

This is CS50

Fall 2015

ohai!

Brian Scassellati (i.e Scaz)

Professor of Computer Science

Jason Hirschhorn

Lecturer in Computer Science

Andi Peng

Head TA for CS50

What's coming up...

- office hours (Mon, Tue, Wed, Thu)
- sections (Sun, Mon, Tue)
 - sectioning happens this week
- Problem Set 1 due Thu 9/17 at noon (or Fri 9/18 with late day)
- CS50 Lunches (Fri)
- ...

cs50.yale.edu

cs50.harvard.edu

Resources

- lectures
 - notes, slides, source code, videos
- sections
 - slides, source code, videos
- shorts
- office hours
- CS50 Study <<https://study.cs50.net/>>
- Reference50 <<https://reference.cs50.net/>>
- ...

Agenda

- data types
- operators
- libraries
- printf
- conditionals
- loops
- CS50 IDE
- terminal commands
- demos, practice

Data types

data type	sizeof();
char	1
int	4
float	4
long	8
double	8
string	?

Operators

add	+	$x + y$
subtract	-	$x - y$
multiply	*	$x * y$
divide	/	x / y
modulo	%	$x \% y$

equal	==	$x == y$
not equal	!=	$x != y$
less than	<	$x < y$
less than or equal to	<=	$x <= y$
greater than	>	$x > y$
greater than or equal to	>=	$x >= y$
logical AND	&&	$(x > y) \&\& (a < b)$
logical OR		$(x > y) (a < 2b)$

Libraries

- a set of functions
 - code that has already been written by somebody else, but usable by you
 - `#include`
- C standard libraries
 - input/output (`stdio.h`)
 - string (`string.h`)
 - math (`math.h`)
- CS50 Library
 - `cs50.h`

printf()

```
#include <stdio.h>
```

```
printf("Hello, my name is Andi, and I am 20 years old.\n");
```

Hello, my name is Andi, and I am 20 years old.

printf()

```
#include <stdio.h>
```

```
#include <cs50.h>
```

```
int age = GetInt();
```

```
printf("Hello, my name is Andi, and I am %i years old.\n", age);
```

Hello, my name is Andi, and I am 20 years old.

printf()

```
#include <stdio.h>
```

```
#include <cs50.h>
```

```
int age = GetInt();
```

```
string name = GetString();
```

```
printf("Hello, my name is %s, and I am %i years old.\n", name, age);
```

Hello, my name is Andi, and I am years 20 old.

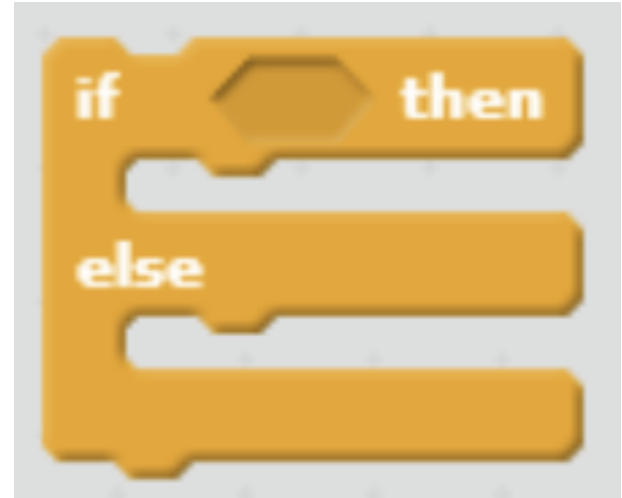
Conditionals

```
if (condition)
{
    // do if condition is true
}
```



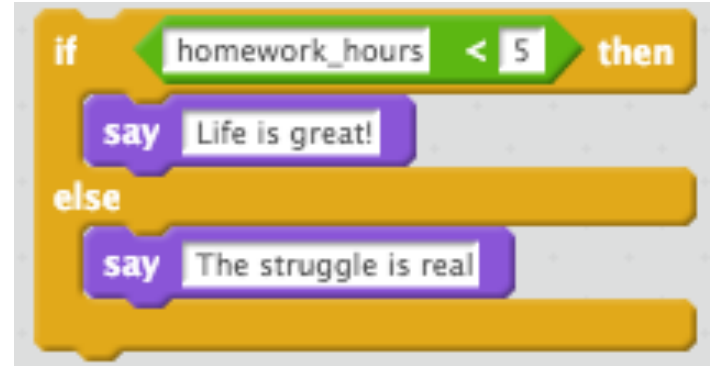
Conditionals

```
if (condition)
{
    // do if condition is true
}
else
{
    // do if condition is false
}
```



Conditionals

```
if (homework_hours < 5)
{
    printf("Life is great!\n");
}
else
{
    printf("The struggle is real\n");
}
```



Conditionals (what's wrong?)

```
int grade = GetInt();

if (grade >= 90)
{
    printf("You got an A!\n");
}
if (grade >= 80)
{
    printf("You got a B!\n");
}
if (grade >= 70)
{
    printf("You got a C!\n");
}
...
```

Conditionals

```
int grade = GetInt();

if (grade >= 90)
{
    printf("You got an A!\n");
}
else if (grade >= 80)
{
    printf("You got a B!\n");
}
else if (grade >= 70)
{
    printf("You got a C!\n");
}
...
```


Conditionals: Switch Statement

```
switch (n)
{
    case constant1:
        // do if n is equal to constant1
        break;
    case constant2:
        // do if n is equal to constant1
        break;
    ...
    default:
        // do if n is not equal to any cases above
        break;
}
```

Conditionals: Switch Statement

```
int year_founded = GetInt();

switch (year_founded)
{
    case 1636:
        printf("Shouldn't you be at that school up north?\n");
        break;
    case 1701:
        printf("Welcome to Yale!\n");
        break;
    default:
        printf("Hello, Internet!\n");
        break;
}
```

Conditionals: Ternary Operator

(condition ?) <do if condition is true> : <do if condition is false>;

```
int n = GetInt();
```

```
string s = (n > 100) ? "high" : "low";
```

Conditionals: Ternary Operator

```
string s;
```

```
if (n < 100)
```

```
{
```

```
    s = "low";
```

```
}
```

```
else
```

```
{
```

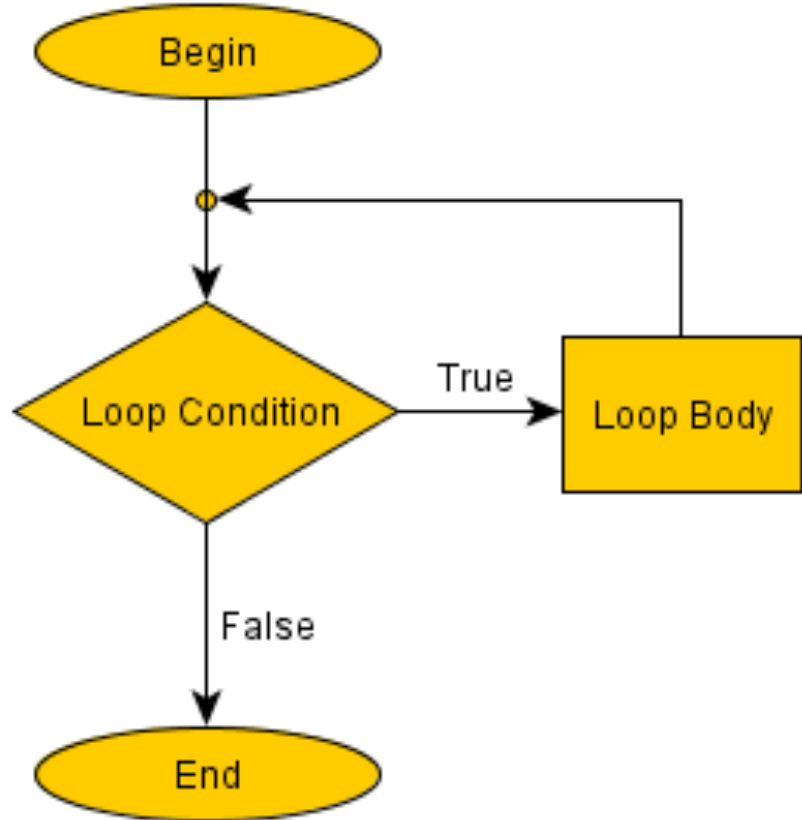
```
    s = "high";
```

```
}
```

```
string s = (n < 100) ? "low" : "high";
```

While Loops

```
while (condition)
{
    // do this repeatedly
}
```



While Loops

```
while (true)
{
    printf("I love SAJ!\n");
}
```

Do While Loops

```
do
{
    // do this repeatedly
}
while (condition);
```

Do While Loops

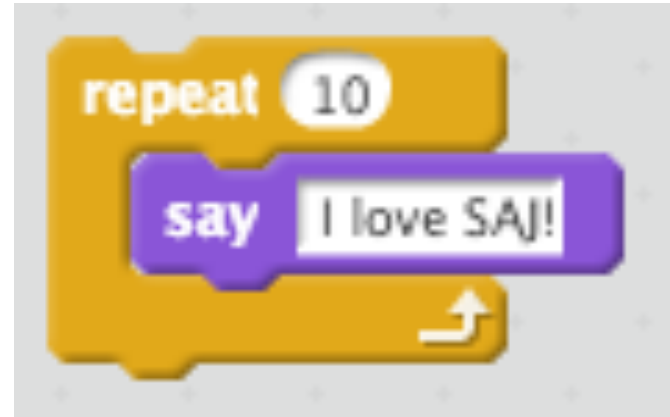
```
int age;  
  
do  
{  
    printf("What is your age?: ");  
    age = GetInt();  
}  
while (age < 1);
```


For Loops

```
for (initialization; condition; update)
{
    // do this
}
```

For Loops

```
for (int i = 0; i < 10; i++)  
{  
    printf("I love SAJ!\n");  
}
```



For Loops (Pseudocode)

```
for (int i = 0; i < 10; i++)  
{  
    printf("I love SAJ!\n");  
}
```

1. create variable *i* and initialize it to 0
2. check is $i < 10$?
3. if so, print the line
4. increment *i*
5. go back to beginning of loop, check is $i < 10$?
6. eventually, if *i* is not < 10 , exit loop without printing the line

Loops

```
int i = 0;

while (i < 10)
{
    printf("I love SAJ!\n");
    i = i + 1;
}
```

For Loops

```
// prints a table of x's

for (int row = 0; row < 3; row++)
{
    for (int column = 0; column < 3; column++)
    {
        printf("x");
    }
    printf("\n");
}
```

Problem Set 1: C

- `water.c`
 - prompt user for input
- `mario.c`
 - nested for loop (for loop within a for loop!)
- `greedy.c`
 - conditionals
 - be careful of floating-point values! (covered in lecture and walkthrough)



Style

- give your variables meaningful names
 - in for loops, single character variables are fine (i, j, k, etc.)
- consistent initialization
 - `int scaz_age, int andi_age = 20, int jason_age;`
 - example of bad style
- consistent curly braces
- consistent spacing
- CS50 Style Guide

consistency is key!

CS50 IDE

The screenshot displays the CS50 IDE interface. The top menu bar includes 'CS50 IDE', 'File', 'Edit', 'Find', 'View', 'Goto', 'Window', 'Support', and a 'Debug' button. On the right side of the top bar, there is a user icon, a help icon, the number '30', the text 'ide50-bs276.c9.io', and status indicators for 'MEMORY', 'CPU', and 'DISK'. A 'Share' button with a gear icon is also present.

The left sidebar, labeled 'Workspace', shows a file tree with the following structure:

- ide50
 - IOCCC
 - SuperSection
 - bunny.c (selected)
 - README.md

The main editor area displays the code for 'bunny.c' with line numbers 1 through 14. The code is as follows:

```
1 #include<stdio.h>
2
3 int main(void)
4 {
5     // print some number of dots at the start
6     printf(".....");
7
8     // print the bunny
9     printf("O.O");
10
11    // print some number of dots at the end
12    printf(".....");
13
14 }
```

At the bottom, a 'Terminal' window is open, showing the command prompt 'bs276@ide50:~/workspace/SuperSection \$' with a cursor.

On the far right, a vertical sidebar contains the following options: 'Collaborate', 'Outline', and 'Debugger'.

Terminal commands

- **ls**
 - list files / directories
- **ls -l**
 - list files / directories with additional information (incl. permissions)
- **cd ..**
 - return to previous directory
- **cd [PATH]**
 - enter directory at [PATH]
- **mkdir [DIRECTORY_NAME]**
 - create a directory named [DIRECTORY_NAME]

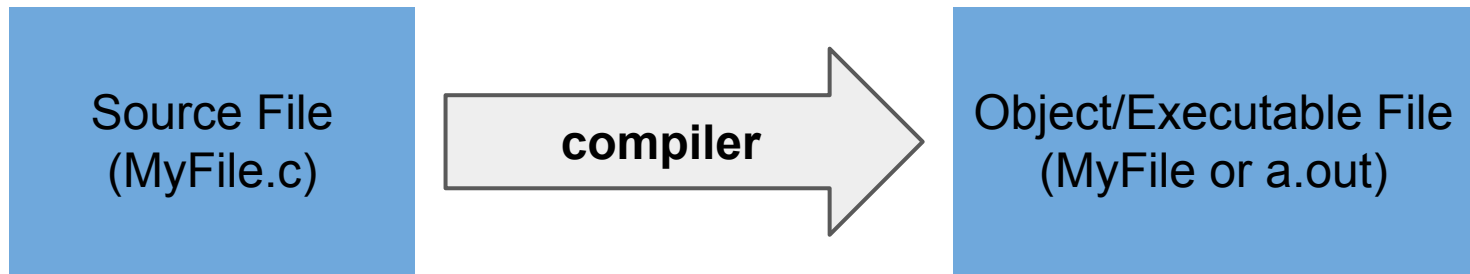
Terminal commands

- **cp** [FILE_NAME] [NEW_FILE_NAME]
 - copy file named [FILE_NAME] as a new file named [NEW_FILE_NAME]
- **mv** [FILE_NAME] [DIRECTORY_PATH]
 - moves text file named [FILE_NAME] into directory at [DIRECTORY_PATH]
- **rm** [FILE_NAME]
 - remove [FILE_NAME]
- **rm -f** [FILE_NAME]
 - remove [FILE_NAME] without asking for confirmation. **BEWARE!**
- **rm -r** [DIRECTORY_PATH]
 - remove everything inside and including a directory at [DIRECTORY_PATH]. **BEWARE!**
- **rmdir** [DIRECTORY_PATH]
 - remove an empty directory at [DIRECTORY_PATH]

Terminal commands

- **clear**
 - clear terminal screen
- **more [FILE_NAME]**
 - display the text of a file [FILE_NAME] in the terminal screen
- **CTRL+c**
 - terminate program (if infinite loop)
- **up & down arrow keys**
 - scroll through previous commands
- **tab key**
 - autocomplete

Compilation



```
bs276@ide50:~/workspace/SuperSection $ make MyFile
```

```
clang -ggdb3 -O0 -std=c99 -Wall -Werror MyFile.c -lcs50 -lm -o MyFile
```

Cell animation using ASCII art

....().().....

.().()...... ..().()...... ...().()......().()......().()......

.....().().....().()....().()...().()..().().()

Cell animation using ASCII art

... () . () ...

■ (((((X))))))

$\cdot () \cdot () \dots\dots\dots \cdot () \cdot () \dots\dots\dots \cdot \cdot () \cdot () \dots\dots\dots \cdot \cdot \cdot () \cdot () \dots\dots\dots \cdot \cdot \cdot \cdot () \cdot () \dots\dots\dots$

..... $()()$ $()()$ $()()$ $()()$ $()()$

The 2012 International Obfuscated C Code Contest

<http://www.ioccc.org>

Honorable mention - Most complex ASCII fluid

Yusuke Endoh