CS 161
Introduction to CS I
Lecture 5

- More decision making (conditional statements)
- Assignment 2 Design
- How do we handle invalid input?
- What should we test?
- How can we generate random numbers?

Must-haves for comments and style

Required for Assignments 2-5

- File header
- Comments:
  - Block (multi-line) comments/headers: `/** comments **/`
  - Single line: `/* comment */` (preferred – why?) or `// comment`
- Max line length: 80 characters
- Vertical space (blank lines) between code sections
- Horizontal space between values/variables/operators in expressions:
  - `y = x + 3;` not `y=x+3;` or `y = x+3;` or `y=x + 3;`
- Indentation to indicate flow of execution
  - Especially important for if/then, switch, loops, and any nested commands
- Comment next to `else` to indicate what is true at that point

http://classes.engr.oregonstate.edu/eecs/winter2020/cs161-020/assignments/cs161-style-guidelines.pdf

If/then/else example 1

- What will the output be if I run this code?

```
char letter = 'k';

if (letter == 'k')
  {
    cout << "The letter is k." << endl;
  }

cout << "The letter is not k." << endl;
```

Prints no matter what

If/then/else example 1

- What will the output be if I run this code?

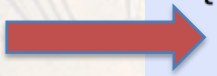
```
char letter = 'k';

if (letter == 'k')
  {
    cout << "The letter is k." << endl;
  }
else
  {
    cout << "The letter is not k." << endl;
  }
```

Indentation

Indentation

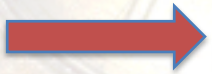
Variable scope

- Curly brackets {} restrict visibility of variable (with **block**)



```
/* If user is under 25,
 * they must pay an extra fee */
if (age < 25)
  {
    float fee = 3.95;
    cout << "You must pay an extra fee of " << fee
         << "to rent this car." << endl;
  }
else
  {
    cout << "No extra fee for you!" << endl;
  }
cout << "The underage driver fee is " << fee << endl;
```

"fee" not found!



Oregon State University
College of Engineering

Supercharge your Boolean expressions

Logical operators

- `&&` and
- `||` or
- `!` not

Operator precedence

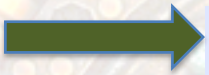
++ --
!
* / %
+ -
< <= > >=
&&
||
== !=
=

- Examples
  - `true && false`
  - `true || false`
  - `!0`
  - `2 || 3 > 4`
  - `2 && 3 > 4`
  - `!(3 > 4)`
  - `!2 + 1 > 4`

If/then/else example 2

- What will the output be if I run this code?

Initialize Boolean variable



```
bool feeling_good = true;

if (feeling_good)
  {
    cout << "Today is a great day!" << endl;
  }
else /* feeling_good is false */
  {
    cout << "Today is not going well." << endl;
  }
```

If/then/else example 3

- Checking user input

```
char user_feeling = 'n';

cout << "Are you feeling good today? (y/n) " << endl;
cin >> user_feeling; /* reads in a single character */

if ((user_feeling == 'y') ||
    (user_feeling == 'Y'))
  {
    cout << "Today is a great day!" << endl;
  }
else if ((user_feeling == 'n') ||
         (user_feeling == 'N'))
  {
    cout << "Today is not going well." << endl;
  }
else /* user_feeling is not 'y'/'Y' or 'n'/'N' */
  {
    cout << "Invalid choice." << endl;
  }
```

Allow multiple variations

Describe what must be true at this point

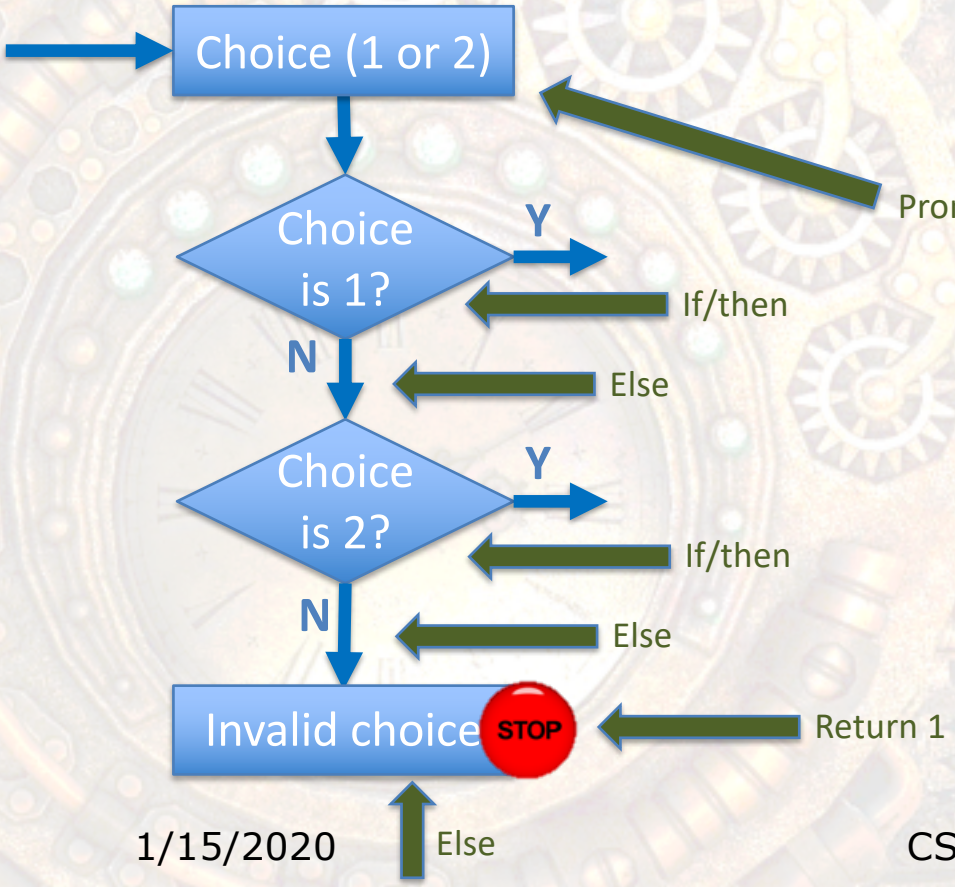
Oregon State University
College of Engineering

Assignment 2 – Text Adventure

- + Design
  - Graded by course staff
- + Peer review
  - Randomly assigned
  - Due following Weds.
  - Valuable for refining your assignment
  - Provide useful feedback to get credit

```
wkiri@madrone demos % ./assign2_game
Welcome to Mythago Wood!
 You have 0 points.

You are in a forest and see a cottage.  Do you:
(1) Knock on the door, or
(2) Keep walking?
```

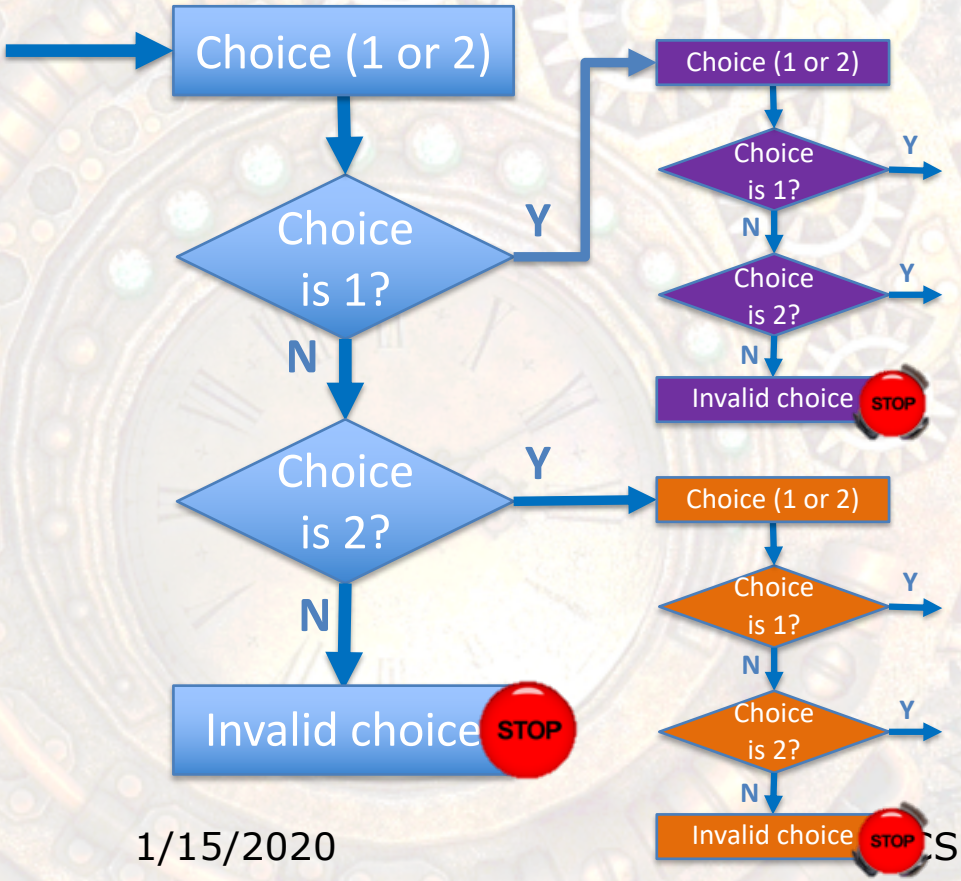
Example flowchart: First choice



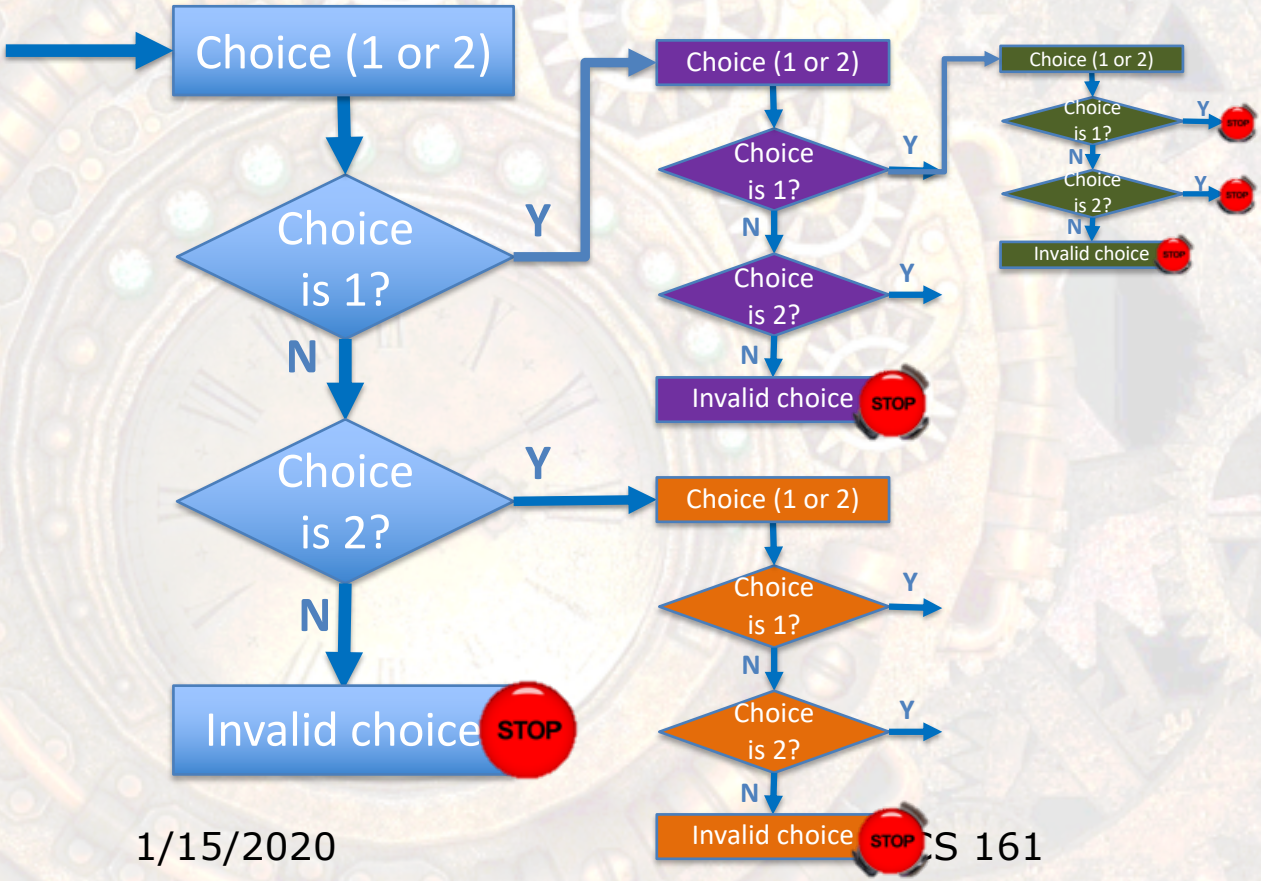
What C++ code
does each part correspond to?

Oregon State University
College of Engineering

Example flowchart: Second choice



Example flowchart: Third choice



Assignment 2 requirements

- You must read user input
- You can have more than 2 options for each choice
- Some paths can end before 3 choices are made
  - But you must have 2 endings that require 3 choices

Testing… before you write the program

- Your goal: protect against user mistakes (intentional or not)
- **Test case**: Test setting, user input, and expected output
  - You must have: Good, bad, and “edge” (tricky) case
- Brainstorm:
  - What I would do
  - What my 10-year-old niece might do
  - What my practical joker friend might do
  - What my cat might do
  - Get creative!

Exercise: Test cases

- Problem statement: Read in number and check if digits are all equal

Test case type	Test setting	User input	Expected result
Good			
Bad			
Edge			

Exercise: Test cases

- Problem statement: Read in number and check if digits are all equal

Test case type	Test setting	User input	Expected result
Good	Read in number	222	Yes
Bad	Read in number	Hello	Print error and quit
Edge	Read in number	7	Yes

Checking user input inside the program

- **Use the "else" clause**
  - If the user input isn't anything you were expecting, you can deal with it here
    - `return` from `main()` ends the program
- **Add a comment** indicating what the else clause is handling (`age <= 0`)

```
/* If user is under 25,
 * they must pay an extra fee */
if (0 < age && age < 25)
  {
    cout << "You must pay an extra fee "
         << "to rent this car." << endl;
  }
else if (age >= 25)
  {
    cout << "No extra fee for you!" << endl;
  }
else /* age <= 0 */
  {
    cout << "Age must be positive." << endl;
    return 1;
  }

return 0;
```

Indicates an error

Indicates success

Decision making with random chance

- Now the program can make decisions – but they are always deterministic.
- Life is more complicated than that!
- We can generate random numbers and combine them with if/then/else

Decision making with random chance

```
int dice_roll = 0;

/* Roll the dice */
dice_roll = rand()%6 + 1;
cout << "You rolled " << dice_roll << endl;
if (dice_roll < 3)
  {
    cout << "You lose..." << endl;
  }
else /* Rolled 3 or higher */
  {
    cout << "You win!" << endl;
  }
```

What vocabulary did we learn today?

- Variable scope
- Logical operators: &&, ||, !
  - | is a “pipe”
- Flowchart
- Test case
- rand()

What ideas and skills did we learn today?

- Programming style requirements
- The importance of “else”
- Flowcharts for program design
- Test cases for program design
- Checking user input
- Generating random numbers

Week 2 continues!

- ❑ Attend lab (laptop required)
- ❑ Read **Rao pp. 40 (Booleans) and 41 (characters)**
  Rao pp. 122-125 (switch)
  Random numbers:
  http://www.cplusplus.com/reference/cstdlib/rand/
- ❑ Look at **Assignment 2** and plan your **design** (due **Sunday, Jan. 19**)

See you on Friday!