CS 161
Intro to CS I

More Decisions/Begin Loops
Odds and Ends...

• Recitation Design for Assignment 3 is due Sunday on Canvas!!!
The while loop

```cpp
for(x=1; x <= 100; x++)
    cout << "hello world\n";
VS.
int x=1;
while(x<=100) {
    cout << "hello world\n";
    x++;
}
```
Common Mistakes

```cpp
int x=1;
while(x<=100) {
    cout << "hello world\n";
    x++;  // What if we forget this?
}
```

What if we forget this?

What if we forget this?
The do/while loop

```cpp
int x=1;
do {
    cout << "hello world\n";
    x++;
} while(x<=100);

• Difference b/w while and do/while?
Demo...
The for Loop Examples

for(x=-100; x <= 100; x++)
    cout << "hello world" << endl;
for(x=2+2; x <= 17*3; x++)
    cout << "hello world" << endl;
for(x=0; x <= 100; x++)
    cout << "hello world" << endl;
for(x=0; x < 100; x++)
    cout << "hello world" << endl;
for(x=-100; x <= -1; x++)
    cout << "hello world" << endl;
The for Loop Examples

```c++
for(x=1; x <= 1; x++) {
    cout << "hello world" << endl;
}
for(x=1; x < 1; x++) {
    cout << "hello world" << endl;
}
```

- Why is it better to use curly braces?
The for Loop Pattern

```c
for(<variable> = n; <variable> <= p; <variable>++) {
    <statement>;
    ...
}
for(<variable> = n; <variable> >= p; <variable>--) {
    <statement>;
    ...
}
```
Nested for Loops

```c++
for(x = 0; x < 10; x++) {
    for(y = 0; y < 10; y++) {
        cout << "hello world" << endl;
    }
}
```

• How many times is Hello World printed?
Reuse Variables

```cpp
for(x = 0; x < 10; x++) {
    cout << "The value of x is: " << x << endl;
}
for(x = 0; x < 10; x++) {
    cout << "The value of x is: " << x << endl;
}
```
Variables with same name

```cpp
int x;
for(x = 0; x < 10; x++) {
    for(x = 0; x < 10; x++) {
        cout << "The value of x is: " << x << endl;
    }
}
```

• What is the output from this nested loop?
Infinite Loops

```cpp
int x;
for(x = 0; x < 10; x++) {
    for(x = 0; x < 5; x++) {
        cout << "The value of x is: " << x << endl;
    }
}
```
Infinite Loops

int x, y;
for(x = 0; x < 10; x++) {
    for(y = 0; y < 5; x++) {
        cout << "The value of x is: " << x << endl;
    }
}
}
Infinite Loops

```cpp
int x, y;
for(x = 0; x < 10; x++) {
    for(y = 0; x < 5; y++) {
        cout << "The value of x is: " << x << endl;
    }
}
```
Infinite Loops

```cpp
int x;
for(x = 1; x <= 10; x++) {
    cout << "The value of x is: " << --x << endl;
}
```
Why is this good/bad?

for(int x = 0; x < 10; x++) {
    for(int y = 0; y < 10; y++) {
        cout << "hello world" << endl;
    }
}

• Where can we access x and y?
Looping Recap...

• **for loops**
  – Repeat for specific number of times
  – Example?

• **while loops**
  – Repeat while a condition is being met
  – Example?

• **do while loops**
  – Always do once, and repeat while condition is met
  – Example?
In-Class Exercise #2

• How would you determine if user entered a good positive int without using cin.clear() and cin.ignore()?
Decomposition

• Divide Problem (task) Into Subtasks
  – Procedural Decomposition
  – Examples: cooking, cleaning, etc.

• Incremental Programming
  – Iterative Enhancement (Stepwise Refinement)

• Examples: Replicating Code
Procedural Decomposition

• Functions
  – int main() {  }
  – User defined
    void draw_box() {  }

• Function Call
  – draw_box();
Procedural Decomposition

```cpp
#include <iostream>
using std::cout;

int main() {
    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
    return 0;
}

void draw_box() {  //Define function
    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
    return 0;
}
```

```cpp
#include <iostream>
using std::cout;

void draw_box();  //Declare function

int main() {  //Use function
    draw_box();
    draw_box();
    return 0;
}

void draw_box() {  //Define function
    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
    cout << "|   |\n    cout << "+--------+
}
```
Functions Calling Other Functions

```cpp
#include <iostream>
void draw_box();
void draw_top_bottom();
void draw_sides();
int main() {
    draw_box();
    return 0;
}

void draw_box() {
    draw_top_bottom();
    draw_sides();
    draw_top_bottom();
}

void draw_top_bottom() {
    std::cout << “+--------+";  
}

void draw_sides() {
    std::cout << “|       |";  
}
```
Functions

• What is a function?
  – Block of code to perform action/subroutine

• When have we seen functions already?
  – Predefined

• What is the purpose?
  – Reduce
  – Reuse
  – Readability
Generalization

• Does a function make a task more specific or more general?
  – Justification
  – Examples
Predefined Functions

- sqrt()
- pow()
- abs()
- rand()
- srand()

- What is the difference between srand() and others?
void Functions

• Doesn’t return a value
• Still has arguments/parameters

• Can we write a `void good_pos_int()`?
• Is it more useful to return a value?
Demo...