4.14 LAB: Convert to reverse binary

Write a program that takes in a positive integer as input, and outputs a string of 1's and 0's representing the integer in reverse binary. For an integer x, the algorithm is:

```
As long as x is greater than 0
  Output x % 2 (remainder is either 0 or 1)
  x = x / 2
```

Note: The above algorithm outputs the 0's and 1's in reverse order.

Ex: If the input is:

```
6
```

the output is:

```
011
```

6 in binary is 110; the algorithm outputs the bits in reverse.

Solution:

```
import java.util.Scanner;

public class LabProgram {
  public static void main(String[] args) {
    Scanner scnr = new Scanner(System.in);
    int userNum;

    userNum = scnr.nextInt();

    while (userNum > 0) {
      System.out.print(userNum % 2);
      userNum = userNum / 2;
    }
    System.out.println();
  }
}
```

4.18 LAB: Output range with increment of 5

Write a program whose input is two integers, and whose output is the first integer and subsequent increments of 5 as long as the value is less than or equal to the second integer.

Ex: If the input is: `-15 10`

the output is: `-15 -10 -5 0 5 10`

Ex: If the second integer is less than the first as in: `20 5`

the output is: `Second integer can't be less than the first.`

For coding simplicity, output a space after every integer, including the last.

Solution:

```
import java.util.Scanner;

public class LabProgram {
  public static void main(String[] args) {
    Scanner scnr = new Scanner(System.in);
    int num1;
    int num2;
    int i;

    num1 = scnr.nextInt();
    num2 = scnr.nextInt();

    if (num2 < num1) {
      System.out.println("Second integer can't be less than the first.");
    }
    else {
      for (i = num1; i <= num2; i = i + 5) {
        System.out.print(i + " ");
      }
      System.out.println();
    }
  }
}
```

23.9 LAB: Find largest number

Write a program that repeatedly reads in integers until a negative integer is read. The program also keeps track of the largest integer that has been read so far and outputs the largest integer at the end.

Ex: If the input is:

```
2 77 17 4 -1
```

the output is:

```
77
```

Assume a user will enter at least one non-negative integer.

Solution:

```
import java.util.Scanner;

public class LabProgram {
  public static void main(String[] args) {
    Scanner scnr = new Scanner(System.in);

    int maxSoFar;
    int value;

    value = scnr.nextInt();
    maxSoFar = value; // Initialize current max value
    while (value >= 0) {
      if (value > maxSoFar) { // Test if input is greater than current max
        maxSoFar = value;
      }
      value = scnr.nextInt();
    }
    System.out.println(maxSoFar);
  }
}
```

23.8 LAB: Count multiples

Write a program that takes three integers as input: low, high, and x. The program then outputs the number of multiples of x between low and high inclusive.

Ex: If the input is:

`1 10 2`

the output is:

`5`

Solution:

```
import java.util.Scanner;


public class LabProgram {
  public static void main(String[] args) {
    Scanner scnr = new Scanner(System.in);


    int low, high, x;
    int count = 0; // Declare variable to hold running count.


    low = scnr.nextInt();
    high = scnr.nextInt();
    x = scnr.nextInt();


    for (int i = low; i <= high; i++) { // Loop to test each integer from low to high, inclusive.
      if (i % x == 0)  // Test if current integer is a multiple of digit.
        count++;
    }
    System.out.println(count);
  }
}
```

23.11 LAB: Warm up: Drawing a right triangle

This program will output a right triangle based on user specified height triangleHeight and symbol triangleChar.

(1) The given program outputs a fixed-height triangle using a * character. Modify the given program to output a right triangle that instead uses the user-specified triangleChar character. (1 pt)

(2) Modify the program to use a nested loop to output a right triangle of height triangleHeight. The first line will have one user-specified character, such as % or *. Each subsequent line will have one additional user-specified character until the number in the triangle's base reaches triangleHeight. Output a space after each user-specified character, including after the line's last user-specified character. (2 pts)

Example output for triangleChar = % and triangleHeight = 5:

```
Enter a character: %
Enter triangle height:5

%
% %
% % %
% % % %
% % % % %
```

Solution:

```
import java.util.Scanner;

public class DrawRightTriangle {
   public static void main(String[] args) {
      Scanner scnr = new Scanner(System.in);
      char triangleChar;
      int triangleHeight;

      int i;
      int j;

      System.out.println("Enter a character:");
      triangleChar = scnr.next().charAt(0);

      System.out.println("Enter triangle height:");
      triangleHeight = scnr.nextInt();
      System.out.println("");

      // Draw right triangle
      for (i = 0; i < triangleHeight; ++i) {
         for (j = 0; j <= i; ++j) {
            System.out.print(triangleChar + " ");
         }
         System.out.println("");
      }
   }
}
```