Name: ____

1. (1 pt.)
   - **Read all material carefully.**
   - *If in doubt whether something is allowed, ask, don't assume.*
   - You may refer to your books, papers, and notes during this test.
   - Write, and draw, carefully. Ambiguous or cryptic answers receive zero credit.
   - Use class and textbook conventions for notation, algorithmic options, etc.
   - For the duration of the exam, the only communication (live or network) should be with the instructor for clarifications, etc.
   - At the end of the exam, scan your work to a PDF file named using the following template and upload it in the usual way:
     `cos454-fin-`*`lastname`*`-`*`firstname`*`-`*`pqrs`*`.pdf`
     (replacing *lastname* and *firstname* with yours and *pqrs* with an arbitrary 4-digit number).

   Write your name in the space provided above.

WAIT UNTIL INSTRUCTED TO CONTINUE TO REMAINING QUESTIONS.

Do not write in the following table.

Q	Full	Score
1	1	
2	19	
3	30	
4	20	
5	30	
total	100	

2. (19 pts.) Solve each of the following recurrence using your choice of one of the three main methods described in the textbook and in class:

   (a) $T(n) = 4T(n/7) + 32n + 5n \log n$

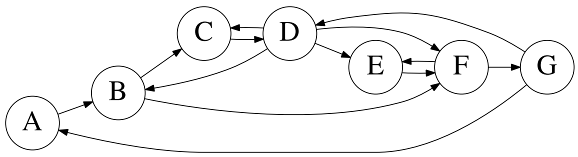
   (b) $S(n) = S(n - 2) + 1/n$

   Show enough work to make it obvious how a method is being used to solve each recurrence.

[additional space for answering the earlier question]

3. (30 pts.) Trace the operation of DFS-VISIT(G, A), for the following directed graph G using the conventions of Figure 22.4 (p. 605) of the textbook. In particular:

- Depict the state of the graph after each iteration of the for loop.
- Annotate each vertex with a letter denoting its color: **W**hite, **G**ray, **B**lack.
- Record the discovery and finishing times in the format `d/f`.
- Highlight *tree* edges using *double lines*, and annotate **F**orward, **B**ackward, and **C**ross edges with the corresponding letters.



[additional space for answering the earlier question]

[additional space for answering the earlier question]

4. (20 pts.) Given a positive integer $n > 2$, is it always possible to generate a set S of points in the x-y plane such that the convex hull of S is the set S itself?

   If so, then provide pseudocode for an algorithm that takes as input a positive integer $n > 2$ and that produces such a set of coordinates as output *Explain why your algorithm and pseudocode are correct.*

   Otherwise, provide a counterexample. That is, provide an integer $k > 2$ and *prove that no set of k points is its own convex hull.*

[additional space for answering the earlier question]

5. (30 pts.)

(a) Provide pseudocode for a $O(n^2)$ *divide-and-conquer algorithm for the convex hull* of points in the x-y plane.

The x- and y co-ordinates of the n points forming the *input* are provided in arrays $X[1, 2, \ldots, n]$ and $Y[1, 2, \ldots, n]$ respectively. The *output* is a binary array $H[1, 2, \ldots, n]$ such that $H[i] = 1$ iff the point $(X[i], Y[i])$ is on the convex hull of the set of points in the input.

[Hint: An $O(n \log n)$ algorithm is also $O(n^2)$ but the $\Theta(n^2)$ algorithm discussed in class may be an easier option.]

(b) Prove the correctness of your pseudocode using appropriate loop invariants and other claims.

(c) Analyze the running time of your pseudocode by following the textbook's method (Section 2.2).

[additional space for answering the earlier question]

[additional space for answering the earlier question]