# C Programming

Yipeng Huang

Rutgers University

January 26, 2021

# Table of contents

# Announcements

## Short quiz

- ▶ Graded. Please check if you did not get grade back.
- ▶ We'll continue with this rhythm of having quiz due by Monday night.
- ▶ Goal is to make sure you are up to date with class, so you are ready to use lecture and recitations.

## Programming assignment

- ▶ Due in 16 days: 11:59pm Thursday, February 11.
- ▶ You should already have a programming environment set up for PA1
- ▶ Use Piazza to ask and respond to questions.
- ▶ By end of this week, you will have everything you need for at least part 1, goldbach, and part 2, maximum.

# Recitation sections

## Starting this week, and in most weeks

- ▶ You can attend all, any, or none of the recitations for reviewing the material and help on the programming assignments.
- ▶ Marie and Abhinav have posted recitation slides with guides for PA1.
- ▶ Abhinav and Prince have recitations later today.

## New section, Section 09

- ▶ TA Azita Nouri
- ▶ Mondays 1:00-1:55 PM ET
- ▶ Indicate on short quiz if you would like to switch to this section.

# See one, do one, teach one

# See one, do one, teach one

To practice programming, you can...

- ▶ Read about it from a book (e.g., Modern C).
- ▶ Listen about it from someone talking about concepts (e.g., lecture slides in Supplementary Reading).
- ▶ Watch someone else program.
- ▶ Practice it.
- ▶ Explain / teach it to a friend.

# Table of contents

# Including headers

- `#include <stdio.h>`
- `#include <stdlib.h>`
- `#include <stdbool.h>`

# Function prototypes

Allows implementation in different files. Or in libraries.

# First encounter with pointers
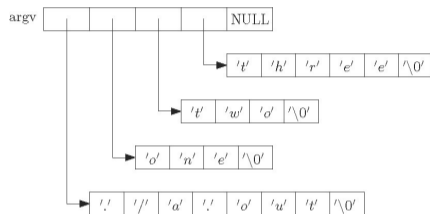
## What is `char* argv[]`



Figure: Image credit: `http://www.csc.villanova.edu/~mdamian`

## In C, Strings, `char*`, and `char[]` are all the same

▶ `char greeting[6] = {'H','e','l','l','o','\0'};`

▶ `char greeting[] = "Hello";`

# Control flow

- Conditionals
- Loops
- `break;`
- `continue;`

# Printing to command line

The format string in `printf(char* format, args)`

- `%d`: integer
- `%ld`: long integer
- `%s`: string
- `%c`: character
- `%f`: float

# Compiling and running your program

How does a program end up on your computer?

## How a Makefile works

- `$@`: target file name
- `$<`: first prerequisite
- `$^`: all prerequisites

# Table of contents

# Opening a file

The mode in `FILE *fopen(const char *filename, const char *mode)`

- `"r"`: read from the file
- `"w"`: write, starting at the beginning of the file
- `"a"`: write, starting at the end of the file (append)

# Reading characters from a file

- `int fgetc(FILE *stream)`
- `char *fgets(char *str, int n, FILE *stream)`
- `int fscanf(FILE *stream, const char *format, ...)`

# Table of contents