

## Chapter 5 Review

Due: Monday (10:00 pm)

Points Possible: 20

No late submissions will be accepted

Print this document, work through the questions prior to starting the self assessment. You will be submitting your answers using the Assessment Tool, with the link found in the Assignments module labeled **Chapter 5 Review**. There will be a 1 hour time limit when submitting your answers, so be sure to answer all questions prior to your submission. Each question is worth 1 point unless otherwise noted.

**NOTE:** In the survey, the answers will be presented in a random order, so make sure you select the correct response and not the letter you circle here.

1. The three sections of the `'for'` loop are most commonly used for \_\_\_\_\_ the loop control variable.
  - a. testing, outputting, and incrementing
  - b. initializing, testing, and incrementing
  - c. incrementing, selecting, and testing
  - d. initializing, converting, and outputting
2. Which of the following is not required of a loop control variable in a correctly working loop?
  - a. It is initialized before the loop starts.
  - b. It is tested.
  - c. It is reset to its initial value before the loop ends.
  - d. It is altered in the loop body.
3. If a loop body must be executed at least once, which loop structure would be the best option?
  - a. switch
  - b. for
  - c. while
  - d. do
4. If a loop body uses a numeric value that is incremented by three with each iteration through the loop until it reaches 1000, which loop structure would probably be the best option?
  - a. switch
  - b. for
  - c. while
  - d. do
5. Which of the following is a valid C# pretest conditional expression that enables a loop to be executed as long as the counter variable is less than 12?
  - a. `do while (counter < 12)`
  - b. `while (counter < 12)`
  - c. `switch (counter == 12)`
  - d. none of these
6. Which of the following `'for'` statements would be executed the same number of times as the following `'while'` statement?

```
const int SIZE = 10;
int num = SIZE;
while (num > 0)
{
    Console.Write("{0} ", num);
```

```
    num--;
}
```

- a. `for(num = 1; num < SIZE; num++)`
- b. `for(num = 0; num < SIZE; num++)`
- c. `for(num = 100; num == SIZE; num += SIZE)`
- d. `for(num = SIZE; num < 0; num--);`
- e. none of these

7. What is the output produced from the following statements?

```
const int SIZE = 6;
int num = SIZE;
while (num > 0)
{
    Console.WriteLine("{0} ", num);
    num--;
}
```

8. What is the output produced from the following statement?

```
int aValue = 1;
do
{
    aValue++;
    Console.WriteLine(aValue++);
} while (aValue < 3);
```

- a. 23   b. 234   c. 1234   d. 2   e. none of these

9. Assume the variables 'aValue', 'i' and 'n' have been properly declared as int, what does the following code snippet (fragment) do?

```
aValue = 0; n = 10;
for (i = n; i > 0; i--)
    if (i % 2 == 0)
        aValue = aValue + i;
```

- a. Computes the sum of the integers from 1 through n
- b. Computes the sum of the integers from 1 through n-1
- c. Computes the sum of the even integers from 1 through n
- d. Computes the sum of the odd integers from 1 through n
- e. None of these

10. Write a code snippet (fragment) to display the output 2 4 6 8 10 all on one line with one space between the numbers, using a posttest looping structure. Assume the integer variable 'count' has been properly declared and initialized to 0. (2 point)

11. Write a 'for' loop to display every third number beginning with 10 and continuing through 40 (inclusive) all on one line with one space between the numbers.

12. Convert the following code fragment into a 'for' loop. Use the same variable identifiers.

```
int counter = 100;
do
```

```
{  
    Console.WriteLine(counter);  
    counter--;  
}  
while (counter > 0);
```

13. Use a while loop and write a code snippet (fragment) that will display the following: (2 points)

```
count 1  
count 2  
count 3  
count 4  
count 5  
sum 15
```

Assume that the integer variables 'count' and 'sum' have been declared and initialized as follows:

```
int count = 1;  
int sum = 0;
```

14. Assume that the integer constants MIN and MAX have been properly declared and initialized to 70 and 100 respectively. Write a code snippet (fragment) which will prompt the user for a whole number. A valid number is between MIN and MAX, inclusive. Validate the data by creating a loop of your choice that continues while the entered number is out of range. As an error message, display the valid range of the number and reprompt the user. Use the Parse () method to convert the number. For the answer type the loop and the loop body only. (3 points)

15. Convert the following 'for' loop to a posttest loop. Keep the same variable identifiers and their use. (2 points)

```
int count = 4;  
for (int ctrlVar = 1; ctrlVar <= count; ctrlVar ++)  
{  
    Console.WriteLine("Current value of ctrlVar is " + ctrlVar.ToString());  
}
```

Remember that this is a self assessment; please do your own work, and I strongly encourage you to "test" the questions by writing a small program in Visual Studio.