



CS 161

Intro to CS I

Conditions



Odds and Ends

- Assignment 2 and Back Evals due Sunday
- Questions?



Additional Operators

- Common operation: fetch/store same variable
 - `var=var + 2; //increment variable contents`
 - `var=var * 2; //double variable contents`
 - operator/assignment combination (all ops supported):
 - `var += 2;`
 - `var *= 2;`
- Pre/Post increment/decrement: `++` and `--`
 - Example: `age++` vs. `++age`

Demo...



Oregon State University
College of Engineering



Decisions in Life

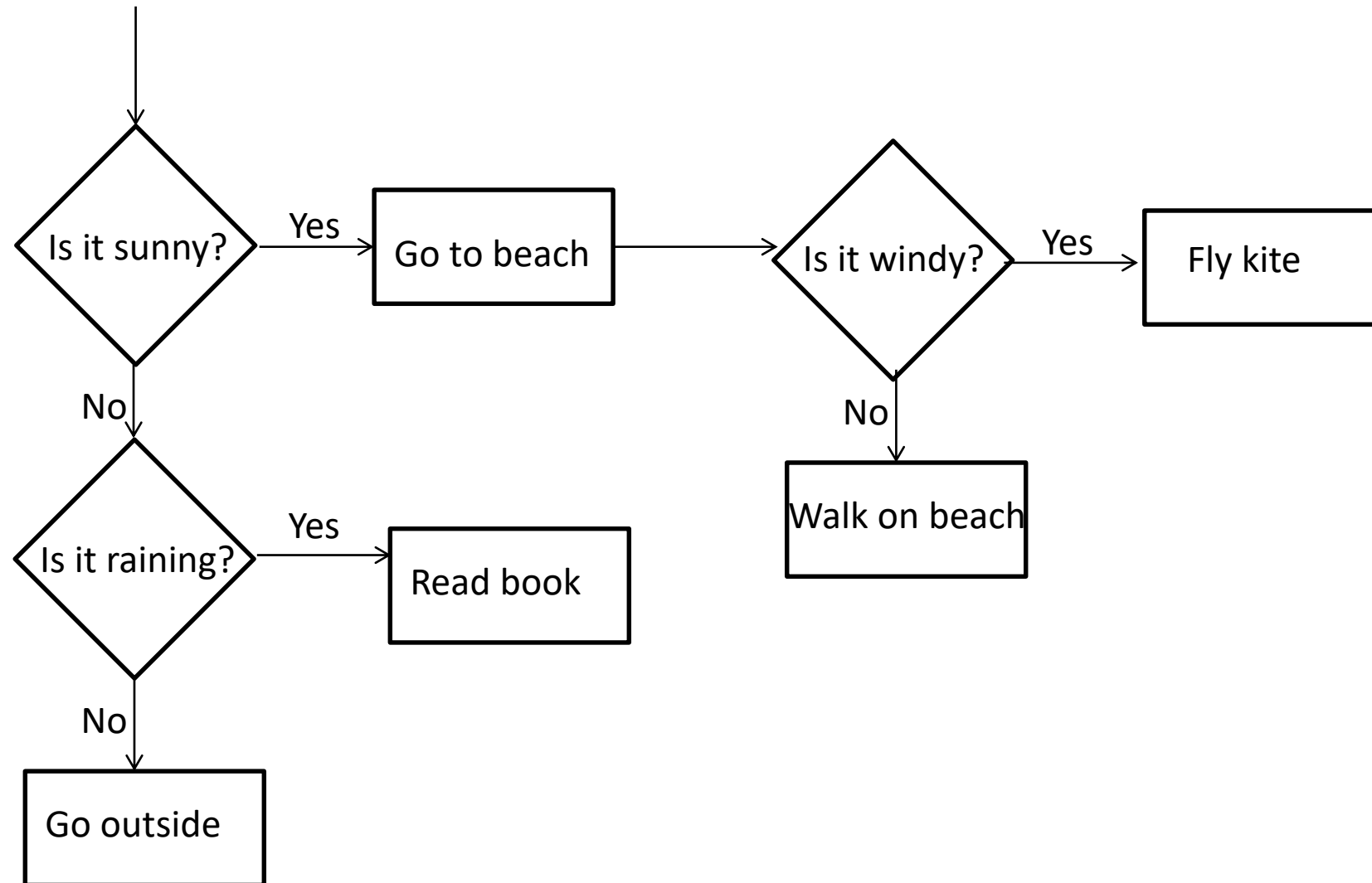
- What is a decision?
- When do we make decisions?
- How do we make decisions?
 - If it is sunny today
 - then I'll go to the beach and fly a kite
 - Else if it is raining today
 - then I'll stay inside and read a book
 - Else if it is snowing
 - then I'll go to the mountains to ski



Decisions within Decisions

- What happens if there is no wind at the beach?
- How does this change our decisions?
 - If it is sunny today
 - then I'll go to the beach
 - if it is windy at the beach
 - then I'll fly a kite
 - if it is not windy at the beach
 - then I'll walk on the shore

Flow chart for decisions





Decisions in our programs

- Use an if/else

```
if (<expression>) {  
    <statement>;  
    ...  
    <statement>;  
}  
else {  
    <statement>;  
    ...  
}
```




What is the <expression>?

Could be a relational expression:

<expression> <relational op> <expression>

- Relational Ops

== - equal to

!= - not equal to

< - less than

> - greater than

<= - less than or equal to

>= - greater than or equal to



C++ If/Else Syntax...

```
if( x > y) {  
    std::cout << "X is greater than Y";  
}  
else {  
    std::cout << "X is less than Y";  
}
```

- When does this logic fail?



C++ If/Else...

```
if( x > y) {  
    std::cout << "X is greater than Y";  
}  
else if( x < y) {  
    std::cout << "X is less than Y";  
}  
else {  
    std::cout << "X is equal to Y";  
}
```

What are the curly braces for?



```
if( x > y)
    std::cout << "X is greater than Y";
else if( x < y)
    std::cout << "X is less than Y";
else
    std::cout << "X is equal to Y";
```



What if we are testing for ==?

```
if( x == 0) {  
    std::cout << "X is zero";  
}  
else if( x == 1) {  
    std::cout << "X is one";  
}  
else if( x == 2) {  
    std::cout << "X is two";  
}  
else {  
    std::cout << "You have entered an invalid number!";  
}
```

Demo...



Oregon State University
College of Engineering



Logical Operators

- AND: `if((1 > 2) && (2 < 5))`
- OR: `if((1 > 2) || (2 < 5))`
- NOT: `if(!(1 > 2) && (2 < 5))`
- Precedence of Operators:
http://en.cppreference.com/w/cpp/language/operator_precedence

Demo...



Oregon State University
College of Engineering



We can use a switch...

```
switch( <expression> ) {  
    case <const-expr> :  
        <statement>;  
    ...  
    case <const-expr> :  
        <statement>;  
    ...  
    default:  
        <statement>;  
    ...  
}
```



C++ Switch Example

```
switch( x ) {  
    case 0:  
        std::cout <<"X is zero";  
        break;  
    case 1:  
        std::cout <<"X is one";  
        break;  
    case 2:  
        std::cout <<"X is two";  
        break;  
    default:  
        std::cout <<"You have entered an invalid number!";  
}
```



C++ Switch Example

```
switch( x ) {  
    case 0:  
    case 1:  
        std::cout << "X is zero or one";  
        break;  
    case 2:  
        std::cout << "X is two";  
        break;  
    default:  
        std::cout << "You have entered an invalid number!";  
}
```

Demo...



Oregon State University
College of Engineering