EECE.2160: ECE Application Programming
Fall 2017

Lecture 10: Key Questions
September 27, 2017

1. Explain the usage and basic structure of a `while` loop.

2. **Example:** What does each of the following short programs print?

```
a. x = 7;
   while ( x < 10  )
   {
     printf("%d ",x);
     x = x + 1;
   }
```

```
b. x = 7;
   while ( x < 3  )
   {
     printf("%d ",x);
     x = x + 1;
   }
```

3. Explain how `while` loops can be used:
a. When number of iterations is dependent on a variable (flexible limit) (`while2.c`)

b. When you want to repeat an operation until a given value (sentinel) is entered (`while3.c`)

4. What is the difference between a `while` loop and a `do-while` loop?

5. Show the difference between the outputs of the loops below

```
x = 7;                                x = 7;
do {                                  while ( x < 3  )
    printf("%d",x);                   {
    x = x + 1;                            printf("%d",x);
} while ( x < 3  );                       x = x + 1;
                                      }
```

6. Recall the example for using a while loop with a sentinel value in the grade average program and show that loop written as a `do-while` loop.

7. **Example:** Write a while or do-while loop for each of the following tasks:
a. Print all multiples of 3 between 0 and 100 (including 0)

b. Given two variables, x and y, repeatedly increment x by 1 and decrement y by 1 until x is greater than y. Print the initial values of x and y before the loop starts. Also, count the number of iterations this loop takes and print that number when the loop is done

c. Repeatedly prompt for and read a single non-space character into a variable, cmd, until the user enters either 'X' or 'x'.