NAME: ____________________________

SPIRE ID: ____________________________

COMPSCI 250
Introduction to Computation
Second Midterm Fall 2025

D. A. M. Barrington and M. Golin 10 November 2025

DIRECTIONS:

- Answer the problems on the exam pages.
- There are four problems on pages 2-13, some with multiple parts, for 100 total points plus 5 extra credit. Final scale will be determined after the exam.
- We are also providing you with one blank piece or paper to use for scrap. Do NOT write final answers on this since we will not mark anything not in the exam booklet.
- If you need extra space use the back of a page – both sides are scanned.
- But, if you do write on the back, you must explicitly add a note on the front side stating that you are continuing on the back page. Otherwise, we might not see your solution on Gradescope.
- No books, notes, calculators, or collaboration.
- In case of a numerical answer, an arithmetic expression like "$2^{17} - 4$" need not be reduced to a single integer.
- Your answers must be LEGIBLE, and not cramped. Write only short paragraphs with space between paragraphs

1	/20+5
2	/ 20
3	/40
4	/20
Total	/100+5

Family Name: ____________________

Question 1: Induction 1

(A) (10 points)

For naturals n, let $S(n)$ be the sum $1 \cdot 2^0 + 2 \cdot 2^1 + 3 \cdot 2^2 + \ldots + (n+1) \cdot 2^n$.

Formally, $S(n) = \sum_{i=0}^{n}(i+1)2^i$.

Examples: $S(0) = 1 \cdot 2^0 = 1$, $S(1) = 1 \cdot 2^0 + 2 \cdot 2^1 = 5$, $S(2) = 1 \cdot 2^0 + 2 \cdot 2^1 + 3 \cdot 2^2 = 17$.

Prove *by induction*, that for *all* naturals n, $S(n) = n2^{n+1} + 1$.

i) First write your induction hypothesis in the box below. This should be in the form $P(x)$, where you *must* explicitly explain what x is and write an unambiguous statement of $P(x)$.

(ii) Next, write your base case(s) in the box below.

(iii) Finally, provide your induction step. This step will be marked on how clear and mathematically precise your proof is. Ambiguous explanations will have points deducted.
Be sure to clearly describe your induction goal, explicitly identify where the induction hypothesis is being used and justify every one of your manipulation steps.

If you run out of space, continue the proof on the back page (with a note stating that you are writing on the back).

Family Name: ____________________

(B) (10 points) For naturals n, let A_n be defined by

$$A_0 = 0, \quad A_1 = 1 \quad \text{and for all } n \geq 1, \; A_{n+1} = 7A_n - 12A_{n-1}.$$

Examples: $A_2 = 7A_1 - 12A_0 = 7$ and $A_3 = 7A_2 - 12A_1 = 7 \cdot 7 - 12 \cdot 1 = 37$.

Prove *by induction*, that for *all* naturals n, $A_n = 4^n - 3^n$.

As a validity check note that $A_2 = 7 = 4^2 - 3^2$ and $A_3 = 37 = 64 - 27 = 4^3 - 3^3$.

i) First write your induction hypothesis in the box below. This should be in the form $P(x)$, where you *must* explicitly explain what x is and write an unambiguous statement of $P(x)$.

(ii) Next, write your base case(s) in the box below.

(iii) Finally, provide your induction step. This step will be marked on how clear and mathematically precise your proof is. Ambiguous explanations will have points deducted.
Be sure to clearly describe your induction goal, explicitly identify where the induction hypothesis is being used and justify every one of your manipulation steps.

If you run out of space, continue the proof on the back page (with a note stating that you are writing on the back).

Family Name: ____________________

(C) Extra Credit (5 points) Recall the Fibonacci numbers F_n are defined on the naturals n by: $F_0 = 0$, $F_1 = 1$, and,

$$\forall n \geq 1,\ F_{n+1} = F_n + F_{n-1}. \tag{1}$$

The first 11 F_n are shown below.

F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
0	1	1	2	3	5	8	13	21	34	55

In CICS 210 we learn about a data structure called AVL trees and find that the minimum number of nodes N_n in an AVL tree of height n satisfies $N_0 = 1$, $N_1 = 2$, and,

$$\forall n \geq 1,\ N_{n+1} = N_n + N_{n-1} + 1. \tag{2}$$

The first 8 N_n are shown below.

N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7
1	2	4	7	12	20	33	54

Prove, *by induction,* that for all naturals, $N_n = F_{n+3} - 1$.

i) First write your induction hypothesis in the box below. This should be in the form $P(x)$, where you *must* explicitly explain what x is and write an unambiguous statement of $P(x)$.

(ii) Next, write your base case(s) in the box below.

(iii) Finally, provide your induction step. This step will be marked on how clear and mathematically precise your proof is. Ambiguous explanations will have points deducted.
Be sure to clearly describe your induction goal, explicitly identify where the induction hypothesis is being used and justify every one of your manipulation steps.

If you run out of space, continue the proof on the back page (with a note stating that you are writing on the back).

Family Name: ____________________

Question 2 (20): Induction 2

A **rooted unary-ternary tree (RUTT)** is constructed as follows:

R0: It is either a single node, which is its root, or

R1: it is a new node, its root, which is connected to the roots of one other RUTT or

R2: it is a new node, its root, which is connected to the roots of three other RUTT's.

Furthermore,

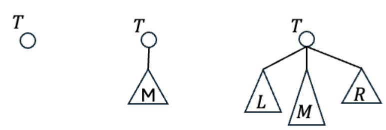
R3: The only RUTT's are those made by the first three rules.

Let $\mathcal{T}$ represent the set of all RUTT's. For tree $T \in \mathcal{T}$,

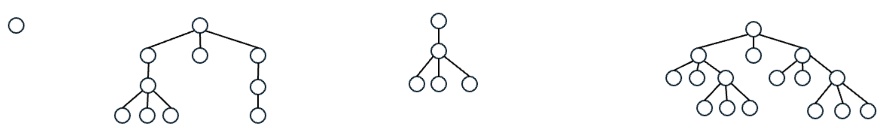
$$\begin{aligned} |T| &= \text{the number of nodes in } T, \\ L(T) &= \text{the number of leaves in } T, \\ I(T) &= \text{the number of internal nodes in } T, \\ d(T) &= \text{the depth of } T. \end{aligned}$$

Recall that the depth of T is the length of the longest path from the root to any leaf.

Diagrams (i-iii) are illustrations of the rules. Diagrams (iv)-(vii) are examples of some RUTT's and their associated values.



(i) (ii) (ii)



(iv)	(v)	(vi)	(vii)
$L(T) = 1$	$L(T) = 5$	$L(T) = 3$	$L(T) = 11$
$I(T) = 0$	$I(T) = 5$	$I(T) = 2$	$I(T) = 5$
$d(T) = 0$	$d(T) = 3$	$d(T) = 2$	$d(T) = 3$
$\|T\| = 1$	$\|T\| = 10$	$\|T\| = 5$	$\|T\| = 16$

Parts (A) and (B) of this problem are on the following pages. When writing the solutions to parts (A) and (B), you must use the the mathematical notation we provided above.

If your proof has multiple pieces, place each piece in a separate paragraph with space between the paragraphs. Be sure to clearly describe your induction goal.

If you run out of space, continue the proof on the back page of the problem (with a note stating that you are writing on the back).

Family Name: ____________________

(A) (10 points)

Prove, by induction, that for every $T \in \mathcal{T}$, it is true that $L(T)$ is odd.

a) First write your induction hypothesis in the box below. This should be in the form $P(x)$, where you *must* explicitly explain what x is and write an unambiguous statement of $P(x)$.

(b) Next, write your base case(s) in the box below.

(c) Finally, provide your induction step. This step will be marked on how clear and mathematically precise your proof is. Ambiguous explanations or explanations missing details will have points deducted.

Family Name: ____________________

(B) (10 points)

Prove, *by induction,* that for every $T \in \mathcal{T}$, it is true that $|T| \leq \frac{3^{d(T)+1}-1}{2}$.

a) First write your induction hypothesis in the box below. This should be in the form $P(x)$, where you *must* explicitly explain what x is and write an unambiguous statement of $P(x)$.

(b) Next, write your base case(s) in the box below.

(c) Finally, provide your induction step. This step will be marked on how clear and mathematically precise your proof is. Ambiguous explanations or explanations missing details will have points deducted.

Question 3 (40):

In this problem, we are going to carry out four searches on three graphs: an undirected graph U, a directed graph D, and an undirected labeled graph G. Each graph has the same six nodes, named for planets in the *Star Trek* universe. Each graph has the same nine edges, but they have the edge types appropriate to their type of graph.

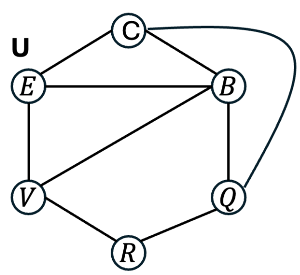
Because of technicalities involving faster-than-light travel, and navigation hazards such as black holes and quasars, it is not always possible to travel *directly* from any planet to any other planet.

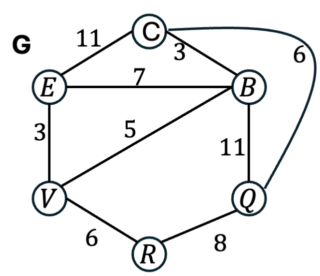
Graph U has an edge between any two nodes (x, y) that allow direct travel from x to y or vice versa.

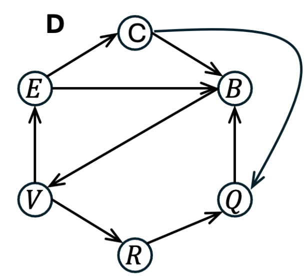
Graph D describes a scenario in which (due to interstellar treaties) each of the edges in U may only be traversed in one direction.

Finally, each edge (x, y) of graph G is labeled with the number of days a starship takes to travel from x to y, or from y to x.

The planets in the graph are Bajor (B), Cardassia Prime (C), Earth (E), Qo'noS (Q), Romulus (R), and Vulcan (V).

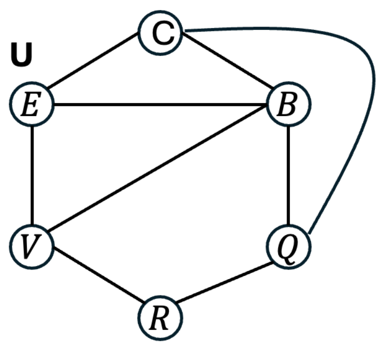






- (A, 10) **Undirected Breadth-First Search:**

  (i) Carry out a **BFS search** for the **undirected** graph U, starting with Earth and with **no goal node**.



List all the events that occur in the running of the BFS, in the order in which they occur.

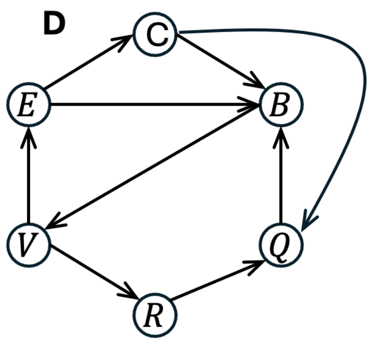
An event is either the placement of a list-item on the open list or the removal of a list-item off the open list. Describe list-items in the form of pairs such as "(a, b)", meaning that the node a went on the list while node b was processed.

When two or more list-items need to come off the open list, and they entered at the same time, take the one first that comes earlier alphabetically.

(ii) In addition to writing the list, **also draw** the BFS tree from this search, starting with Earth and with no goal node. Indicate the tree edges and the non-tree edges.

- (B, 10)

  (i) Carry out a **DFS search** for the **directed** graph D, starting with Earth and **with no goal node**.



List all the events that occur in the running of the DFS, in the order in which they occur. An event is either the placement of a list-item on the open list or the removal of a list-item off the open list.

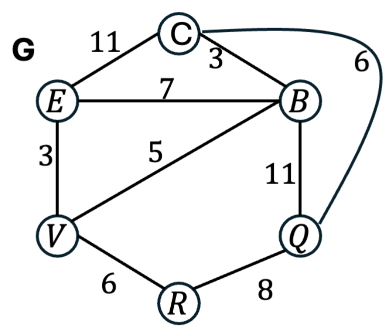
Describe list-items in the form of pairs such as "(a, b)", meaning that the node a went on the list while node b was processed.

When two or more nodes need to come off the open list, and they entered at the same time, take the one first that comes earlier alphabetically.

(ii) In addition to writing the list, **also draw** the DFS tree for this search, indicating the tree and non-tree edges, starting with Earth and with no goal node. For each non-tree edge, **identify** it as a back, cross, or forward edge.

- (C, 10) **Uniform-Cost Search:**

  (C) Conduct a uniform-cost search of the labeled undirected graph G, with Earth as the start node and Qo'noS as the goal node.



List all the events that occur in the running of the UCS, in the order in which they occur. An event is either the placement of a list-item on the open list or the removal of a list-item off the open list.

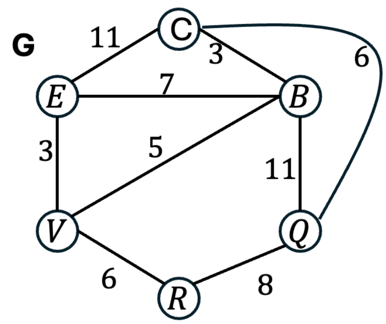
Describe list-items using triples such as "(a, t, b)", meaning that this node a went on the list while b was processed, with t being its priority value.

When two or more list-items need to come off the list, they should follow the rules of uniform-cost, and you may break ties as you like.

- (D, 10) A^* **Search:**

  We will now ask you to conduct an A^* search for the labeled undirected graph G using the following heuristic function h. For any node x, $h(x)$ will be the a lower bound on the distance in G from x to Q. Here are the values of h:

$$\begin{array}{rclrclrcl} h(B) & = & 7, & h(C) & = & 4, & h(E) & = & 12 \\ h(Q) & = & 0, & h(R) & = & 8, & h(V) & = & 12 \end{array}$$



Conduct an A^* search of the labeled graph G to determine the shortest path and the distance along that from Earth to Qo'noS, using the heuristic function h above. (Thus Earth is the start node and Qo'noS is the goal node.)

List all the events that occur in the running of the A^* search, in the order in which they occur. An event is either the placement of a list-item on the open list or the removal of a list-item off the open list.

Describe list-items using tuples such as "(a, t, d, b)", meaning that this node a went on the list while b was processed, with t being its priority value and d being the distance (along the known path) from Earth to a.

When two or more list-items need to come off the list, they should follow the rules of uniform-cost, and you may break ties as you like.

Family Name: ____________________

Question 4 (20) The following are ten true/false questions, with no explanation needed or wanted, no partial credit for wrong answers, and no penalty for guessing.

After reading the questions, write the correct answer, either T (for true) or F (for false), in the corresponding column in the table below. Anything not written in the table will be ignored.

(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)

(a) Let $P(n)$ be a predicate on the naturals. If $\forall n : P(n)$ can be proved by ordinary induction, then it can also be proved by strong induction.

(b) Let $P(n)$ be a predicate on the naturals, and assume that $P(1)$ and $\forall n : P(n) \rightarrow P(n+2)$ are true. Then $P(n)$ must be true for all positive naturals.

(c) Let $\Sigma = \{a\}$ be an alphabet with one letter, and consider the set of strings Σ^*. Define a function f from $\mathbb{N}$ to Σ^* so that $f(n) = a^n$ and a function g from Σ^* to $\mathbb{N}$ such that $g(w) = |w|$. Then f and g are not inverses of one another.

(d) Consider an infinite plane crossed by n (infinite) lines, no two of which are parallel. Then the lines divide the plane into exactly $\frac{n^2+n+2}{2}$ regions.

(e) Let T be a tree with four or more nodes. Then it may be possible to add two edges to T to form a new graph G that has three different cycles.

(f) Let F be a forest consisting of exactly two trees T and U. If we create a new graph G by adding one edge to F, with one endpoint in T and the other in U, then G must be a tree.

(g) Let G be a graph containing nodes u and v, and consider both a BFS and a DFS of G, with start node u and goal node v, where each search uses a closed list to avoid revisiting nodes. Then it is possible that the BFS reaches v and the DFS does not.

(h) Consider a BFS (using a closed list) of an undirected graph that is a forest. Then it is possible for two copies of the same node to be stored on the queue at the same time during the search.

(i) Let G be a weighted graph with positive edge costs, let g be a node of G, and let $h(x)$ be an admissible, consistent heuristic for an A^* search where g is the goal node. Let s and y be two nodes of G such that the best-path distance from s to y is less than the best-path distance from s to g. Then, as in the corresponding uniform cost search, an A^* search from s to g will eventually take y off of its priority queue before taking g off.

(j) Consider any two-player game represented by a finite game tree such that there is more than one leaf, each leaf value is either $+1$ or -1, and each internal node has exactly two children. If the first player to move has a winning strategy, then it is possible to change at least one of the leaf values to its opposite without changing the value of the game.