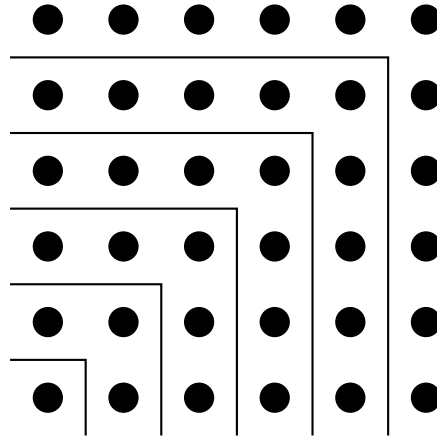


3 September 2020

1. **Warm up:** Consider the picture below.



(a) Use this to compute $1 + 3 + 5 + 7 + 9$. What does $1 + 3 + \dots + 97 + 99$ equal?

(b) Generalize this pattern to dots in a triangular shape.

2. Consider the pseudocode below, which takes as input a set of numbers $X = \{x_1, \dots, x_n\}$.

```
1  for  $i = 2, \dots, n$ :  
2     $x = x_i$   
3     $j = i - 1$   
4    while  $j > 0$  and  $x_j > x$ :  
5       $x_{j+1} = x_j$   
6       $j = j - 1$   
7       $x_{j+1} = x$ 
```

(a) What do you think this code does to X ?

(b) Fix n . Which lines are always called the same number of times, independent of X ?

(c) How many times is line 5 called?

(d) If $x_i = i \pmod{3}$ at the beginning, what is x_1 at the end? What is x_n at the end?

The steps below describe (in a simplified manner) how your code will be checked for correctness.

- (1) Install and start VirtualBox on your computer
- (2) Download Xubuntu from xubuntu.org and install it as a virtual machine
- (3) Install a C++ compiler (such as g++)
- (4) Write C++ code
 - (a) Open a text editor (the default one is “Mousepad”)
 - (b) Enter the following text and save it as `square.cpp`

```
#include <iostream>
using namespace std;
int main()
{
    int x;
    cin >> x;
    int square = x*x;
    cout << "The square of " << x << " is " << square;
    return 0;
}
```

- (5) Compile the C++ code
 - (a) Open the console (the default one is “Terminal Emulator”)
 - (b) Type `g++ -o square square.cpp` and press “Enter”
- (6) Create and execute a test case
 - (a) Open a new file in a text editor
 - (b) Type the number 4 and save it as `test1.in`
 - (c) Open the console
 - (d) Type `./square < test1.in > test1.out` and press “Enter”
- (7) Compare the test case with the expected result
 - (a) Open a new file in a text editor
 - (b) Type `The square of 4 is 16` and save it as `test1.expected`
 - (c) Open the console
 - (d) Type `diff test1.expected test1.out` and press “Enter”
 - (e) If nothing is returned, the files are the same

3. Write a C++ program called `dropunits` that takes as input an integer, and outputs the same integer, but without the units (that is, as a multiple of 10). For example, if the input 145 is given, then the program will print out 140.

4. Write a C++ program called `power` that implements exponentiation without the `<cmath>` package, and only the `<iostream>` package. The program will take two inputs, x and y , and will print out x^y . You may assume that y is an integer greater than 0.