1 Loops

1. (a) What is the value of `result` when the following code is executed?

```
int result = 0;
int a = 5;
int b = 3;
for(int i = 1; i <= a; i++) {
        result+=b;
}
```

   (b) What is the value stored in `result` in terms of `a, b`?

2. (a) What is the value of `result` when the following code is executed?

```
int result = 0;
int a = 5;
int b = 3;
for(int i = 1; i < a; i++) {
        result+=b;
}
```

   (b) What is the value stored in `result` in terms of `a, b`?

3. (a) What is the value of `result` when the following code is executed?

```
int result = 1;
int x = 5;
int n = 3;
for(int i = 1; i <= n; i++) {
        result*=x;
}
```

   (b) What is the value stored in `result` in terms of `x, n`?

4. What is the value of `result` when the following code is executed?

```
int result = 0;
for(int i = 1; i <= 10; i++) {
        if(i != 4) {
                result++;
        }
        else {
                result+=2;
        }
}
```

5. (a) What is the value of `result` when the following code is executed?

```
int result = 0;
int a = 4;
int b = 3;
for(int i = 1; i <= a; i++) {
        for(int k=1; k <= b; k++) {
                result++;
        }
}
```

(b) What is the value stored in `result` in terms of `a, b`?

(c) Based on your answer to part (b), what is the value of `result` when the following code is executed?

```
int result = 0;
int a = 10;
int b = 10;
for(int i = 1; i <= a; i++) {
        for(int k=1; k <= b; k++) {
                result++;
        }
}
```

6. (a) What is the value of `result` when the following code is executed?

```
int result = 0;
int a = 10;
int b = 10;
for(int i = 1; i <= a; i++) {
        for(int k=i; k <= b; k++) {
                result++;
        }
}
```

(b) **(challenging)** What is the value stored in `result` in terms of `a, b`?

7. Write a piece of code that outputs the following in the console, using a **loop**. Use `print()` statement.

```
2 5 8 11 14 17 20 23 26 29 32 35 38 41 44 47 50
```

8. Write a piece of code that outputs the following in the console, using a **loop**. Use `print()` statement.

```
100 95 90 85 80 75 70 65 60 55 50 45 40 35 30 25 20
```

9. Write a piece of code that outputs the following in the console, using a **loop**. Use `print()` statement.

```
1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192
```

10. Write a piece of code that outputs the following in the console, using a **loop**. Use `print()` statement. IMPORTANT: Pay attention to the pattern, this is a tricky one.

```
7 14 21 28 35 42 49 56 63 77 84 91 98 105 112 119 126 133 147
```

11. Assuming an integer variable `n` such that $1 \leq n \leq 10$, write a piece of code that stores into an integer variable `fact` the product of the first `n` positive integers, that is, $1 * 2 * \ldots n$.

12. Assuming an integer variable $n \geq 1$, write a piece of code that stores into a `boolean` variable `isPrime`,

    - `true` if `n` is a prime number.
    - `false` if `n` is not a prime number.

    An integer is called a *prime* if it more than 1 and is divisible only by 1 and itself.