

Homework 2**Out:** 9.17.20**Due:** 9.27.20**1. [Asymptotic comparison, 25 points]**

For each of these problems enter “yes” or “no” indicating whether A is O , o , Θ , ω , Ω of B. Justify your answers.

A	B	O	o	Θ	ω	Ω
$n^3 - 100n - 10,000$	$100n^2 + 5n + 3$					
2^{2n+1}	2^n					
2489^{200}	$\log_{2489}(n)$					
n	$\sum_{i=1}^n \frac{50}{i}$					
$200 n^9$	e^n					

2. [Asymptotics, 25 points]

Place the following functions from asymptotically smallest to largest. When two functions have the same asymptotic order, put an equal sign between them. Provide an explanation for your ordering.

$$\log n, \log \log n, \left(\frac{3}{2}\right)^n, \sqrt{n}, 1000, n^{\frac{1}{\log n}}, 100^{\log n}, \sqrt{\log n}, 5n + 3, (\sqrt{2})^{\lg n}$$

3. [Algorithmic intuition, 50 points]

Write and briefly explain the following C++ function:

long MaxProduct (string file);

that accepts an input file containing sequences of integers. Each sequence starts on a new line, may continue on several subsequent lines, contains at most 100 numbers, and ends with the number -999999 (which is not part of the sequence).

The function outputs to the screen the maximum continuous sub-sequence product of up to 3 numbers for each sequence, one output per line. It returns the maximum of all the outputs.

Sample input:

```
1 2 3 -999999
-5 -2 2 -30 -999999
6 9 -10 1 -999999
-8 -999999
```

Sample output:

6

120

54

-8

The overall max product is: 120

The *MaxProduct* method is a member function of the *MaxProductClass* class, which should be implemented in *MaxProduct.cpp* and declared in *MaxProduct.h*. Try to make your function as efficient as you can.

Submit your solution, in two files: *MaxProduct.cpp*, containing your function, and *MaxProduct.h*, which is required for your code to compile with the provided main file, *HW2_Q3_main.cpp*. Make sure to write your name in a comment at the top of the program, and verify that your program compiles with the provided file on the lab computers.