CS265 Intro to Programming Languages
Program 3 – Spring 2023

Objective
Practice if statements
Practice with loops
Practice with formatted output
Practice with functions/methods

Write each of the following programs in C++, Python, and Java. Make sure to use the appropriate extension for each file (.c, .cpp, .py, and .java, respectively). Name each file Prog3ABC.extension, where ABC are your initials. My 3 files would be Prog3TLS.cpp, Prog3TLS.py, and Prog3TLS.java.

In order to avoid unnecessary duplications, Java/C++ function/method headers will not be duplicated if there are no differences. However, all Java methods will require `public static` as part of the header.

```
public static void main (String[] args)
int main()
def main():
```

Write a main function that tests each of the required functions/methods. You do not have to use keyboard input in `main`, just ensure that each function/method is called more than one.

```
int square (int value)
def square (value):
```

This function returns the square of `value.`

```
int summation (int value)
def summation (int value):
```

This function returns the summation of 1 to `intValue`. You can assume that `intValue` will be positive. For example, `summation (5)` would return 15 (1 + 2 + 3 + 4 + 5).

```
int sumOfSquare (int value)
def sumOfSquare (value):
```

This function returns the sum of the squares from 1 to `value`. For example, `sumOfSquares(5)` would return 55 (1 + 4 + 9 + 16 + 25). You **MUST** use a loop and the `square` function to determine the returned value.

```
void rectangle (int width, int height, char character)
void rectangle (int width, int height, char character, char fillChar)
def rectangle (width, height, character, fillChar=' ')
```

`rectangle` will draw the outline on the screen of a rectangle with the specified `width` and `height`. Use `character` to draw the outline. Use `fillChar` for all characters inside the outline. Use a blank space as the fill character when not specified. You can assume that the width and height will be positive. The output for this function will be several strings containing consecutive characters, such as the first line of the rectangle will be a string with `width` number of `characters`.

In Java and C++ all output must originate in one of the `drawRect` methods. The “other” `rectangle` will use the other `rectangle` method to create the output. When creating the output, the outline will use the `character` char. Python will only have a single `rectangle` function.

For example, `rectangle (4, 5)` would create an output of:

```
****
*  *
*  *
*  *
****
```

```
int inputWithinRange (s/String prompt, int lowerLimit, int upperLimit)  //Java, C++
int inputWithinRange (s/String prompt, int upperLimit)                  //Java, C++
def inputWithinRange (prompt, lowerLimit, upperLimit=None)
```

`inputWithinRange` will ask for an integer value, using the provided prompt. The value must be inclusively within `lowerLimit` and `upperLimit`, if not, continue to ask for a value until the entered value is within the provided range. Once a valid value has been entered return it from the function.

In Java (use `String` for the prompt datatype) and C++ (use `string` from the prompt datatype) overload the function. Use a default `lowerLimit` of 0 when only the `upperLimit` is provided. All of the work **must** be done in one function, the second function should call the first function. We want multiple options in calling the function, but do not want to duplicate the work. In Python, if only the prompt and a single number is provided use 0 for the `lowerLimit` and the provided number for the `upperLimit`.

```
boolean isAFactor (int value1, int value2)  //Java
bool isAFactor (int value1, int value2)     //C++
def isAFactor (value1, value2):
```

`isAFactor` returns `True` (Python) or `true` (Java/C++) if the second argument is a factor of (divides evenly into) the first argument. If not, it returns `False` (Python) or `false` (Java/C++).