Programming for Engineers I
Lab 02
Comparison Operators and Conditional Statements

Hina Ashraf

September 12, 2012

1 Comparison Operators

The comparison operators can be used to compare two values. The values can also be variables. For example, `x == y`. This should be read as *Is x equal to y?* Most commonly used operators are:

- `==` (equals?)
- `!=` (not equals?)
- `>` (greater than?)
- `<` (less than?)
- `>=` (equals or greater than?)
- `<=` (equals or less than?)

2 Conditional Statements

The syntax for conditional statement is,

```
if (condition...)
{
    do this if condition is true...
}
else
{
    do this if condition is false...
}
```

Operators in previous section are useful for testing a certain condition. For example,

```
int x = 2;
int y = 4;
if (x == y)
{
    printf("x is equal to y");
}
else
{
    printf("x is not equal to y");
}
```

In this case the output will be `x is not equal to y`. The `else` part is not mandatory and an `if` statement can be used standalone. For example,

```
int x;
scanf ("%d", &x);
if (x > 0)
{
    printf("You have entered a positive number");
}
```

3 Exercises

Create a single solution. Exercises must be created as separate projects. Your solution must contain three different projects corresponding to each of the following exercises.

3.1 `printf()` exercise

- Triangle
- Diamond

3.2 Conditional Statements

Write a program that asks user to enter a number. Using `% operator` test if the number is even or odd. HINT: `x % y` gives the remainder when `x` is divided by `y`.

3.3 Calculator Improvement

Improve upon the calculator from Lab 01 by providing a menu and asking the user to perform the desired operation. The user can enter a number 1–4 for his choice of operation. If the user enters an invalid input display an appropriate error message.

```
Calculator Version 1.1
1. Add +
2. Subtract -
3. Multipy *
4. Divide /
Please select your desired operation:
```