CSC180: Lecture 6

Wael Aboulsaadat

wael@cs.toronto.edu

http://portal.utoronto.ca/

Acknowledgement: These slides are partially based on the slides supplied with Prof. Savitch book: Problem Solving with C

Tutor info

- Victor Lo
- CSC180 EngSci Club Tutor
- Tutoring Sessions: Thursdays 6-8pm
- Email: the.victor.lo@gmail.com

- Please come with your questions!

Selection

Syntax for an *if-else* Statement

A Single Statement for Each Alternative:

```
if (Boolean_Expression)
    Yes_Statement
else
    No_Statement
```

A Sequence of Statements for Each Alternative:

```
if (Boolean_Expression)
{
    Yes_Statement_1
    Yes_Statement_2
      . . .
    Yes_Statement_Last
}
else
{
    No_Statement_1
    No_Statement_2
      . . .
    No_Statement_Last
}
```

Branches/Selection

- Can you
  - Write an if-else statement that outputs the word High if the value of the variable score is greater than 100 and Low if the value of score is at most 100? The variables are of type int.

  - Write an if-else statement that outputs the word Warning provided that either the value of the variable temperature is greater than or equal to 100, or the of the variable pressure is greater than or equal to 200, or both. Otherwise, the if_else sttement outputs the word OK. The variables are of type int.

Branches/Selection

- Can you
  - Write a function that takes 3 numbers

    Finds if the 3 numbers are the same, in ascending order or descending order

    Also, find if the numbers are all less than 100

Iteration

Simple Loops

- When an action must be repeated, a loop is used
- C includes several ways to create loops
  - while loops
  - for loops
  - do while loops
- We start with the while-loop

while loop

- Example:

```
int counter = 0;
while (counter < 10)
{
      printf ("Hello “) ;
      counter  = counter + 1;
 }
```

while Loop Operation

- First, the Boolean expression is evaluated
  - If false, the program skips to the line following the while loop
  - If true, the body of the loop is executed
    - During execution, some item from the boolean expression is changed
  - After executing the loop body, the boolean expression is checked again repeating the process until the expression becomes false

- A while loop might not execute at all if the Boolean expression is false on the first check

Syntax of the `while` Statement

A Loop Body with Several Statements:

```
while (Boolean_Expression )
{
      Statement_1
      Statement_2
        . . .
      Statement_Last
}
```

body

Do NOT put a semicolon here.

A Loop Body with a Single Statement:

```
while (Boolean_Expression )
      Statement
```

body

Infinite Loops

- Loops that never stop are infinite loops
- The loop body should contain a line that will eventually cause the boolean expression to become false
- Example: Print the odd numbers less than 12

```
x = 1;
while (x != 12)
{
        printf( “x is %d”, x );
        x = x + 2;
}
```

- Better to use this comparison: while (x < 12)

Loops

- Can you
  - Tell the output of this code?

```
int x = 10;
while ( x > 0)
{
      printf( “x is %d”, x );
      x = x – 3;
}
```

  - Tell the output of the previous code using the comparison x < 0 instead of x > 0?

Loops

- Can you
  - Write a function that prints the square of each of the first 100 numbers (1..99)
  - Write a function that computes the factorial of an integer
  - Write a function that sums the squares of each of the numbers in an input range