Homework 5: Due: Thu Apr 5, end of class
Please try to write answers in space provided. If required, attach extra sheets.

1. Order the following functions by rate of growth from increasing to decreasing:

\[ n, \log n, n^2, n^5, (n \log n), 2^n, \log n, \]

You may take help of a plotting software, it is in fact encouraged (though, don’t submit the plots).

Answer.

2. What is the smallest value of \( n \) such that an algorithm whose running time is \( 10n^2 \) runs faster than an algorithm whose running time is \( 2^n \) on the same machine?

Answer.

What about \( 25000n^2 \) and \( 2^n \)?

Answer.

3. This question is for Honors students only. Others don’t need to solve it, though, of course encouraged. No space is provided to answer this question, so please attach an extra sheet.

Show that:

(a) \( 3n^2 = O(n^2) \)
(b) \( 6n^2 + 3n = O(n^2) \)
(c) \( n = O(n^2) \)
(d) [Extra credit] \( n = O(2^n) \)

Do not use any of the rules given in the book, you are required to prove from the defintion:

**Definition.** \( T(n) = O(f(n)) \) if there are positive constants \( c \) and \( n_0 \) such that \( T(n) \leq cf(n) \) when \( n \geq n_0 \).

Here is an example, so you know what sort of an answer is expected.

**Question.** Show that \( 2 + 3n = O(n) \).

**Answer.**

\[
2 + 3n \leq n + 3n \leq 4n \quad \text{(for } n \geq 2) \quad (3+1=4)
\]

Therefore, if we choose \( c = 4 \) and \( n_0 = 2 \) in the definition above, we can conclude that \( 2 + 3n = O(n) \).
4. Consider two functions below to check if a number is a prime or not, `isPrime1` and `isPrime2`. They differ on line 4.

```java
public static bool isPrime1(long n) {
    if(n <= 1) return false;
    for(long i = 2; i <= n-1; ++i)
        if( n % i == 0 )
            return false;
    return true;
}

public static bool isPrime2(long n) {
    if(n <= 1) return false;
    for(long i = 2; i*i <= n; ++i)
        if( n % i == 0 )
            return false;
    return true;
}
```

(a) In each case, how many times (approximately) does the for-loop iterate when \( n \) is

(i) 101

(ii) 10,007

(iii) 1,000,037

[Hint: You may run the program to count]

(b) What is the \( O(.) \) (“Big-O”) running time of the algorithm in terms of \( n \)? [Hint: Write the number of steps taken in the worst case as a mathematical expression in terms of \( n \) and ignore the additive and multiplicative constants.]

Answer.

(c) Assuming it takes 3 nanoseconds \( (10^{-9} \text{ seconds}) \) for each iteration, approximately how long will \( \text{isPrime1} \) take when \( n \) is a prime number as big as \( 10^{16} \).

Answer.

How about \( \text{isPrime2} \)?

Answer.
5. Consider a potential implementation of the `add(..)` function in `ArrayList<T>` class, which (roughly) doubles the array being used internally if too small.

```java
public class MyArrayList<T> {
    private T[] a;
    private int size;

    public void add( T x ) {
        if( a.length == size ) {
            T[] old = a;
            a = (T[]) new Object[(size*2+1)];
            for( int i = 0; i < size; i++ )
                a[i] = old[i];
        } // end of if
        a[size] = x;
        size++;
    } // end of add
}
```

(a) Suppose you add elements to `MyArrayList` using the `add` method repeatedly. Assume the size of the array `a` is initially 0. Then, what is the length of `a` after you have added:

(i) 1 element ___________________________ (ii) 5 elements ___________________________

(iii) 10 elements______________________ (iv) 20 elements ___________________________

For the rest of the parts, answer in terms of an expression in `N`.
(b) Suppose that `size` and `a.length` were equal when `add()` method is called. How many steps will the call to the method take if the array is of size `N`?
   **Answer.**

(c) After the resize has happened in (b), how many times can `add()` be called before another resize is required?
   **Answer.**

(d) What is the average number of steps per `add` from one resize to the next? (In other words, calculate the total number of steps and divide by the total number of calls to `add`.) What do you conclude?
   **Answer.**