Amazing Algorithmic Race 2022 **(Due Wednesday)**

Algorithms Homework 1

Instructor: Jessie

Write up at least one problem!

(1) Minimum and Maximum

(a) In class, we talked about an algorithm for finding the minimum element of a list. What would you change about that algorithm to find the maximum instead?

(b) Suppose we want to find both the minimum element and the maximum element. One way to do this would be to run our algorihtm for finding the minimum element and then run the algorithm for finding the maximum. How many steps would that take?

(c) Here is another possible algorithm:

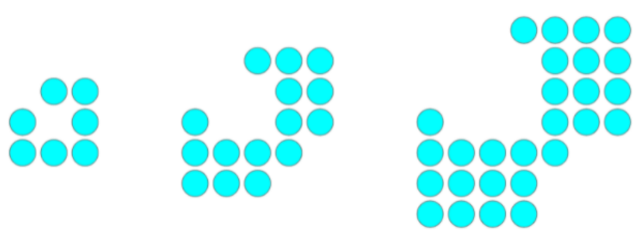
- Compare the first two elements of the list to each other. Keep track of the lesser one as the current min and the greater one as the current max.
- Compare the third and fourth elements of the list to each other. Then compare the lesser of them to the current min and the greater of them to the current max. Replace the min or max if needed.
- Repeat with each pair of elements (fifth and sixth, seventh and eighth, etc.).

Why does this algorithm work?

(d) How many steps does this algorithm take? Explain how you know.

(2) Pattern/Function Growth

Consider the pattern below.



(a) If these are steps 1, 2, and 3, how many dots would be in step n, and how do you know?

(b) What is a pattern that grows more slowly than this pattern?

(c) What is a pattern that grows more quickly than this pattern?

(3) A Calculation Algorithm

This algorithm takes in positive integer inputs n and m. It also keeps track of a number we'll call adder.

1. Let adder be 0.

2. If n is even, go to step 3. If n is odd and greater than 1, go to step 4. If n is equal to 1, go to step 5.
3. Divide n by 2. Multiply m by 2. Go back to step 2.
4. Increase adder by m. Set n to $(n-1)/2$. Multiply m by 2. Go back to step 2.
5. Output $m+$ adder.

Based on that algorithm, answer the following questions!

(a) What operations does this algorithm use?

(b) Try out some values of n and m and show the steps here. What does this algorithm do?

(c) Explain why this algorithm works.

Amazing Algorithmic Race 2022 **(Due Thursday)**

Algorithms Homework 2

Instructor: Jessie

Write up at least one problem!

(1) Time Complexity & Big-O Notation

Identify whether each of the following statements is true or false, and explain why. If it's false, try to figure out what is actually true!

(a) Rule of sums: $\mathcal{O}(f+g) = \mathcal{O}(f) + \mathcal{O}(g)$

(b) Rule of products: $\mathcal{O}(f \cdot g) = \mathcal{O}(f) \cdot \mathcal{O}(g)$

(c) If $g = \mathcal{O}(f)$ and $f = \mathcal{O}(h)$, then $g = \mathcal{O}(h)$.

(2) Time Complexity and Actual Time

(a) Algorithms A and B spend exactly $T_A(n) = c_A n \log_2 n$ and $T_B(n) = c_B n^2$ microseconds, respectively, for a problem of size n. Find the best algorithm for processing $n = 2^{20}$ data items if the algoritm A spends 10 microseconds to process 1024 (or 2^{10}) items and the algorithm B spends only 1 microsecond to process 1024 items.

(b) Algorithms A and B spend exactly $T_A(n) = 5n \log_{10} n$ and $T_B(n) = 25n$ microseconds, respectively, for a problem of size n. Which algorithm is better in the Big-O sense? For which problem sizes does it outperform the other?

(3) Karatsuba Multiplication

This algorithm takes in two two-digit numbers, m and n.

1. Multiply the tens digit of m by the tens digit of n.
2. Multiply the units digit of m by the units digit of n.
3. Find the sum of digits of each of m and n.
4. Multiply the sums of digits of m and n.
5. Subtract results of steps 1 and 2 from step 4.
6. Multiply the result of step 1 by 100 and the result of step 5 by 10. Add these two numbers and the result of step 2.

Based on that algorithm, answer the following questions!

(a) Why does this algorithm work?

(b) How would you generalize it to multiplying numbers with more than two digits? (I recommend starting by thinking about four-digit numbers next.)

(c) How many single-digit multiplications does a standard/traditional multiplication algorithm use to multiply two four-digit numbers? How many would your generalization of this algorithm use?

Amazing Algorithmic Race 2022 **(Due Friday)**

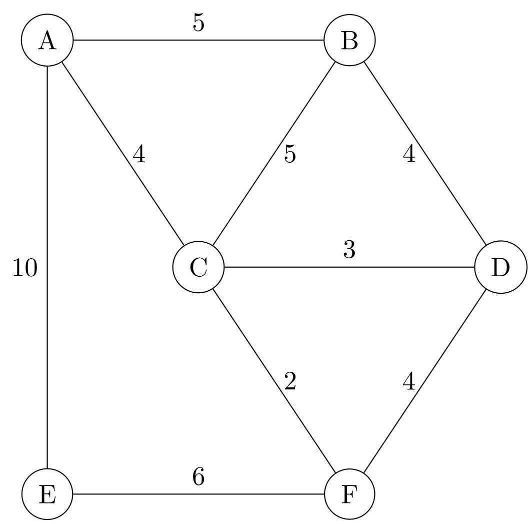
Algorithms Homework 3

Instructor: Jessie

Write up at least one problem! As in class today, for all of these assume that you have a list of all the vertices, that you know what the neighbors of each vertex are, and that you know all the edge weights (if applicable).

(1) Shortest Path Algorithm

Consider the graph with edge weights below. Use a shortest-path algorithm to find the shortest path from vertex B to vertex E, and write out the steps in the process of finding that path.



(2) Widest Path Problem

Find an algorithm that finds the path from vertex s to vertex t in a graph such that the lowest-weight edge has as high a weight as possible, and explain why your algorithm works. (So the path doesn't need to be short! It just needs to not have too low a weight on any edge it uses.)

(3) Connected Components

Two vertices are in the same **connected component** of a graph if there is a path between the two vertices. Find an algorithm to determine the number of connected components of a graph, and explain why it works.