00000111 | 7          | 0111 | 7      | 00000111 | 7       | 0111 | 7        | 00000111 | 7          | 0111 | 7      | 00000111     |
| 8           | \8   | 8       | 1000 | 10       | 00001000 | 10         | 1000 | 10     | 00001000 | 10      | 1000 | 10       | 00001000 | 10         | 1000 | 10     | 00001000     |
| 9           | \9   | 9       | 1001 | 11       | 00001001 | 11         | 1001 | 11     | 00001001 | 11      | 1001 | 11       | 00001001 | 11         | 1001 | 11     | 00001001     |
| 10          | \a   | 10      | 1010 | 12       | 00001010 | 12         | 1010 | 12     | 00001010 | 12      | 1010 | 12       | 00001010 | 12         | 1010 | 12     | 00001010     |
| 11          | \b   | 11      | 1011 | 13       | 00001011 | 13         | 1011 | 13     | 00001011 | 13      | 1011 | 13       | 00001011 | 13         | 1011 | 13     | 00001011     |
| 12          | \c   | 12      | 1100 | 14       | 00001100 | 14         | 1100 | 14     | 00001100 | 14      | 1100 | 14       | 00001100 | 14         | 1100 | 14     | 00001100     |
| 13          | \d   | 13      | 1101 | 15       | 00001101 | 15         | 1101 | 15     | 00001101 | 15      | 1101 | 15       | 00001101 | 15         | 1101 | 15     | 00001101     |
| 14          | \e   | 14      | 1110 | 16       | 00001110 | 16         | 1110 | 16     | 00001110 | 16      | 1110 | 16       | 00001110 | 16         | 1110 | 16     | 00001110     |
| 15          | \f   | 15      | 1111 | 17       | 00001111 | 17         | 1111 | 17     | 00001111 | 17      | 1111 | 17       | 00001111 | 17         | 1111 | 17     | 00001111     |
| 16          | \n   | 16      | 2000 | 20       | 00010000 | 20         | 2000 | 20     | 00010000 | 20      | 2000 | 20       | 00010000 | 20         | 2000 | 20     | 00010000     |
| 17          | \r   | 17      | 2001 | 21       | 00010001 | 21         | 2001 | 21     | 00010001 | 21      | 2001 | 21       | 00010001 | 21         | 2001 | 21     | 00010001     |
| 18          | \t   | 18      | 2010 | 22       | 00010010 | 22         | 2010 | 22     | 00010010 | 22      | 2010 | 22       | 00010010 | 22         | 2010 | 22     | 00010010     |
| 19          | \v   | 19      | 2011 | 23       | 00010011 | 23         | 2011 | 23     | 00010011 | 23      | 2011 | 23       | 00010011 | 23         | 2011 | 23     | 00010011     |
| 20          | \b   | 20      | 2100 | 24       | 00010100 | 24         | 2100 | 24     | 00010100 | 24      | 2100 | 24       | 00010100 | 24         | 2100 | 24     | 00010100     |
| 21          | \f   | 21      | 2101 | 25       | 00010101 | 25         | 2101 | 25     | 00010101 | 25      | 2101 | 25       | 00010101 | 25         | 2101 | 25     | 00010101     |
| 22          | \n   | 22      | 2110 | 26       | 00010110 | 26         | 2110 | 26     | 00010110 | 26      | 2110 | 26       | 00010110 | 26         | 2110 | 26     | 00010110     |
| 23          | \r   | 23      | 2111 | 27       | 00010111 | 27         | 2111 | 27     | 00010111 | 27      | 2111 | 27       | 00010111 | 27         | 2111 | 27     | 00010111     |
| 24          | \012 | 24      | 2200 | 28       | 00011000 | 28         | 2200 | 28     | 00011000 | 28      | 2200 | 28       | 00011000 | 28         | 2200 | 28     | 00011000     |
| 25          | \015 | 25      | 2201 | 29       | 00011001 | 29         | 2201 | 29     | 00011001 | 29      | 2201 | 29       | 00011001 | 29         | 2201 | 29     | 00011001     |
| 26          | \016 | 26      | 2210 | 30       | 00011010 | 30         | 2210 | 30     | 00011010 | 30      | 2210 | 30       | 00011010 | 30         | 2210 | 30     | 00011010     |
| 27          | \017 | 27      | 2211 | 31       | 00011011 | 31         | 2211 | 31     | 00011011 | 31      | 2211 | 31       | 00011011 | 31         | 2211 | 31     | 00011011     |
| 28          | \020 | 28      | 2220 | 32       | 00011100 | 32         | 2220 | 32     | 00011100 | 32      | 2220 | 32       | 00011100 | 32         | 2220 | 32     | 00011100     |
| 29          | \021 | 29      | 2221 | 33       | 00011101 | 33         | 2221 | 33     | 00011101 | 33      | 2221 | 33       | 00011101 | 33         | 2221 | 33     | 00011101     |
| 30          | \022 | 30      | 2222 | 34       | 00011110 | 34         | 2222 | 34     | 00011110 | 34      | 2222 | 34       | 00011110 | 34         | 2222 | 34     | 00011110     |
| 31          | \023 | 31      | 2223 | 35       | 00011111 | 35         | 2223 | 35     | 00011111 | 35      | 2223 | 35       | 00011111 | 35         | 2223 | 35     | 00011111     |
| 32          | \024 | 32      | 2224 | 36       | 00011111 | 36         | 2224 | 36     | 00011111 | 36      | 2224 | 36       | 00011111 | 36         | 2224 | 36     | 00011111     |
| 33          | \025 | 33      | 2225 | 37       | 00011111 | 37         | 2225 | 37     | 00011111 | 37      | 2225 | 37       | 00011111 | 37         | 2225 | 37     | 00011111     |
| 34          | \026 | 34      | 2226 | 38       | 00011111 | 38         | 2226 | 38     | 00011111 | 38      | 2226 | 38       | 00011111 | 38         | 2226 | 38     | 00011111     |
| 35          | \027 | 35      | 2227 | 39       | 00011111 | 39         | 2227 | 39     | 00011111 | 39      | 2227 | 39       | 00011111 | 39         | 2227 | 39     | 00011111     |
| 36          | \030 | 36      | 2228 | 40       | 00011111 | 40         | 2228 | 40     | 00011111 | 40      | 2228 | 40       | 00011111 | 40         | 2228 | 40     | 00011111     |
| 37          | \031 | 37      | 2229 | 41       | 00011111 | 41         | 2229 | 41     | 00011111 | 41      | 2229 | 41       | 00011111 | 41         | 2229 | 41     | 00011111     |
| 38          | \032 | 38      | 2230 | 42       | 00011111 | 42         | 2230 | 42     | 00011111 | 42      | 2230 | 42       | 00011111 | 42         | 2230 | 42     | 00011111     |
| 39          | \033 | 39      | 2231 | 43       | 00011111 | 43         | 2231 | 43     | 00011111 | 43      | 2231 | 43       | 00011111 | 43         | 2231 | 43     | 00011111     |
| 40          | \034 | 40      | 2232 | 44       | 00011111 | 44         | 2232 | 44     | 00011111 | 44      | 2232 | 44       | 00011111 | 44         | 2232 | 44     | 00011111     |
| 41          | \035 | 41      | 2233 | 45       | 00011111 | 45         | 2233 | 45     | 00011111 | 45      | 2233 | 45       | 00011111 | 45         | 2233 | 45     | 00011111     |
| 42          | \036 | 42      | 2234 | 46       | 00011111 | 46         | 2234 | 46     | 00011111 | 46      | 2234 | 46       | 00011111 | 46         | 2234 | 46     | 00011111     |
| 43          | \037 | 43      | 2235 | 47       | 00011111 | 47         | 2235 | 47     | 00011111 | 47      | 2235 | 47       | 00011111 | 47         | 2235 | 47     | 00011111     |
| 44          | \040 | 44      | 2236 | 48       | 00011111 | 48         | 2236 | 48     | 00011111 | 48      | 2236 | 48       | 00011111 | 48         | 2236 | 48     | 00011111     |
| 45          | \041 | 45      | 2237 | 49       | 00011111 | 49         | 2237 | 49     | 00011111 | 49      | 2237 | 49       | 00011111 | 49         | 2237 | 49     | 00011111     |
| 46          | \042 | 46      | 2238 | 50       | 00011111 | 50         | 2238 | 50     | 00011111 | 50      | 2238 | 50       | 00011111 | 50         | 2238 | 50     | 00011111     |
| 47          | \043 | 47      | 2239 | 51       | 00011111 | 51         | 2239 | 51     | 00011111 | 51      | 2239 | 51       | 00011111 | 51         | 2239 | 51     | 00011111     |
| 48          | \044 | 48      | 2240 | 52       | 00011111 | 52         | 2240 | 52     | 00011111 | 52      | 2240 | 52       | 00011111 | 52         | 2240 | 52     | 00011111     |
| 49          | \045 | 49      | 2241 | 53       | 00011111 | 53         | 2241 | 53     | 00011111 | 53      | 2241 | 53       | 00011111 | 53         | 2241 | 53     | 00011111     |
| 50          | \046 | 50      | 2242 | 54       | 00011111 | 54         | 2242 | 54     | 00011111 | 54      | 2242 | 54       | 00011111 | 54         | 2242 | 54     | 00011111     |
| 51          | \047 | 51      | 2243 | 55       | 00011111 | 55         | 2243 | 55     | 00011111 | 55      | 2243 | 55       | 00011111 | 55         | 2243 | 55     | 00011111     |
| 52          | \050 | 52      | 2244 | 56       | 00011111 | 56         | 2244 | 56     | 00011111 | 56      | 2244 | 56       | 00011111 | 56         | 2244 | 56     | 00011111     |
| 53          | \051 | 53      | 2245 | 57       | 00011111 | 57         | 2245 | 57     | 00011111 | 57      | 2245 | 57       | 00011111 | 57         | 2245 | 57     | 00011111     |
| 54          | \052 | 54      | 2246 | 58       | 00011111 | 58         | 2246 | 58     | 00011111 | 58      | 2246 | 58       | 00011111 | 58         | 2246 | 58     | 00011111     |
| 55          | \053 | 55      | 2247 | 59       | 00011111 | 59         | 2247 | 59     | 00011111 | 59      | 2247 | 59       | 00011111 | 59         | 2247 | 59     | 00011111     |
| 56          | \054 | 56      | 2248 | 60       | 00011111 | 60         | 2248 | 60     | 00011111 | 60      | 2248 | 60       | 00011111 | 60         | 2248 | 60     | 00011111     |
| 57          | \055 | 57      | 2249 | 61       | 00011111 | 61         | 2249 | 61     | 00011111 | 61      | 2249 | 61       | 00011111 | 61         | 2249 | 61     | 00011111     |
| 58          | \056 | 58      | 2250 | 62       | 00011111 | 62         | 2250 | 62     | 00011111 | 62      | 2250 | 62       | 00011111 | 62         | 2250 | 62     | 00011111     |
| 59          | \057 | 59      | 2251 | 63       | 00011111 | 63         | 2251 | 63     | 00011111 | 63      | 2251 | 63       | 00011111 | 63         | 2251 | 63     | 00011111     |
| 60          | \060 | 60      | 2252 | 64       | 00011111 | 64         | 2252 | 64     | 00011111 | 64      | 2252 | 64       | 00011111 | 64         | 2252 | 64     | 00011111     |
| 61          | \061 | 61      | 2253 | 65       | 00011111 | 65         | 2253 | 65     | 00011111 | 65      | 2253 | 65       | 00011111 | 65         | 2253 | 65     | 00011111     |
| 62          | \062 | 62      | 2254 | 66       | 00011111 | 66         | 2254 | 66     | 00011111 | 66      | 2254 | 66       | 00011111 | 66         | 2254 | 66     | 00011111     |
| 63          | \063 | 63      | 2255 | 67       | 00011111 | 67         | 2255 | 67     | 00011111 | 67      | 2255 | 67       | 00011111 | 67         | 2255 | 67     | 00011111     |
| 64          | \064 | 64      | 2256 | 68       | 00011111 | 68         | 2256 | 68     | 00011111 | 68      | 2256 | 68       | 00011111 | 68         | 2256 | 68     | 00011111     |
| 65          | \065 | 65      | 2257 | 69       | 00011111 | 69         | 2257 | 69     | 00011111 | 69      | 2257 | 69       | 00011111 | 69         | 2257 | 69     | 00011111     |
| 66          | \066 | 66      | 2258 | 70       | 00011111 | 70         | 2258 | 70     | 00011111 | 70      | 2258 | 70       | 00011111 | 70         | 2258 | 70     | 00011111     |
| 67          | \067 | 67      | 2259 | 71       | 00011111 | 71         | 2259 | 71     | 00011111 | 71      | 2259 | 71       | 00011111 | 71         | 2259 | 71     | 00011111     |
| 68          | \068 | 68      | 2260 | 72       | 00011111 | 72         | 2260 | 72     | 00011111 | 72      | 2260 | 72       | 00011111 | 72         | 2260 | 72     | 00011111     |
| 69          | \069 | 69      | 2261 | 73       | 00011111 | 73         | 2261 | 73     | 00011111 | 73      | 2261 | 73       | 00011111 | 73         | 2261 | 73     | 00011111     |
| 70          | \070 | 70      | 2262 | 74       | 00011111 | 74         | 2262 | 74     | 00011111 | 74      | 2262 | 74       | 00011111 | 74         | 2262 | 74     | 00011111     |
| 71          | \071 | 71      | 2263 | 75       | 00011111 | 75         | 2263 | 75     | 00011111 | 75      | 2263 | 75       | 00011111 | 75         | 2263 | 75     | 00011111</td |

- `ord(c)`: returns Unicode (ASCII) of the character.

## Converting from Character to Code:

*(There is an ASCII table on the back of today's lecture slip.)*

| ASCII TABLE |     |       |      |      |         |     |       |         |      |         |     |       |         |      |
|-------------|-----|-------|------|------|---------|-----|-------|---------|------|---------|-----|-------|---------|------|
| Decimal     | Hex | Octal | Name | Char | Decimal | Hex | Octal | Name    | Char | Decimal | Hex | Octal | Name    | Char |
| 0           | 00  | 0     | NULL | \0   | 32      | 20  | 40    | SIGKILL | \000 | 64      | 40  | 100   | SIGPOLL | \001 |
| 1           | 01  | 1     | SOH  | \001 | 33      | 21  | 41    | SIGALRM | \002 | 65      | 41  | 101   | SIGSTOP | \003 |
| 2           | 02  | 2     | STX  | \002 | 34      | 22  | 42    | SIGPOLL | \004 | 66      | 42  | 102   | SIGCONT | \005 |
| 3           | 03  | 3     | ETX  | \003 | 35      | 23  | 43    | SIGPOLL | \006 | 67      | 43  | 103   | SIGKILL | \007 |
| 4           | 04  | 4     | EOT  | \004 | 36      | 24  | 44    | SIGPOLL | \008 | 68      | 44  | 104   | SIGPOLL | \009 |
| 5           | 05  | 5     | ENQ  | \005 | 37      | 25  | 45    | SIGPOLL | \010 | 69      | 45  | 105   | SIGPOLL | \011 |
| 6           | 06  | 6     | ACK  | \006 | 38      | 26  | 46    | SIGPOLL | \012 | 70      | 46  | 106   | SIGPOLL | \013 |
| 7           | 07  | 7     | NAK  | \007 | 39      | 27  | 47    | SIGPOLL | \014 | 71      | 47  | 107   | SIGPOLL | \015 |
| 8           | 08  | 10    | SYN  | \008 | 40      | 28  | 48    | SIGPOLL | \016 | 72      | 48  | 108   | SIGPOLL | \017 |
| 9           | 09  | 11    | BT   | \009 | 41      | 29  | 49    | SIGPOLL | \018 | 73      | 49  | 109   | SIGPOLL | \019 |
| 10          | 0A  | 12    | HT   | \00A | 42      | 2A  | 4A    | SIGPOLL | \01A | 74      | 4A  | 110   | SIGPOLL | \01B |
| 11          | 0B  | 13    | LF   | \00B | 43      | 2B  | 4B    | SIGPOLL | \01B | 75      | 4B  | 111   | SIGPOLL | \01C |
| 12          | 0C  | 14    | VT   | \00C | 44      | 2C  | 4C    | SIGPOLL | \01C | 76      | 4C  | 112   | SIGPOLL | \01D |
| 13          | 0D  | 15    | FF   | \00D | 45      | 2D  | 4D    | SIGPOLL | \01D | 77      | 4D  | 113   | SIGPOLL | \01E |
| 14          | 0E  | 16    | CR   | \00E | 46      | 2E  | 4E    | SIGPOLL | \01E | 78      | 4E  | 114   | SIGPOLL | \01F |
| 15          | 0F  | 17    | SO   | \00F | 47      | 2F  | 4F    | SIGPOLL | \01F | 79      | 4F  | 115   | SIGPOLL | \020 |
| 16          | 10  | 20    | SI   | \010 | 48      | 30  | 50    | SIGPOLL | \01F | 80      | 50  | 116   | SIGPOLL | \021 |
| 17          | 11  | 21    | DC1  | \011 | 49      | 31  | 51    | SIGPOLL | \01F | 81      | 51  | 117   | SIGPOLL | \022 |
| 18          | 12  | 22    | DC2  | \012 | 50      | 32  | 52    | SIGPOLL | \01F | 82      | 52  | 118   | SIGPOLL | \023 |
| 19          | 13  | 23    | DC3  | \013 | 51      | 33  | 53    | SIGPOLL | \01F | 83      | 53  | 119   | SIGPOLL | \024 |
| 20          | 14  | 24    | DC4  | \014 | 52      | 34  | 54    | SIGPOLL | \01F | 84      | 54  | 120   | SIGPOLL | \025 |
| 21          | 15  | 25    | NAK  | \015 | 53      | 35  | 55    | SIGPOLL | \01F | 85      | 55  | 121   | SIGPOLL | \026 |
| 22          | 16  | 26    | SYN  | \016 | 54      | 36  | 56    | SIGPOLL | \01F | 86      | 56  | 122   | SIGPOLL | \027 |
| 23          | 17  | 27    | ETX  | \017 | 55      | 37  | 57    | SIGPOLL | \01F | 87      | 57  | 123   | SIGPOLL | \028 |
| 24          | 18  | 28    | ENQ  | \018 | 56      | 38  | 58    | SIGPOLL | \01F | 88      | 58  | 124   | SIGPOLL | \029 |
| 25          | 19  | 29    | ACK  | \019 | 57      | 39  | 59    | SIGPOLL | \01F | 89      | 59  | 125   | SIGPOLL | \02A |
| 26          | 1A  | 2A    | BT   | \01A | 58      | 3A  | 5A    | SIGPOLL | \01F | 90      | 5A  | 126   | SIGPOLL | \02B |
| 27          | 1B  | 2B    | HT   | \01B | 59      | 3B  | 5B    | SIGPOLL | \01F | 91      | 5B  | 127   | SIGPOLL | \02C |
| 28          | 1C  | 2C    | LF   | \01C | 60      | 3C  | 5C    | SIGPOLL | \01F | 92      | 5C  | 128   | SIGPOLL | \02D |
| 29          | 1D  | 2D    | VT   | \01D | 61      | 3D  | 5D    | SIGPOLL | \01F | 93      | 5D  | 129   | SIGPOLL | \02E |
| 30          | 1E  | 2E    | FF   | \01E | 62      | 3E  | 5E    | SIGPOLL | \01F | 94      | 5E  | 130   | SIGPOLL | \02F |
| 31          | 1F  | 2F    | CR   | \01F | 63      | 3F  | 5F    | SIGPOLL | \01F | 95      | 5F  | 131   | SIGPOLL | \030 |
| 32          | 20  | 30    | SO   | \020 | 64      | 40  | 60    | SIGPOLL | \01F | 96      | 60  | 132   | SIGPOLL | \031 |
| 33          | 21  | 31    | SI   | \021 | 65      | 41  | 61    | SIGPOLL | \01F | 97      | 61  | 133   | SIGPOLL | \032 |
| 34          | 22  | 32    | DC1  | \022 | 66      | 42  | 62    | SIGPOLL | \01F | 98      | 62  | 134   | SIGPOLL | \033 |
| 35          | 23  | 33    | DC2  | \023 | 67      | 43  | 63    | SIGPOLL | \01F | 99      | 63  | 135   | SIGPOLL | \034 |
| 36          | 24  | 34    | DC3  | \024 | 68      | 44  | 64    | SIGPOLL | \01F | 100     | 64  | 136   | SIGPOLL | \035 |
| 37          | 25  | 35    | DC4  | \025 | 69      | 45  | 65    | SIGPOLL | \01F | 101     | 65  | 137   | SIGPOLL | \036 |
| 38          | 26  | 36    | NAK  | \026 | 70      | 46  | 66    | SIGPOLL | \01F | 102     | 66  | 138   | SIGPOLL | \037 |
| 39          | 27  | 37    | SYN  | \027 | 71      | 47  | 67    | SIGPOLL | \01F | 103     | 67  | 139   | SIGPOLL | \038 |
| 40          | 28  | 38    | ETX  | \028 | 72      | 48  | 68    | SIGPOLL | \01F | 104     | 68  | 140   | SIGPOLL | \039 |
| 41          | 29  | 39    | ENQ  | \029 | 73      | 49  | 69    | SIGPOLL | \01F | 105     | 69  | 141   | SIGPOLL | \03A |
| 42          | 2A  | 3A    | ACK  | \02A | 74      | 4A  | 6A    | SIGPOLL | \01F | 106     | 6A  | 142   | SIGPOLL | \03B |
| 43          | 2B  | 3B    | BT   | \02B | 75      | 4B  | 6B    | SIGPOLL | \01F | 107     | 6B  | 143   | SIGPOLL | \03C |
| 44          | 2C  | 3C    | HT   | \02C | 76      | 4C  | 6C    | SIGPOLL | \01F | 108     | 6C  | 144   | SIGPOLL | \03D |
| 45          | 2D  | 3D    | LF   | \02D | 77      | 4D  | 6D    | SIGPOLL | \01F | 109     | 6D  | 145   | SIGPOLL | \03E |
| 46          | 2E  | 3E    | VT   | \02E | 78      | 4E  | 6E    | SIGPOLL | \01F | 110     | 6E  | 146   | SIGPOLL | \03F |
| 47          | 2F  | 3F    | FF   | \02F | 79      | 4F  | 6F    | SIGPOLL | \01F | 111     | 6F  | 147   | SIGPOLL | \040 |
| 48          | 30  | 40    | CR   | \030 | 80      | 50  | 70    | SIGPOLL | \01F | 112     | 70  | 148   | SIGPOLL | \041 |
| 49          | 31  | 41    | SO   | \031 | 81      | 51  | 71    | SIGPOLL | \01F | 113     | 71  | 149   | SIGPOLL | \042 |
| 50          | 32  | 42    | SI   | \032 | 82      | 52  | 72    | SIGPOLL | \01F | 114     | 72  | 150   | SIGPOLL | \043 |
| 51          | 33  | 43    | DC1  | \033 | 83      | 53  | 73    | SIGPOLL | \01F | 115     | 73  | 151   | SIGPOLL | \044 |
| 52          | 34  | 44    | DC2  | \034 | 84      | 54  | 74    | SIGPOLL | \01F | 116     | 74  | 152   | SIGPOLL | \045 |
| 53          | 35  | 45    | DC3  | \035 | 85      | 55  | 75    | SIGPOLL | \01F | 117     | 75  | 153   | SIGPOLL | \046 |
| 54          | 36  | 46    | DC4  | \036 | 86      | 56  | 76    | SIGPOLL | \01F | 118     | 76  | 154   | SIGPOLL | \047 |
| 55          | 37  | 47    | NAK  | \037 | 87      | 57  | 77    | SIGPOLL | \01F | 119     | 77  | 155   | SIGPOLL | \048 |
| 56          | 38  | 48    | SYN  | \038 | 88      | 58  | 78    | SIGPOLL | \01F | 120     | 78  | 156   | SIGPOLL | \049 |
| 57          | 39  | 49    | ETX  | \039 | 89      | 59  | 79    | SIGPOLL | \01F | 121     | 79  | 157   | SIGPOLL | \04A |
| 58          | 3A  | 4A    | ENQ  | \03A | 90      | 5A  | 7A    | SIGPOLL | \01F | 122     | 7A  | 158   | SIGPOLL | \04B |
| 59          | 3B  | 4B    | ACK  | \03B | 91      | 5B  | 7B    | SIGPOLL | \01F | 123     | 7B  | 159   | SIGPOLL | \04C |
| 60          | 3C  | 4C    | BT   | \03C | 92      | 5C  | 7C    | SIGPOLL | \01F | 124     | 7C  | 160   | SIGPOLL | \04D |
| 61          | 3D  | 4D    | HT   | \03D | 93      | 5D  | 7D    | SIGPOLL | \01F | 125     | 7D  | 161   | SIGPOLL | \04E |
| 62          | 3E  | 4E    | LF   | \03E | 94      | 5E  | 7E    | SIGPOLL | \01F | 126     | 7E  | 162   | SIGPOLL | \04F |
| 63          | 3F  | 4F    | VT   | \03F | 95      | 5F  | 7F    | SIGPOLL | \01F | 127     | 7F  | 163   | SIGPOLL | \050 |
| 64          | 40  | 50    | FF   | \040 | 96      | 60  | 80    | SIGPOLL | \01F | 128     | 80  | 164   | SIGPOLL | \051 |
| 65          | 41  | 51    | CR   | \041 | 97      | 61  | 81    | SIGPOLL | \01F | 129     | 81  | 165   | SIGPOLL | \052 |
| 66          | 42  | 52    | SO   | \042 | 98      | 62  | 82    | SIGPOLL | \01F | 130     | 82  | 166   | SIGPOLL | \053 |
| 67          | 43  | 53    | SI   | \043 | 99      | 63  | 83    | SIGPOLL | \01F | 131     | 83  | 167   | SIGPOLL | \054 |
| 68          | 44  | 54    | DC1  | \044 | 100     | 64  | 84    | SIGPOLL | \01F | 132     | 84  | 168   | SIGPOLL | \055 |
| 69          | 45  | 55    | DC2  | \045 | 101     | 65  | 85    | SIGPOLL | \01F | 133     | 85  | 169   | SIGPOLL | \056 |
| 70          | 46  | 56    | DC3  | \046 | 102     | 66  | 86    | SIGPOLL | \01F | 134     | 86  | 170   | SIGPOLL | \057 |
| 71          | 47  | 57    | DC4  | \047 | 103     | 67  | 87    | SIGPOLL | \01F | 135     | 87  | 171   | SIGPOLL | \058 |
| 72          | 48  | 58    | NAK  | \048 | 104     | 68  | 88    | SIGPOLL | \01F | 136     | 88  | 172   | SIGPOLL | \059 |
| 73          | 49  | 59    | SYN  | \049 | 105     | 69  | 89    | SIGPOLL | \01F | 137     | 89  | 173   | SIGPOLL | \05A |
| 74          | 4A  | 5A    | ETX  | \04A | 106     | 6A  | 8A    | SIGPOLL | \01F | 138     | 8A  | 174   | SIGPOLL | \05B |
| 75          | 4B  | 5B    | ENQ  | \04B | 107     | 6B  | 8B    | SIGPOLL | \01F | 139     | 8B  | 175   | SIGPOLL | \05C |
| 76          | 4C  | 5C    | ACK  | \04C | 108     | 6C  | 8C    | SIGPOLL | \01F | 140     | 8C  | 176   | SIGPOLL | \05D |
| 77          | 4D  | 5D    | BT   | \04D | 109     | 6D  | 8D    | SIGPOLL | \01F | 141     | 8D  | 177   | SIGPOLL | \05E |
| 78          | 4E  | 5E    | HT   | \04E | 110     | 6E  | 8E    | SIGPOLL | \01F | 142     | 8E  | 178   | SIGPOLL | \05F |
| 79          | 4F  | 5F    | LF   | \04F | 111     | 6F  | 8F    | SIGPOLL | \01F | 143     | 8F  | 179   | SIGPOLL | \060 |
| 80          | 50  | 60    | VT   | \050 | 112     | 70  | 90    | SIGPOLL | \01F | 144     | 90  | 180   | SIGPOLL | \061 |
| 81          | 51  | 61    | FF   | \051 | 113     | 71  | 91    | SIGPOLL | \01F | 145     | 91  | 181   | SIGPOLL | \062 |
| 82          | 52  | 62    | CR   | \052 | 114     | 72  | 92    | SIGPOLL | \01F | 146     | 92  | 182   | SIGPOLL | \063 |
| 83          | 53  | 63    | SO   | \053 | 115     | 73  | 93    | SIGPOLL | \01F | 147     | 93  | 183   | SIGPOLL | \064 |
| 84          | 54  | 64    | SI   | \054 | 116     | 74  | 94    | SIGPOLL | \01F | 148     | 94  | 184   | SIGPOLL | \065 |
| 85          | 55  | 65    | DC1  | \055 | 117     | 75  | 95    | SIGPOLL | \01F | 149     | 95  | 185   | SIGPOLL | \066 |
| 86          | 56  | 66    | DC2  | \056 | 118     | 76  | 96    | SIGPOLL | \01F | 150     | 96  | 186   | SIGPOLL | \067 |
| 87          | 57  | 67    | DC3  | \057 | 119     | 77  | 97    | SIGPOLL | \01F | 151     | 97  | 187   | SIGPOLL | \068 |
| 88          | 58  | 68    | DC4  | \058 | 120     | 78  | 98    | SIGPOLL | \01F | 152     | 98  | 188   | SIGPOLL | \069 |
| 89          | 59  | 69    | NAK  | \059 | 121     | 79  | 99    | SIGPOLL | \01F | 153     | 99  | 189   | SIGPOLL | \06A |
| 90          | 5A  | 6A    | SYN  | \05A | 122     | 7A  | 9A    | SIGPOLL | \01F | 154     | 9A  | 190   | SIGPOLL | \06B |
| 91          | 5B  | 6B    | ETX  | \05B | 123     | 7B  | 9B    | SIGPOLL | \01F | 155     | 9B  | 191   | SIGPOLL | \06C |
| 92          | 5C  | 6C    | ENQ  | \05C | 124     | 7C  | 9C    | SIGPOLL | \01F | 156     | 9C  | 192   | SIGPOLL | \06D |
| 93          | 5D  | 6D    | ACK  | \05D | 125     | 7D  | 9D    | SIGPOLL | \01F | 157     | 9D  | 193   | SIGPOLL | \06E |
| 94          | 5E  | 6E    | BT   | \05E | 126     | 7E  | 9E    | SIGPOLL | \01F | 158     | 9E  | 194   | SIGPOLL | \06F |
| 95          | 5F  | 6F    | HT   | \05F | 127     | 7F  | 9F    | SIGPOLL | \01F | 159     | 9F  | 195   | SIGPOLL | \070 |
| 96          | 60  | 70    | LF   | \060 | 128     | 80  | A0    | SIGPOLL | \01F | 160     | A0  | 196   | SIGPOLL | \071 |
| 97          | 61  | 71    | VT   | \061 | 129     | 81  | A1    | SIGPOLL | \01F | 161     | A1  | 197   | SIGPOLL | \072 |
| 98          | 62  | 72    | FF   | \062 | 130     | 82  | A2    | SIGPOLL | \01F | 162     | A2  | 198   | SIGPOLL | \073 |
| 99          | 63  | 73    | CR   | \063 | 131     | 83  | A3    | SIGPOLL | \01F | 163     | A3  | 199   | SIGPOLL | \074 |
| 100         | 64  | 74    | SO   | \064 | 132     | 84  | A4    | SIGPOLL | \01F | 164     | A4  | 200   | SIGPOLL | \075 |
| 101         | 65  | 75    | SI   | \065 | 133     | 85  | A5    | SIGPOLL | \01F | 165     | A5  | 201   | SIGPOLL | \076 |
| 102         | 66  | 76    | DC1  | \066 | 134     | 86  | A6    | SIGPOLL | \01F | 166     | A6  | 202   | SIGPOLL | \077 |
| 103         | 67  | 77    | DC2  | \067 | 135     | 87  | A7    | SIGPOLL |      |         |     |       |         |      |

- `ord(c)`: returns Unicode (ASCII) of the character.
  - Example: `ord('a')` returns 97.

# Converting from Character to Code:

*(There is an ASCII table on the back of today's lecture slip.)*

| Decimal Num Char | Octal Num Char | Hex Num Char | Decimal Num Char | Octal Num Char | Hex Num Char |
|------------------|----------------|--------------|------------------|----------------|--------------|
| '\000'           | '000'          | '000'        | '\001'           | '001'          | '001'        |
| '\002'           | '002'          | '002'        | '\003'           | '003'          | '003'        |
| '\004'           | '004'          | '004'        | '\005'           | '005'          | '005'        |
| '\006'           | '006'          | '006'        | '\007'           | '007'          | '007'        |
| '\010'           | '010'          | '00A'        | '\011'           | '011'          | '00B'        |
| '\012'           | '012'          | '00C'        | '\013'           | '013'          | '00D'        |
| '\014'           | '014'          | '00E'        | '\015'           | '015'          | '00F'        |
| '\016'           | '016'          | '010'        | '\017'           | '017'          | '011'        |
| '\020'           | '020'          | '012'        | '\021'           | '021'          | '013'        |
| '\022'           | '022'          | '014'        | '\023'           | '023'          | '015'        |
| '\024'           | '024'          | '016'        | '\025'           | '025'          | '017'        |
| '\026'           | '026'          | '018'        | '\027'           | '027'          | '019'        |
| '\030'           | '030'          | '01A'        | '\031'           | '031'          | '01B'        |
| '\032'           | '032'          | '01C'        | '\033'           | '033'          | '01D'        |
| '\034'           | '034'          | '01E'        | '\035'           | '035'          | '01F'        |
| '\036'           | '036'          | '020'        | '\037'           | '037'          | '021'        |
| '\040'           | '040'          | '022'        | '\041'           | '041'          | '023'        |
| '\042'           | '042'          | '024'        | '\043'           | '043'          | '025'        |
| '\044'           | '044'          | '026'        | '\045'           | '045'          | '027'        |
| '\046'           | '046'          | '028'        | '\047'           | '047'          | '029'        |
| '\048'           | '048'          | '02A'        | '\049'           | '049'          | '02B'        |
| '\050'           | '050'          | '02C'        | '\051'           | '051'          | '02D'        |
| '\052'           | '052'          | '02E'        | '\053'           | '053'          | '02F'        |
| '\054'           | '054'          | '030'        | '\055'           | '055'          | '031'        |
| '\056'           | '056'          | '032'        | '\057'           | '057'          | '033'        |
| '\060'           | '060'          | '034'        | '\061'           | '061'          | '035'        |
| '\062'           | '062'          | '036'        | '\063'           | '063'          | '037'        |
| '\064'           | '064'          | '038'        | '\065'           | '065'          | '039'        |
| '\066'           | '066'          | '03A'        | '\067'           | '067'          | '03B'        |
| '\070'           | '070'          | '03C'        | '\071'           | '071'          | '03D'        |
| '\072'           | '072'          | '03E'        | '\073'           | '073'          | '03F'        |
| '\074'           | '074'          | '040'        | '\075'           | '075'          | '041'        |
| '\077'           | '077'          | '042'        | '\078'           | '078'          | '043'        |
| '\080'           | '080'          | '044'        | '\081'           | '081'          | '045'        |
| '\082'           | '082'          | '046'        | '\083'           | '083'          | '047'        |
| '\084'           | '084'          | '048'        | '\085'           | '085'          | '049'        |
| '\086'           | '086'          | '04A'        | '\087'           | '087'          | '04B'        |
| '\088'           | '088'          | '04C'        | '\089'           | '089'          | '04D'        |
| '\090'           | '090'          | '04E'        | '\091'           | '091'          | '04F'        |
| '\092'           | '092'          | '050'        | '\093'           | '093'          | '051'        |
| '\094'           | '094'          | '052'        | '\095'           | '095'          | '053'        |
| '\096'           | '096'          | '054'        | '\097'           | '097'          | '055'        |
| '\098'           | '098'          | '056'        | '\099'           | '099'          | '057'        |
| '\0A0'           | '0A0'          | '058'        | '\0A1'           | '0A1'          | '059'        |
| '\0A2'           | '0A2'          | '05A'        | '\0A3'           | '0A3'          | '05B'        |
| '\0A4'           | '0A4'          | '05C'        | '\0A5'           | '0A5'          | '05D'        |
| '\0A6'           | '0A6'          | '05E'        | '\0A7'           | '0A7'          | '05F'        |
| '\0A8'           | '0A8'          | '060'        | '\0A9'           | '0A9'          | '061'        |
| '\0AA'           | '0AA'          | '062'        | '\0AB'           | '0AB'          | '063'        |
| '\0AC'           | '0AC'          | '064'        | '\0AD'           | '0AD'          | '065'        |
| '\0AE'           | '0AE'          | '066'        | '\0AF'           | '0AF'          | '067'        |
| '\0B0'           | '0B0'          | '068'        | '\0B1'           | '0B1'          | '069'        |
| '\0B2'           | '0B2'          | '06A'        | '\0B3'           | '0B3'          | '06B'        |
| '\0B4'           | '0B4'          | '06C'        | '\0B5'           | '0B5'          | '06D'        |
| '\0B6'           | '0B6'          | '06E'        | '\0B7'           | '0B7'          | '06F'        |
| '\0B8'           | '0B8'          | '070'        | '\0B9'           | '0B9'          | '071'        |
| '\0BA'           | '0BA'          | '072'        | '\0BB'           | '0BB'          | '073'        |
| '\0BC'           | '0BC'          | '074'        | '\0BD'           | '0BD'          | '075'        |
| '\0BE'           | '0BE'          | '076'        | '\0BF'           | '0BF'          | '077'        |
| '\0C0'           | '0C0'          | '078'        | '\0C1'           | '0C1'          | '079'        |
| '\0C2'           | '0C2'          | '07A'        | '\0C3'           | '0C3'          | '07B'        |
| '\0C4'           | '0C4'          | '07C'        | '\0C5'           | '0C5'          | '07D'        |
| '\0C6'           | '0C6'          | '07E'        | '\0C7'           | '0C7'          | '07F'        |
| '\0C8'           | '0C8'          | '080'        | '\0C9'           | '0C9'          | '081'        |
| '\0CA'           | '0CA'          | '082'        | '\0CB'           | '0CB'          | '083'        |
| '\0CC'           | '0CC'          | '084'        | '\0CD'           | '0CD'          | '085'        |
| '\0CE'           | '0CE'          | '086'        | '\0CF'           | '0CF'          | '087'        |
| '\0D0'           | '0D0'          | '088'        | '\0D1'           | '0D1'          | '089'        |
| '\0D2'           | '0D2'          | '08A'        | '\0D3'           | '0D3'          | '08B'        |
| '\0D4'           | '0D4'          | '08C'        | '\0D5'           | '0D5'          | '08D'        |
| '\0D6'           | '0D6'          | '08E'        | '\0D7'           | '0D7'          | '08F'        |
| '\0D8'           | '0D8'          | '090'        | '\0D9'           | '0D9'          | '091'        |
| '\0DA'           | '0DA'          | '092'        | '\0DB'           | '0DB'          | '093'        |
| '\0DC'           | '0DC'          | '094'        | '\0DD'           | '0DD'          | '095'        |
| '\0DE'           | '0DE'          | '096'        | '\0DF'           | '0DF'          | '097'        |
| '\0E0'           | '0E0'          | '098'        | '\0E1'           | '0E1'          | '099'        |
| '\0E2'           | '0E2'          | '09A'        | '\0E3'           | '0E3'          | '09B'        |
| '\0E4'           | '0E4'          | '09C'        | '\0E5'           | '0E5'          | '09D'        |
| '\0E6'           | '0E6'          | '09E'        | '\0E7'           | '0E7'          | '09F'        |
| '\0E8'           | '0E8'          | '0A0'        | '\0E9'           | '0E9'          | '0A1'        |
| '\0EA'           | '0EA'          | '0A2'        | '\0EB'           | '0EB'          | '0A3'        |
| '\0EC'           | '0EC'          | '0A4'        | '\0ED'           | '0ED'          | '0A5'        |
| '\0EE'           | '0EE'          | '0A6'        | '\0EF'           | '0EF'          | '0A7'        |
| '\0F0'           | '0F0'          | '0A8'        | '\0F1'           | '0F1'          | '0A9'        |
| '\0F2'           | '0F2'          | '0A0'        | '\0F3'           | '0F3'          | '0A1'        |
| '\0F4'           | '0F4'          | '0A2'        | '\0F5'           | '0F5'          | '0A3'        |
| '\0F6'           | '0F6'          | '0A4'        | '\0F7'           | '0F7'          | '0A5'        |
| '\0F8'           | '0F8'          | '0A6'        | '\0F9'           | '0F9'          | '0A7'        |
| '\0FA'           | '0FA'          | '0A8'        | '\0FB'           | '0FB'          | '0A9'        |
| '\0FC'           | '0FC'          | '0A0'        | '\0FD'           | '0FD'          | '0A1'        |
| '\0FE'           | '0FE'          | '0A2'        | '\0FF'           | '0FF'          | '0A3'        |

- `ord(c)`: returns Unicode (ASCII) of the character.
- Example: `ord('a')` returns 97.
- `chr(x)`: returns the character whose Unicode is x.

## Converting from Character to Code:

*(There is an ASCII table on the back of today's lecture slip.)*

| ASCII TABLE |     |       |      |      |         |     |       |         |      |         |     |       |         |      |
|-------------|-----|-------|------|------|---------|-----|-------|---------|------|---------|-----|-------|---------|------|
| Decimal     | Hex | Octal | Name | Char | Decimal | Hex | Octal | Name    | Char | Decimal | Hex | Octal | Name    | Char |
| 0           | 00  | 0     | NULL | \0   | 32      | 20  | 40    | SIGKILL | \000 | 64      | 40  | 100   | SIGPOLL | \001 |
| 1           | 01  | 1     | SOH  | \001 | 33      | 21  | 41    | SIGALRM | \002 | 65      | 41  | 101   | SIGSTOP | \003 |
| 2           | 02  | 2     | STX  | \002 | 34      | 22  | 42    | SIGPOLL | \004 | 66      | 42  | 102   | SIGCONT | \005 |
| 3           | 03  | 3     | ETX  | \003 | 35      | 23  | 43    | SIGPOLL | \006 | 67      | 43  | 103   | SIGKILL | \007 |
| 4           | 04  | 4     | ENQ  | \004 | 36      | 24  | 44    | SIGPOLL | \008 | 68      | 44  | 104   | SIGPOLL | \009 |
| 5           | 05  | 5     | KSYN | \005 | 37      | 25  | 45    | SIGPOLL | \010 | 69      | 45  | 105   | SIGPOLL | \011 |
| 6           | 06  | 6     | ACK  | \006 | 38      | 26  | 46    | SIGPOLL | \012 | 70      | 46  | 106   | SIGPOLL | \013 |
| 7           | 07  | 7     | NAK  | \007 | 39      | 27  | 47    | SIGPOLL | \014 | 71      | 47  | 107   | SIGPOLL | \015 |
| 8           | 08  | 10    | SYN  | \008 | 40      | 28  | 48    | SIGPOLL | \016 | 72      | 48  | 108   | SIGPOLL | \017 |
| 9           | 09  | 11    | EOT  | \009 | 41      | 29  | 49    | SIGPOLL | \018 | 73      | 49  | 109   | SIGPOLL | \019 |
| 10          | 0A  | 12    | EM   | \00A | 42      | 2A  | 4A    | SIGPOLL | \01A | 74      | 4A  | 110   | SIGPOLL | \01B |
| 11          | 0B  | 13    | END  | \00B | 43      | 2B  | 4B    | SIGPOLL | \01B | 75      | 4B  | 111   | SIGPOLL | \01C |
| 12          | 0C  | 14    | ESC  | \00C | 44      | 2C  | 4C    | SIGPOLL | \01C | 76      | 4C  | 112   | SIGPOLL | \01D |
| 13          | 0D  | 15    | SUSP | \00D | 45      | 2D  | 4D    | SIGPOLL | \01D | 77      | 4D  | 113   | SIGPOLL | \01E |
| 14          | 0E  | 16    | DC1  | \00E | 46      | 2E  | 4E    | SIGPOLL | \01E | 78      | 4E  | 114   | SIGPOLL | \01F |
| 15          | 0F  | 17    | DC2  | \00F | 47      | 2F  | 4F    | SIGPOLL | \01F | 79      | 4F  | 115   | SIGPOLL | \020 |
| 16          | 10  | 20    | DC3  | \010 | 48      | 30  | 50    | SIGPOLL | \01F | 80      | 50  | 116   | SIGPOLL | \021 |
| 17          | 11  | 21    | DC4  | \011 | 49      | 31  | 51    | SIGPOLL | \01F | 81      | 51  | 117   | SIGPOLL | \022 |
| 18          | 12  | 22    | DC5  | \012 | 50      | 32  | 52    | SIGPOLL | \01F | 82      | 52  | 118   | SIGPOLL | \023 |
| 19          | 13  | 23    | DC6  | \013 | 51      | 33  | 53    | SIGPOLL | \01F | 83      | 53  | 119   | SIGPOLL | \024 |
| 20          | 14  | 24    | DC7  | \014 | 52      | 34  | 54    | SIGPOLL | \01F | 84      | 54  | 120   | SIGPOLL | \025 |
| 21          | 15  | 25    | DC8  | \015 | 53      | 35  | 55    | SIGPOLL | \01F | 85      | 55  | 121   | SIGPOLL | \026 |
| 22          | 16  | 26    | DC9  | \016 | 54      | 36  | 56    | SIGPOLL | \01F | 86      | 56  | 122   | SIGPOLL | \027 |
| 23          | 17  | 27    | DC10 | \017 | 55      | 37  | 57    | SIGPOLL | \01F | 87      | 57  | 123   | SIGPOLL | \028 |
| 24          | 18  | 28    | DC11 | \018 | 56      | 38  | 58    | SIGPOLL | \01F | 88      | 58  | 124   | SIGPOLL | \029 |
| 25          | 19  | 29    | DC12 | \019 | 57      | 39  | 59    | SIGPOLL | \01F | 89      | 59  | 125   | SIGPOLL | \02A |
| 26          | 1A  | 2A    | DC13 | \01A | 58      | 3A  | 5A    | SIGPOLL | \01F | 90      | 5A  | 126   | SIGPOLL | \02B |
| 27          | 1B  | 2B    | DC14 | \01B | 59      | 3B  | 5B    | SIGPOLL | \01F | 91      | 5B  | 127   | SIGPOLL | \02C |
| 28          | 1C  | 2C    | DC15 | \01C | 5A      | 3C  | 5C    | SIGPOLL | \01F | 92      | 5C  | 128   | SIGPOLL | \02D |
| 29          | 1D  | 2D    | DC16 | \01D | 5B      | 3D  | 5D    | SIGPOLL | \01F | 93      | 5D  | 129   | SIGPOLL | \02E |
| 30          | 1E  | 2E    | DC17 | \01E | 5C      | 3E  | 5E    | SIGPOLL | \01F | 94      | 5E  | 130   | SIGPOLL | \02F |
| 31          | 1F  | 2F    | DC18 | \01F | 5D      | 3F  | 5F    | SIGPOLL | \01F | 95      | 5F  | 131   | SIGPOLL | \030 |
| 32          | 20  | 30    | DC19 | \020 | 5E      | 40  | 60    | SIGPOLL | \01F | 96      | 60  | 132   | SIGPOLL | \031 |
| 33          | 21  | 31    | DC1A | \021 | 5F      | 41  | 61    | SIGPOLL | \01F | 97      | 61  | 133   | SIGPOLL | \032 |
| 34          | 22  | 32    | DC1B | \022 | 60      | 42  | 62    | SIGPOLL | \01F | 98      | 62  | 134   | SIGPOLL | \033 |
| 35          | 23  | 33    | DC1C | \023 | 61      | 43  | 63    | SIGPOLL | \01F | 99      | 63  | 135   | SIGPOLL | \034 |
| 36          | 24  | 34    | DC1D | \024 | 62      | 44  | 64    | SIGPOLL | \01F | 100     | 64  | 136   | SIGPOLL | \035 |
| 37          | 25  | 35    | DC1E | \025 | 63      | 45  | 65    | SIGPOLL | \01F | 101     | 65  | 137   | SIGPOLL | \036 |
| 38          | 26  | 36    | DC1F | \026 | 64      | 46  | 66    | SIGPOLL | \01F | 102     | 66  | 138   | SIGPOLL | \037 |
| 39          | 27  | 37    | DC20 | \027 | 65      | 47  | 67    | SIGPOLL | \01F | 103     | 67  | 139   | SIGPOLL | \038 |
| 40          | 28  | 38    | DC21 | \028 | 66      | 48  | 68    | SIGPOLL | \01F | 104     | 68  | 140   | SIGPOLL | \039 |
| 41          | 29  | 39    | DC22 | \029 | 67      | 49  | 69    | SIGPOLL | \01F | 105     | 69  | 141   | SIGPOLL | \03A |
| 42          | 2A  | 3A    | DC23 | \02A | 68      | 4A  | 6A    | SIGPOLL | \01F | 106     | 6A  | 142   | SIGPOLL | \03B |
| 43          | 2B  | 3B    | DC24 | \02B | 69      | 4B  | 6B    | SIGPOLL | \01F | 107     | 6B  | 143   | SIGPOLL | \03C |
| 44          | 2C  | 3C    | DC25 | \02C | 6A      | 4C  | 6C    | SIGPOLL | \01F | 108     | 6C  | 144   | SIGPOLL | \03D |
| 45          | 2D  | 3D    | DC26 | \02D | 6B      | 4D  | 6D    | SIGPOLL | \01F | 109     | 6D  | 145   | SIGPOLL | \03E |
| 46          | 2E  | 3E    | DC27 | \02E | 6C      | 4E  | 6E    | SIGPOLL | \01F | 110     | 6E  | 146   | SIGPOLL | \03F |
| 47          | 2F  | 3F    | DC28 | \02F | 6D      | 4F  | 6F    | SIGPOLL | \01F | 111     | 6F  | 147   | SIGPOLL | \040 |
| 48          | 30  | 40    | DC29 | \030 | 6E      | 50  | 70    | SIGPOLL | \01F | 112     | 70  | 148   | SIGPOLL | \041 |
| 49          | 31  | 41    | DC2A | \031 | 6F      | 51  | 71    | SIGPOLL | \01F | 113     | 71  | 149   | SIGPOLL | \042 |
| 50          | 32  | 42    | DC2B | \032 | 70      | 52  | 72    | SIGPOLL | \01F | 114     | 72  | 150   | SIGPOLL | \043 |
| 51          | 33  | 43    | DC2C | \033 | 71      | 53  | 73    | SIGPOLL | \01F | 115     | 73  | 151   | SIGPOLL | \044 |
| 52          | 34  | 44    | DC2D | \034 | 72      | 54  | 74    | SIGPOLL | \01F | 116     | 74  | 152   | SIGPOLL | \045 |
| 53          | 35  | 45    | DC2E | \035 | 73      | 55  | 75    | SIGPOLL | \01F | 117     | 75  | 153   | SIGPOLL | \046 |
| 54          | 36  | 46    | DC2F | \036 | 74      | 56  | 76    | SIGPOLL | \01F | 118     | 76  | 154   | SIGPOLL | \047 |
| 55          | 37  | 47    | DC30 | \037 | 75      | 57  | 77    | SIGPOLL | \01F | 119     | 77  | 155   | SIGPOLL | \048 |
| 56          | 38  | 48    | DC31 | \038 | 76      | 58  | 78    | SIGPOLL | \01F | 120     | 78  | 156   | SIGPOLL | \049 |
| 57          | 39  | 49    | DC32 | \039 | 77      | 59  | 79    | SIGPOLL | \01F | 121     | 79  | 157   | SIGPOLL | \04A |
| 58          | 3A  | 4A    | DC33 | \03A | 78      | 5A  | 7A    | SIGPOLL | \01F | 122     | 7A  | 158   | SIGPOLL | \04B |
| 59          | 3B  | 4B    | DC34 | \03B | 79      | 5B  | 7B    | SIGPOLL | \01F | 123     | 7B  | 159   | SIGPOLL | \04C |
| 60          | 3C  | 4C    | DC35 | \03C | 7A      | 5C  | 7C    | SIGPOLL | \01F | 124     | 7C  | 160   | SIGPOLL | \04D |
| 61          | 3D  | 4D    | DC36 | \03D | 7B      | 5D  | 7D    | SIGPOLL | \01F | 125     | 7D  | 161   | SIGPOLL | \04E |
| 62          | 3E  | 4E    | DC37 | \03E | 7C      | 5E  | 7E    | SIGPOLL | \01F | 126     | 7E  | 162   | SIGPOLL | \04F |
| 63          | 3F  | 4F    | DC38 | \03F | 7D      | 5F  | 7F    | SIGPOLL | \01F | 127     | 7F  | 163   | SIGPOLL | \050 |
| 64          | 40  | 50    | DC39 | \040 | 7E      | 60  | 80    | SIGPOLL | \01F | 128     | 80  | 164   | SIGPOLL | \051 |
| 65          | 41  | 51    | DC3A | \041 | 7F      | 61  | 81    | SIGPOLL | \01F | 129     | 81  | 165   | SIGPOLL | \052 |
| 66          | 42  | 52    | DC3B | \042 | 80      | 62  | 82    | SIGPOLL | \01F | 130     | 82  | 166   | SIGPOLL | \053 |
| 67          | 43  | 53    | DC3C | \043 | 81      | 63  | 83    | SIGPOLL | \01F | 131     | 83  | 167   | SIGPOLL | \054 |
| 68          | 44  | 54    | DC3D | \044 | 82      | 64  | 84    | SIGPOLL | \01F | 132     | 84  | 168   | SIGPOLL | \055 |
| 69          | 45  | 55    | DC3E | \045 | 83      | 65  | 85    | SIGPOLL | \01F | 133     | 85  | 169   | SIGPOLL | \056 |
| 70          | 46  | 56    | DC3F | \046 | 84      | 66  | 86    | SIGPOLL | \01F | 134     | 86  | 170   | SIGPOLL | \057 |
| 71          | 47  | 57    | DC40 | \047 | 85      | 67  | 87    | SIGPOLL | \01F | 135     | 87  | 171   | SIGPOLL | \058 |
| 72          | 48  | 58    | DC41 | \048 | 86      | 68  | 88    | SIGPOLL | \01F | 136     | 88  | 172   | SIGPOLL | \059 |
| 73          | 49  | 59    | DC42 | \049 | 87      | 69  | 89    | SIGPOLL | \01F | 137     | 89  | 173   | SIGPOLL | \05A |
| 74          | 4A  | 5A    | DC43 | \04A | 88      | 6A  | 8A    | SIGPOLL | \01F | 138     | 8A  | 174   | SIGPOLL | \05B |
| 75          | 4B  | 5B    | DC44 | \04B | 89      | 6B  | 8B    | SIGPOLL | \01F | 139     | 8B  | 175   | SIGPOLL | \05C |
| 76          | 4C  | 5C    | DC45 | \04C | 8A      | 6C  | 8C    | SIGPOLL | \01F | 140     | 8C  | 176   | SIGPOLL | \05D |
| 77          | 4D  | 5D    | DC46 | \04D | 8B      | 6D  | 8D    | SIGPOLL | \01F | 141     | 8D  | 177   | SIGPOLL | \05E |
| 78          | 4E  | 5E    | DC47 | \04E | 8C      | 6E  | 8E    | SIGPOLL | \01F | 142     | 8E  | 178   | SIGPOLL | \05F |
| 79          | 4F  | 5F    | DC48 | \04F | 8D      | 6F  | 8F    | SIGPOLL | \01F | 143     | 8F  | 179   | SIGPOLL | \060 |
| 80          | 50  | 60    | DC49 | \050 | 8E      | 70  | 90    | SIGPOLL | \01F | 144     | 90  | 180   | SIGPOLL | \061 |
| 81          | 51  | 61    | DC4A | \051 | 8F      | 71  | 91    | SIGPOLL | \01F | 145     | 91  | 181   | SIGPOLL | \062 |
| 82          | 52  | 62    | DC4B | \052 | 90      | 72  | 92    | SIGPOLL | \01F | 146     | 92  | 182   | SIGPOLL | \063 |
| 83          | 53  | 63    | DC4C | \053 | 91      | 73  | 93    | SIGPOLL | \01F | 147     | 93  | 183   | SIGPOLL | \064 |
| 84          | 54  | 64    | DC4D | \054 | 92      | 74  | 94    | SIGPOLL | \01F | 148     | 94  | 184   | SIGPOLL | \065 |
| 85          | 55  | 65    | DC4E | \055 | 93      | 75  | 95    | SIGPOLL | \01F | 149     | 95  | 185   | SIGPOLL | \066 |
| 86          | 56  | 66    | DC4F | \056 | 94      | 76  | 96    | SIGPOLL | \01F | 150     | 96  | 186   | SIGPOLL | \067 |
| 87          | 57  | 67    | DC50 | \057 | 95      | 77  | 97    | SIGPOLL | \01F | 151     | 97  | 187   | SIGPOLL | \068 |
| 88          | 58  | 68    | DC51 | \058 | 96      | 78  | 98    | SIGPOLL | \01F | 152     | 98  | 188   | SIGPOLL | \069 |
| 89          | 59  | 69    | DC52 | \059 | 97      | 79  | 99    | SIGPOLL | \01F | 153     | 99  | 189   | SIGPOLL | \06A |
| 90          | 5A  | 6A    | DC53 | \05A | 98      | 7A  | 9A    | SIGPOLL | \01F | 154     | 9A  | 190   | SIGPOLL | \06B |
| 91          | 5B  | 6B    | DC54 | \05B | 99      | 7B  | 9B    | SIGPOLL | \01F | 155     | 9B  | 191   | SIGPOLL | \06C |
| 92          | 5C  | 6C    | DC55 | \05C | 9A      | 7C  | 9C    | SIGPOLL | \01F | 156     | 9C  | 192   | SIGPOLL | \06D |
| 93          | 5D  | 6D    | DC56 | \05D | 9B      | 7D  | 9D    | SIGPOLL | \01F | 157     | 9D  | 193   | SIGPOLL | \06E |
| 94          | 5E  | 6E    | DC57 | \05E | 9C      | 7E  | 9E    | SIGPOLL | \01F | 158     | 9E  | 194   | SIGPOLL | \06F |
| 95          | 5F  | 6F    | DC58 | \05F | 9D      | 7F  | 9F    | SIGPOLL | \01F | 159     | 9F  | 195   | SIGPOLL | \070 |
| 96          | 60  | 70    | DC59 | \060 | 9E      | 80  | A0    | SIGPOLL | \01F | 160     | A0  | 196   | SIGPOLL | \071 |
| 97          | 61  | 71    | DC5A | \061 | 9F      | 81  | A1    | SIGPOLL | \01F | 161     | A1  | 197   | SIGPOLL | \072 |
| 98          | 62  | 72    | DC5B | \062 | A0      | 82  | A2    | SIGPOLL | \01F | 162     | A2  | 198   | SIGPOLL | \073 |
| 99          | 63  | 73    | DC5C | \063 | A1      | 83  | A3    | SIGPOLL | \01F | 163     | A3  | 199   | SIGPOLL | \074 |
| 100         | 64  | 74    | DC5D | \064 | A2      | 84  | A4    | SIGPOLL | \01F | 164     | A4  | 200   | SIGPOLL | \075 |
| 101         | 65  | 75    | DC5E | \065 | A3      | 85  | A5    | SIGPOLL | \01F | 165     | A5  | 201   | SIGPOLL | \076 |
| 102         | 66  | 76    | DC5F | \066 | A4      | 86  | A6    | SIGPOLL | \01F | 166     | A6  |       |         |      |

- `ord(c)`: returns Unicode (ASCII) of the character.
  - Example: `ord('a')` returns 97.
  - `chr(x)`: returns the character whose Unicode is x.
  - Example: `chr(97)` returns 'a'.

## Converting from Character to Code:

*(There is an ASCII table on the back of today's lecture slip.)*

| ASCII TABLE |     |       |      |      |         |     |       |         |      |         |     |       |         |      |
|-------------|-----|-------|------|------|---------|-----|-------|---------|------|---------|-----|-------|---------|------|
| Decimal     | Hex | Octal | Name | Char | Decimal | Hex | Octal | Name    | Char | Decimal | Hex | Octal | Name    | Char |
| 0           | 00  | 0     | NULL | \0   | 32      | 20  | 40    | SIGKILL | \000 | 64      | 40  | 100   | SIGPOLL | \001 |
| 1           | 01  | 1     | SOH  | \001 | 33      | 21  | 41    | SIGALRM | \002 | 65      | 41  | 101   | SIGSTOP | \003 |
| 2           | 02  | 2     | STX  | \002 | 34      | 22  | 42    | SIGPOLL | \004 | 66      | 42  | 102   | SIGCONT | \005 |
| 3           | 03  | 3     | ETX  | \003 | 35      | 23  | 43    | SIGPOLL | \006 | 67      | 43  | 103   | SIGKILL | \007 |
| 4           | 04  | 4     | EOT  | \004 | 36      | 24  | 44    | SIGPOLL | \008 | 68      | 44  | 104   | SIGPOLL | \009 |
| 5           | 05  | 5     | ENQ  | \005 | 37      | 25  | 45    | SIGPOLL | \010 | 69      | 45  | 105   | SIGPOLL | \011 |
| 6           | 06  | 6     | ACK  | \006 | 38      | 26  | 46    | SIGPOLL | \012 | 70      | 46  | 106   | SIGPOLL | \013 |
| 7           | 07  | 7     | NAK  | \007 | 39      | 27  | 47    | SIGPOLL | \014 | 71      | 47  | 107   | SIGPOLL | \015 |
| 8           | 08  | 10    | SYN  | \008 | 40      | 28  | 48    | SIGPOLL | \016 | 72      | 48  | 108   | SIGPOLL | \017 |
| 9           | 09  | 11    | BT   | \009 | 41      | 29  | 49    | SIGPOLL | \018 | 73      | 49  | 109   | SIGPOLL | \019 |
| 10          | 0A  | 12    | HT   | \00A | 42      | 2A  | 4A    | SIGPOLL | \01A | 74      | 4A  | 110   | SIGPOLL | \01B |
| 11          | 0B  | 13    | LF   | \00B | 43      | 2B  | 4B    | SIGPOLL | \01B | 75      | 4B  | 111   | SIGPOLL | \01C |
| 12          | 0C  | 14    | VT   | \00C | 44      | 2C  | 4C    | SIGPOLL | \01C | 76      | 4C  | 112   | SIGPOLL | \01D |
| 13          | 0D  | 15    | FF   | \00D | 45      | 2D  | 4D    | SIGPOLL | \01D | 77      | 4D  | 113   | SIGPOLL | \01E |
| 14          | 0E  | 16    | CR   | \00E | 46      | 2E  | 4E    | SIGPOLL | \01E | 78      | 4E  | 114   | SIGPOLL | \01F |
| 15          | 0F  | 17    | SO   | \00F | 47      | 2F  | 4F    | SIGPOLL | \01F | 79      | 4F  | 115   | SIGPOLL | \01F |
| 16          | 10  | 20    | SI   | \010 | 48      | 30  | 50    | SIGPOLL | \01F | 80      | 50  | 116   | SIGPOLL | \01F |
| 17          | 11  | 21    | DC1  | \011 | 49      | 31  | 51    | SIGPOLL | \01F | 81      | 51  | 117   | SIGPOLL | \01F |
| 18          | 12  | 22    | DC2  | \012 | 50      | 32  | 52    | SIGPOLL | \01F | 82      | 52  | 118   | SIGPOLL | \01F |
| 19          | 13  | 23    | DC3  | \013 | 51      | 33  | 53    | SIGPOLL | \01F | 83      | 53  | 119   | SIGPOLL | \01F |
| 20          | 14  | 24    | DC4  | \014 | 52      | 34  | 54    | SIGPOLL | \01F | 84      | 54  | 120   | SIGPOLL | \01F |
| 21          | 15  | 25    | NAK  | \015 | 53      | 35  | 55    | SIGPOLL | \01F | 85      | 55  | 121   | SIGPOLL | \01F |
| 22          | 16  | 26    | SYN  | \016 | 54      | 36  | 56    | SIGPOLL | \01F | 86      | 56  | 122   | SIGPOLL | \01F |
| 23          | 17  | 27    | ETX  | \017 | 55      | 37  | 57    | SIGPOLL | \01F | 87      | 57  | 123   | SIGPOLL | \01F |
| 24          | 18  | 28    | ENQ  | \018 | 56      | 38  | 58    | SIGPOLL | \01F | 88      | 58  | 124   | SIGPOLL | \01F |
| 25          | 19  | 29    | ACK  | \019 | 57      | 39  | 59    | SIGPOLL | \01F | 89      | 59  | 125   | SIGPOLL | \01F |
| 26          | 1A  | 2A    | BT   | \01A | 58      | 3A  | 5A    | SIGPOLL | \01F | 90      | 5A  | 126   | SIGPOLL | \01F |
| 27          | 1B  | 2B    | HT   | \01B | 59      | 3B  | 5B    | SIGPOLL | \01F | 91      | 5B  | 127   | SIGPOLL | \01F |
| 28          | 1C  | 2C    | LF   | \01C | 60      | 3C  | 5C    | SIGPOLL | \01F | 92      | 5C  | 128   | SIGPOLL | \01F |
| 29          | 1D  | 2D    | VT   | \01D | 61      | 3D  | 5D    | SIGPOLL | \01F | 93      | 5D  | 129   | SIGPOLL | \01F |
| 30          | 1E  | 2E    | FF   | \01E | 62      | 3E  | 5E    | SIGPOLL | \01F | 94      | 5E  | 130   | SIGPOLL | \01F |
| 31          | 1F  | 2F    | CR   | \01F | 63      | 3F  | 5F    | SIGPOLL | \01F | 95      | 5F  | 131   | SIGPOLL | \01F |
| 32          | 20  | 30    | SO   | \020 | 64      | 40  | 60    | SIGPOLL | \01F | 96      | 60  | 132   | SIGPOLL | \01F |
| 33          | 21  | 31    | SI   | \021 | 65      | 41  | 61    | SIGPOLL | \01F | 97      | 61  | 133   | SIGPOLL | \01F |
| 34          | 22  | 32    | DC1  | \022 | 66      | 42  | 62    | SIGPOLL | \01F | 98      | 62  | 134   | SIGPOLL | \01F |
| 35          | 23  | 33    | DC2  | \023 | 67      | 43  | 63    | SIGPOLL | \01F | 99      | 63  | 135   | SIGPOLL | \01F |
| 36          | 24  | 34    | DC3  | \024 | 68      | 44  | 64    | SIGPOLL | \01F | 100     | 64  | 136   | SIGPOLL | \01F |
| 37          | 25  | 35    | DC4  | \025 | 69      | 45  | 65    | SIGPOLL | \01F | 101     | 65  | 137   | SIGPOLL | \01F |
| 38          | 26  | 36    | NAK  | \026 | 70      | 46  | 66    | SIGPOLL | \01F | 102     | 66  | 138   | SIGPOLL | \01F |
| 39          | 27  | 37    | SYN  | \027 | 71      | 47  | 67    | SIGPOLL | \01F | 103     | 67  | 139   | SIGPOLL | \01F |
| 40          | 28  | 38    | ETX  | \028 | 72      | 48  | 68    | SIGPOLL | \01F | 104     | 68  | 140   | SIGPOLL | \01F |
| 41          | 29  | 39    | ENQ  | \029 | 73      | 49  | 69    | SIGPOLL | \01F | 105     | 69  | 141   | SIGPOLL | \01F |
| 42          | 2A  | 3A    | ACK  | \02A | 74      | 4A  | 6A    | SIGPOLL | \01F | 106     | 6A  | 142   | SIGPOLL | \01F |
| 43          | 2B  | 3B    | BT   | \02B | 75      | 4B  | 6B    | SIGPOLL | \01F | 107     | 6B  | 143   | SIGPOLL | \01F |
| 44          | 2C  | 3C    | HT   | \02C | 76      | 4C  | 6C    | SIGPOLL | \01F | 108     | 6C  | 144   | SIGPOLL | \01F |
| 45          | 2D  | 3D    | LF   | \02D | 77      | 4D  | 6D    | SIGPOLL | \01F | 109     | 6D  | 145   | SIGPOLL | \01F |
| 46          | 2E  | 3E    | VT   | \02E | 78      | 4E  | 6E    | SIGPOLL | \01F | 110     | 6E  | 146   | SIGPOLL | \01F |
| 47          | 2F  | 3F    | FF   | \02F | 79      | 4F  | 6F    | SIGPOLL | \01F | 111     | 6F  | 147   | SIGPOLL | \01F |
| 48          | 30  | 40    | CR   | \030 | 80      | 50  | 70    | SIGPOLL | \01F | 112     | 70  | 148   | SIGPOLL | \01F |
| 49          | 31  | 41    | SO   | \031 | 81      | 51  | 71    | SIGPOLL | \01F | 113     | 71  | 149   | SIGPOLL | \01F |
| 50          | 32  | 42    | SI   | \032 | 82      | 52  | 72    | SIGPOLL | \01F | 114     | 72  | 150   | SIGPOLL | \01F |
| 51          | 33  | 43    | DC1  | \033 | 83      | 53  | 73    | SIGPOLL | \01F | 115     | 73  | 151   | SIGPOLL | \01F |
| 52          | 34  | 44    | DC2  | \034 | 84      | 54  | 74    | SIGPOLL | \01F | 116     | 74  | 152   | SIGPOLL | \01F |
| 53          | 35  | 45    | DC3  | \035 | 85      | 55  | 75    | SIGPOLL | \01F | 117     | 75  | 153   | SIGPOLL | \01F |
| 54          | 36  | 46    | DC4  | \036 | 86      | 56  | 76    | SIGPOLL | \01F | 118     | 76  | 154   | SIGPOLL | \01F |
| 55          | 37  | 47    | NAK  | \037 | 87      | 57  | 77    | SIGPOLL | \01F | 119     | 77  | 155   | SIGPOLL | \01F |
| 56          | 38  | 48    | SYN  | \038 | 88      | 58  | 78    | SIGPOLL | \01F | 120     | 78  | 156   | SIGPOLL | \01F |
| 57          | 39  | 49    | ETX  | \039 | 89      | 59  | 79    | SIGPOLL | \01F | 121     | 79  | 157   | SIGPOLL | \01F |
| 58          | 3A  | 4A    | ENQ  | \03A | 90      | 5A  | 7A    | SIGPOLL | \01F | 122     | 7A  | 158   | SIGPOLL | \01F |
| 59          | 3B  | 4B    | ACK  | \03B | 91      | 5B  | 7B    | SIGPOLL | \01F | 123     | 7B  | 159   | SIGPOLL | \01F |
| 60          | 3C  | 4C    | BT   | \03C | 92      | 5C  | 7C    | SIGPOLL | \01F | 124     | 7C  | 160   | SIGPOLL | \01F |
| 61          | 3D  | 4D    | HT   | \03D | 93      | 5D  | 7D    | SIGPOLL | \01F | 125     | 7D  | 161   | SIGPOLL | \01F |
| 62          | 3E  | 4E    | LF   | \03E | 94      | 5E  | 7E    | SIGPOLL | \01F | 126     | 7E  | 162   | SIGPOLL | \01F |
| 63          | 3F  | 4F    | VT   | \03F | 95      | 5F  | 7F    | SIGPOLL | \01F | 127     | 7F  | 163   | SIGPOLL | \01F |
| 64          | 40  | 50    | FF   | \040 | 96      | 60  | 80    | SIGPOLL | \01F | 128     | 80  | 164   | SIGPOLL | \01F |
| 65          | 41  | 51    | CR   | \041 | 97      | 61  | 81    | SIGPOLL | \01F | 129     | 81  | 165   | SIGPOLL | \01F |
| 66          | 42  | 52    | SO   | \042 | 98      | 62  | 82    | SIGPOLL | \01F | 130     | 82  | 166   | SIGPOLL | \01F |
| 67          | 43  | 53    | SI   | \043 | 99      | 63  | 83    | SIGPOLL | \01F | 131     | 83  | 167   | SIGPOLL | \01F |
| 68          | 44  | 54    | DC1  | \044 | 100     | 64  | 84    | SIGPOLL | \01F | 132     | 84  | 168   | SIGPOLL | \01F |
| 69          | 45  | 55    | DC2  | \045 | 101     | 65  | 85    | SIGPOLL | \01F | 133     | 85  | 169   | SIGPOLL | \01F |
| 70          | 46  | 56    | DC3  | \046 | 102     | 66  | 86    | SIGPOLL | \01F | 134     | 86  | 170   | SIGPOLL | \01F |
| 71          | 47  | 57    | DC4  | \047 | 103     | 67  | 87    | SIGPOLL | \01F | 135     | 87  | 171   | SIGPOLL | \01F |
| 72          | 48  | 58    | NAK  | \048 | 104     | 68  | 88    | SIGPOLL | \01F | 136     | 88  | 172   | SIGPOLL | \01F |
| 73          | 49  | 59    | SYN  | \049 | 105     | 69  | 89    | SIGPOLL | \01F | 137     | 89  | 173   | SIGPOLL | \01F |
| 74          | 4A  | 5A    | ETX  | \04A | 106     | 6A  | 8A    | SIGPOLL | \01F | 138     | 8A  | 174   | SIGPOLL | \01F |
| 75          | 4B  | 5B    | ENQ  | \04B | 107     | 6B  | 8B    | SIGPOLL | \01F | 139     | 8B  | 175   | SIGPOLL | \01F |
| 76          | 4C  | 5C    | ACK  | \04C | 108     | 6C  | 8C    | SIGPOLL | \01F | 140     | 8C  | 176   | SIGPOLL | \01F |
| 77          | 4D  | 5D    | BT   | \04D | 109     | 6D  | 8D    | SIGPOLL | \01F | 141     | 8D  | 177   | SIGPOLL | \01F |
| 78          | 4E  | 5E    | HT   | \04E | 110     | 6E  | 8E    | SIGPOLL | \01F | 142     | 8E  | 178   | SIGPOLL | \01F |
| 79          | 4F  | 5F    | LF   | \04F | 111     | 6F  | 8F    | SIGPOLL | \01F | 143     | 8F  | 179   | SIGPOLL | \01F |
| 80          | 50  | 60    | VT   | \050 | 112     | 70  | 90    | SIGPOLL | \01F | 144     | 90  | 180   | SIGPOLL | \01F |
| 81          | 51  | 61    | FF   | \051 | 113     | 71  | 91    | SIGPOLL | \01F | 145     | 91  | 181   | SIGPOLL | \01F |
| 82          | 52  | 62    | CR   | \052 | 114     | 72  | 92    | SIGPOLL | \01F | 146     | 92  | 182   | SIGPOLL | \01F |
| 83          | 53  | 63    | SO   | \053 | 115     | 73  | 93    | SIGPOLL | \01F | 147     | 93  | 183   | SIGPOLL | \01F |
| 84          | 54  | 64    | SI   | \054 | 116     | 74  | 94    | SIGPOLL | \01F | 148     | 94  | 184   | SIGPOLL | \01F |
| 85          | 55  | 65    | DC1  | \055 | 117     | 75  | 95    | SIGPOLL | \01F | 149     | 95  | 185   | SIGPOLL | \01F |
| 86          | 56  | 66    | DC2  | \056 | 118     | 76  | 96    | SIGPOLL | \01F | 150     | 96  | 186   | SIGPOLL | \01F |
| 87          | 57  | 67    | DC3  | \057 | 119     | 77  | 97    | SIGPOLL | \01F | 151     | 97  | 187   | SIGPOLL | \01F |
| 88          | 58  | 68    | DC4  | \058 | 120     | 78  | 98    | SIGPOLL | \01F | 152     | 98  | 188   | SIGPOLL | \01F |
| 89          | 59  | 69    | NAK  | \059 | 121     | 79  | 99    | SIGPOLL | \01F | 153     | 99  | 189   | SIGPOLL | \01F |
| 90          | 5A  | 6A    | SYN  | \05A | 122     | 7A  | 9A    | SIGPOLL | \01F | 154     | 9A  | 190   | SIGPOLL | \01F |
| 91          | 5B  | 6B    | ETX  | \05B | 123     | 7B  | 9B    | SIGPOLL | \01F | 155     | 9B  | 191   | SIGPOLL | \01F |
| 92          | 5C  | 6C    | ENQ  | \05C | 124     | 7C  | 9C    | SIGPOLL | \01F | 156     | 9C  | 192   | SIGPOLL | \01F |
| 93          | 5D  | 6D    | ACK  | \05D | 125     | 7D  | 9D    | SIGPOLL | \01F | 157     | 9D  | 193   | SIGPOLL | \01F |
| 94          | 5E  | 6E    | BT   | \05E | 126     | 7E  | 9E    | SIGPOLL | \01F | 158     | 9E  | 194   | SIGPOLL | \01F |
| 95          | 5F  | 6F    | HT   | \05F | 127     | 7F  | 9F    | SIGPOLL | \01F | 159     | 9F  | 195   | SIGPOLL | \01F |
| 96          | 60  | 70    | LF   | \060 | 128     | 80  | A0    | SIGPOLL | \01F | 160     | A0  | 196   | SIGPOLL | \01F |
| 97          | 61  | 71    | VT   | \061 | 129     | 81  | A1    | SIGPOLL | \01F | 161     | A1  | 197   | SIGPOLL | \01F |
| 98          | 62  | 72    | FF   | \062 | 130     | 82  | A2    | SIGPOLL | \01F | 162     | A2  | 198   | SIGPOLL | \01F |
| 99          | 63  | 73    | CR   | \063 | 131     | 83  | A3    | SIGPOLL | \01F | 163     | A3  | 199   | SIGPOLL | \01F |
| 100         | 64  | 74    | SO   | \064 | 132     | 84  | A4    | SIGPOLL | \01F | 164     | A4  | 200   | SIGPOLL | \01F |
| 101         | 65  | 75    | SI   | \065 | 133     | 85  | A5    | SIGPOLL | \01F | 165     | A5  | 201   | SIGPOLL | \01F |
| 102         | 66  | 76    | DC1  | \066 | 134     | 86  | A6    | SIGPOLL | \01F | 166     | A6  | 202   | SIGPOLL | \01F |
| 103         | 67  | 77    | DC2  | \067 | 135     | 87  | A7    | SIGPOLL |      |         |     |       |         |      |

- `ord(c)`: returns Unicode (ASCII) of the character.
  - Example: `ord('a')` returns 97.
  - `chr(x)`: returns the character whose Unicode is  $x$ .
  - Example: `chr(97)` returns 'a'.
  - What is `chr(33)`?

# In Pairs or Triples...

Some review and some novel challenges:

```
1 #Predict what will be printed:  
2  
3 for c in range(65,90):  
4     print(chr(c))  
5  
6 message = "I love Python"  
7 newMessage = ""  
8 for c in message:  
9     print(ord(c))    #Print the Unicode of each number  
10    print(chr(ord(c)+1))    #Print the next character  
11    newMessage = newMessage + chr(ord(c)+1) #add to the new message  
12 print("The coded message is", newMessage)  
13  
14 word = "zebra"  
15 codedWord = ""  
16 for ch in word:  
17     offset = ord(ch) - ord('a') + 1 #how many letters past 'a'  
18     wrap = offset % 26    #if larger than 26, wrap back to 0  
19     newChar = chr(ord('a') + wrap)    #compute the new letter  
20     print(wrap, chr(ord('a') + wrap))    #print the wrap & new lett  
21     codedWord = codedWord + newChar #add the newChar to the coded w  
22  
23 print("The coded word (with wrap) is", codedWord)
```



# Python Tutor

```
1 #Predict what will be printed:  
2  
3 for c in range(65,90):  
4     print(chr(c))  
5  
6 message = "I love Python"  
7 newMessage = ""  
8 for c in message:  
9     print(ord(c)) #Print the Unicode of each number  
10    print(chr(ord(c)+1)) #Print the next character  
11    newMessage = newMessage + chr(ord(c)+1) #Add to the new message  
12 print("The coded message is", newMessage)  
13  
14 word = "zebra"  
15 codedWord = ""  
16 for ch in word:  
17     offset = ord(ch) - ord('a') + 1 #how many letters past 'a'  
18     wrap = offset % 26 #if offset is 26, wrap back to 0  
19     newChar = chr(ord('a') + wrap) #compute the new letter  
20     print(wrap, chr(ord('a') + wrap)) #print the wrap & new lett  
21     codedWord = codedWord + newChar #add the newChar to the coded w  
22  
23 print("The coded word (with wrap) is", codedWord)
```

(Demo with pythonTutor)

# User Input

*Covered in detail in Lab 2:*

---

```
→ 1 mess = input('Please enter a message: ')
  2 print("You entered", mess)
```

---

(Demo with pythonTutor)

## Side Note: '+' for numbers and strings

- `x = 3 + 5` stores the number 8 in memory location `x`.



## Side Note: '+' for numbers and strings



- `x = 3 + 5` stores the number 8 in memory location `x`.
- `x = x + 1` increases `x` by 1.

## Side Note: '+' for numbers and strings



- `x = 3 + 5` stores the number 8 in memory location `x`.
- `x = x + 1` increases `x` by 1.
- `s = "hi" + "Mom"` stores "hiMom" in memory locations `s`.

## Side Note: '+' for numbers and strings



- `x = 3 + 5` stores the number 8 in memory location `x`.
- `x = x + 1` increases `x` by 1.
- `s = "hi" + "Mom"` stores "hiMom" in memory locations `s`.
- `s = s + "A"` adds the letter "A" to the end of the strings `s`.

# Today's Topics



- Research Survey
- For-loops
- `range()`
- Variables
- Characters
- **Strings**

# More on Strings...

From Final Exam, Fall 2017, Version 1, #1:

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

# More on Strings...

Name: \_\_\_\_\_

EmpID: \_\_\_\_\_

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.
- First, go through and write down what we know:

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.
- First, go through and write down what we know:
  - ▶ There are 3 `print()`.

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.
- First, go through and write down what we know:
  - ▶ There are 3 `print()`.
  - ▶ Output will have at least:

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.
- First, go through and write down what we know:
  - ▶ There are 3 `print()`.
  - ▶ Output will have at least:  
`There are ??? fun days in a week`

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.
- First, go through and write down what we know:
  - ▶ There are 3 `print()`.
  - ▶ Output will have at least:

There are ??? fun days in a week  
Two of them are ???

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.
- First, go through and write down what we know:
  - ▶ There are 3 `print()`.
  - ▶ Output will have at least:

There are ??? fun days in a week  
Two of them are ???  
My favorite ??? is Saturday.

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Some we have seen before, some we haven't.
- Don't leave it blank— write what you know & puzzle out as much as possible.
- First, go through and write down what we know:
  - ▶ There are 3 `print()`.
  - ▶ Output will have at least:

There are ??? fun days in a week  
Two of them are ???  
My favorite ??? is Saturday.
- Will get 1/3 to 1/2 points for writing down the basic structure.

## More on Strings: String Methods

```
s = "FridaysSaturdaysSundays"  
num = s.count("s")
```

- The first line creates a variable, called `s`, that stores the string:  
"FridaysSaturdaysSundays"

## More on Strings: String Methods

```
s = "FridaysSaturdaysSundays"  
num = s.count("s")
```

- The first line creates a variable, called `s`, that stores the string:  
"FridaysSaturdaysSundays"
- There are many useful functions for strings (more in Lab 2).

## More on Strings: String Methods

```
s = "FridaysSaturdaysSundays"  
num = s.count("s")
```

- The first line creates a variable, called `s`, that stores the string:  
`"FridaysSaturdaysSundays"`
- There are many useful functions for strings (more in Lab 2).
- `s.count(x)` will count the number of times the pattern, `x`, appears in `s`.

## More on Strings: String Methods

```
s = "FridaysSaturdaysSundays"  
num = s.count("s")
```

- The first line creates a variable, called `s`, that stores the string:  
"FridaysSaturdaysSundays"
- There are many useful functions for strings (more in Lab 2).
- `s.count(x)` will count the number of times the pattern, `x`, appears in `s`.
  - ▶ `s.count("s")` counts the number of lower case `s` that occurs.

## More on Strings: String Methods

```
s = "FridaysSaturdaysSundays"  
num = s.count("s")
```

- The first line creates a variable, called `s`, that stores the string: "FridaysSaturdaysSundays"
- There are many useful functions for strings (more in Lab 2).
- `s.count(x)` will count the number of times the pattern, `x`, appears in `s`.
  - ▶ `s.count("s")` counts the number of lower case `s` that occurs.
  - ▶ `num = s.count("s")` stores the result in the variable `num`, for later.

## More on Strings: String Methods

```
s = "FridaysSaturdaysSundays"  
num = s.count("s")
```

- The first line creates a variable, called `s`, that stores the string:  
"FridaysSaturdaysSundays"
- There are many useful functions for strings (more in Lab 2).
- `s.count(x)` will count the number of times the pattern, `x`, appears in `s`.
  - ▶ `s.count("s")` counts the number of lower case `s` that occurs.
  - ▶ `num = s.count("s")` stores the result in the variable `num`, for later.
  - ▶ What would `print(s.count("sS"))` output?

## More on Strings: String Methods

```
s = "FridaysSaturdaysSundays"  
num = s.count("s")
```

- The first line creates a variable, called `s`, that stores the string:  
`"FridaysSaturdaysSundays"`
- There are many useful functions for strings (more in Lab 2).
- `s.count(x)` will count the number of times the pattern, `x`, appears in `s`.
  - ▶ `s.count("s")` counts the number of lower case `s` that occurs.
  - ▶ `num = s.count("s")` stores the result in the variable `num`, for later.
  - ▶ What would `print(s.count("sS"))` output?
  - ▶ What about:  
`mess = "10 20 21 9 101 35"  
mults = mess.count("0 ")  
print(mults)`

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Don't leave it blank— write what you know & puzzle out as much as possible:

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Don't leave it blank— write what you know & puzzle out as much as possible:

There are 3 fun days in a week

Two of them are ???

My favorite ??? is Saturday.

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)

## More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d  | a  | y  | s  |

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[0]` is

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[0]` is 'F'.

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[1]` is

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[1]` is 'r'.

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[-1]` is

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[-1]` is 's'.

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[3:6]` is

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[3:6]` is ‘day’.

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[:3]` is

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[:3]` is 'Fri'.

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[:-1]` is

# More on Strings: Indexing & Substrings

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- Strings are made up of individual characters (letters, numbers, etc.)
- Useful to be able to refer to pieces of a string, either an individual location or a “substring” of the string.

|   |   |   |   |   |   |   |   |   |     |    |    |    |     |    |    |    |    |
|---|---|---|---|---|---|---|---|---|-----|----|----|----|-----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | 16 | 17 | 18 | 19  | 20 | 21 | 22 |    |
| F | r | i | d | a | y | s | S | a | ... | S  | u  | n  | d   | a  | y  | s  |    |
|   |   |   |   |   |   |   |   |   |     |    |    |    | ... | -4 | -3 | -2 | -1 |

- `s[:-1]` is 'FridaysSaturdaysSunday'.  
*(no trailing 's' at the end)*

# More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.

## More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.
- Cross out the delimiter, and the remaining items are the list.

## More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.
- Cross out the delimiter, and the remaining items are the list.

"Friday~~X~~Saturday~~X~~Sunday"

## More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.
- Cross out the delimiter, and the remaining items are the list.

```
"FridayXSaturdayXSunday"  
days = ['Friday', 'Saturday', 'Sunday']
```

## More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.
- Cross out the delimiter, and the remaining items are the list.

```
"FridayXSaturdayXSunday"  
days = ['Friday', 'Saturday', 'Sunday']
```

- Different delimiters give different lists:

## More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.
- Cross out the delimiter, and the remaining items are the list.

```
"FridayXSaturdayXSunday"  
days = ['Friday', 'Saturday', 'Sunday']
```

- Different delimiters give different lists:

```
days = s[:-1].split("day")
```

## More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.
- Cross out the delimiter, and the remaining items are the list.

```
"FridayXXXXSaturdayXXXXSunday"  
days = ['Friday', 'Saturday', 'Sunday']
```

- Different delimiters give different lists:

```
days = s[:-1].split("day")
```

```
"FridayXXXXsaturdayXXXXsunday"
```

## More on Strings: Splits

```
s = "FridaysSaturdaysSundays"  
days = s[:-1].split("s")
```

- `split()` divides a string into a list.
- Cross out the delimiter, and the remaining items are the list.

```
"FridayXSaturdayXSunday"  
days = ['Friday', 'Saturday', 'Sunday']
```

- Different delimiters give different lists:

```
days = s[:-1].split("day")
```

```
"FriXXXyXXXSaturXXXyXXXSunXXXy"  
days = ['Fri', 'sSatur', 'sSun']
```

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Don't leave it blank— write what you know & puzzle out as much as possible:

# More on Strings...

Name:

EmpID:

CSci 127 Final, V1, F17

1. (a) What will the following Python code print:

```
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

- Don't leave it blank— write what you know & puzzle out as much as possible:

There are 3 fun days in a week  
Two of them are Friday Sunday  
My favorite ??? is Saturday.

# Lecture Slip

- What is printed? Write your answer for each in the output box.

```
months = ["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
#Indices:   0     1     2     3     4     5     6     7     8     9     10    11
#Or:          ....      -3     -2     -1
```

Output:

```
half = months[6]
print(half.upper())
```

```
print(months[-1].lower())
```

```
start = 9
print(months[start-1])
```

```
term = 3
print(months[(start+term-1)%12])
```

# Recap

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

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print()  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

# Recap

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

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print()  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

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

## ► For-loops

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print()  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

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

- ▶ For-loops
- ▶ `range()`

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

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

- ▶ For-loops
- ▶ `range()`
- ▶ Variables: ints and strings

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

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

- ▶ For-loops
- ▶ `range()`
- ▶ Variables: ints and strings
- ▶ Some arithmetic

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

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

- ▶ For-loops
- ▶ `range()`
- ▶ Variables: ints and strings
- ▶ Some arithmetic
- ▶ String concatenation

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

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

- ▶ For-loops
- ▶ `range()`
- ▶ Variables: ints and strings
- ▶ Some arithmetic
- ▶ String concatenation
- ▶ Functions: `ord()` and `char()`

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13     print('How lucky we are to be alive!')
```

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

- ▶ For-loops
- ▶ `range()`
- ▶ Variables: ints and strings
- ▶ Some arithmetic
- ▶ String concatenation
- ▶ Functions: `ord()` and `char()`
- ▶ String Manipulation

# Recap

```
1 #Predict what will be printed:  
2 for i in range(4):  
3     print('The world turned upside down')  
4 for j in [0,1,2,3,4,5]:  
5     print(j)  
6 for count in range(6):  
7     print(count)  
8 for color in ['red', 'green', 'blue']:  
9     print(color)  
10 for i in range(2):  
11     for j in range(2):  
12         print('Look around,')  
13 print('How lucky we are to be alive!')
```

- On lecture slip, write down a topic you wish we had spent more time (and why).
- In Python, we introduced:
  - ▶ For-loops
  - ▶ `range()`
  - ▶ Variables: ints and strings
  - ▶ Some arithmetic
  - ▶ String concatenation
  - ▶ Functions: `ord()` and `char()`
  - ▶ String Manipulation
- Pass your lecture slips to the end of the rows for the UTA's 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 Spring 2018, Mock Exam.

# Writing Boards



- Return writing boards as you leave...