CS 161
Intro to CS I

User Input and Conditional Statements
Odds and Ends...

• Finish Reading Chap. 2
• Assignment 2 and Exercise 2 posted!!!
• Assignment 1 demo through Tuesday, 1/19, for full credit.
• TA email list: cs161-ta@engr.orst.edu
• Labs: you can complete/make-up max 3 pts outside lab
• Monday labs are out of sync in beginning
Expressions

• What is an expression?
  – Set of operations producing a value
    • Combining simple values
      12 * 4 + 6 * 10 vs. ((12 * 4) + 6) * 10
Expressions cont.

• Pieces of an Expression:
  – Operators
    • Indicate operation, e.g. +, *, /, -, %
  – Operands
    • Values in the expression
  – Evaluation
    • Process of obtaining results from operations on operands
Arithmetic Operators

• Add
  34 + 23
• Subtract
  34 - 23
• Multiply
  2 * 23
• Divide
  40 / 10
• Remainder/Mod
  34 % 5
Precedence

• What is precedence?
  – Binding power of operator
  – (*, /, %) vs. (+, -)

• How do we override precedence?
  – Parenthesis!

• Examples:
  12 * 4 + 6 * 10 vs. ((12 * 4) + 6) * 10
Help with Lab 2: rand numbers

• Seed the random generator
  – Use srand(); to seed rand num generator
  – Only do once…
• User rand(); to generate number
• Example demo:
```cpp
#include <iostream>
#include <cstdlib>
#include <ctime>

using namespace std;

int main() {
    int rnum, unum;

    // only do this once to make sure rand gives different numbers each time you run the program
    srand(time(NULL));

    rnum=rand()%10; //range 0-9
    cout << rnum << endl;
    rnum=rand()%10+1; //range 1-10
    cout << rnum << endl;
    rnum=rand()%11; //range 0-10
    cout << rnum << endl;

    return 0;
}
```
How do we read into a variable in C++?

1. Declare a variable
2. Read value from user and store at variable location
3. How do we do this?

```cpp
#include <iostream>
int main() {
    int x;
    std::cin >> x;
    std::cout << x << std::endl;
    return 0;
}
```
```cpp
#include <iostream>
#include <cstdlib>
#include <ctime>

using namespace std;

int main() {
    int rnum, unum;

    //only do this once to make sure rand gives different
    //numbers each time you run the program
    srand(time(NULL));

    rnum=rand()%10; //range 0-9
    cout << rnum << endl;
    rnum=rand()%10+1; //range 1-10
    cout << rnum << endl;
    rnum=rand()%11; //range 0-10
    cout << rnum << endl;

    cout << "Enter a number 0-10: ";
    cin >> unum;
    cout << "user num: " << unum << endl;

    return 0;
}
```
Decisions in Life

• What is a decision?  [Choice]
• When do we make decisions?  [Constantly]
• 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
  - else
    - if it is not windy at the beach
      - then I’ll walk on the shore
Flow chart for decisions

- Is it sunny? Yes → Go to beach
- Is it sunny? No → Is it raining? Yes → Read book
- Is it sunny? No → Go outside
- Is it windy? Yes → Fly kite
- Is it windy? No → Walk on beach
Decisions in our programs

- Use an if/else
  
  
  \[
  \text{if (<expression>)} \\
  \text{<statement>;} \\
  \ldots \\
  \text{<statement>;} \\
  \}
  
  \text{else} \\
  \text{<statement>;} \\
  \ldots \\
  \text{<statement>;} \\
  \}
  
  O \text{ is false}
  
  any other number is \text{true}
What is the `<expression>`?

Could be a relational expression:

```plaintext
<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
Examples

• if(2 + 1)  //expression
• if(2 – 4)  //expression
• if(2 – 2)  //expression
• if(4 == 4)  //expression relational op expression
• if((2+1) == 4)  //expression relational op expression
• if(4.1 != 4)  //expression relational op expression
• if(3 <= 4)  //...
• if(4 >= 4)
• if(3.5 > 4)
• if(4 < 4)
• if(3+2*2 > 9)
• if((3+2)*2 > 9)
Logical Operators

• AND: if((1>2) && (2<5))
• OR: if((1>2) || (2<5))
• NOT: if(!(1>2) && (2<5))

• Precedence of Operators: refer to book

<table>
<thead>
<tr>
<th>T &amp; T</th>
<th>T &amp; F</th>
<th>F &amp; T</th>
<th>F &amp; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>T</th>
<th>T</th>
<th>T</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
</tbody>
</table>

Short circuit
C++ If/Else Syntax...

```cpp
int x, y;
x = 1;
y = 2;
if( x > y) {
    cout << "X is greater than Y" << endl;
}
else {
    cout << "X is less than Y" << endl;
}
```

• When does this logic fail?
C++ If/Else...

```cpp
if( x > y) {
    cout << “X is greater than Y” << endl;
}
else if( x < y) {
    cout << “X is less than Y” << endl;
}
else {  // default if none of the ifs are true.
    cout << “X is equal to Y” << endl;
}
```
What are the curly braces for?

```cpp
if( x > y)
    cout << "X is greater than Y" << endl;
else if( x < y)
    cout << "X is less than Y" << endl;
else
    cout << "X is equal to Y" << endl;
```
What if we are testing for ==?

```cpp
if( x == 0) {
    cout << "X is zero" << endl;
}
else if( x == 1) {
    cout << "X is one" << endl;
}
else if( x == 2) {
    cout << "X is two" << endl;
}
else {
    cout << "Not 0, 1, 2!!!" << endl;
}
```
```cpp
#include <iostream>
#include <cstdlib>
#include <ctime>

using namespace std;

int main() {
    int rnum, unum;

    // only do this once to make sure rand gives different
    // numbers each time you run the program
    srand(time(NULL));

    rnum=rand()%10; // range 0-9
    cout << rnum << endl;
    rnum=rand()%10+1; // range 1-10
    cout << rnum << endl;
    rnum=rand()%11; // range 0-10
    cout << rnum << endl;

    cout << "Enter a number 0-10: ";
    cin >> unum;
    if(unum<0 || unum>10){
        cout << "you idiot! I said 0-10!" << endl;
        exit(1);
    }
    cout << "user num: " << unum << endl;

    return 0;
}
```