# CSE 21 – Winter 2018 – Midterm 2 Draft

## 1. Representing Problems as Graph (10 points)

Coachella is an annual music and arts festival held in California that features many famous musical bands and artists and attracts hundreds of thousands of audiences. At the event, there are several stages that continuously play live music. Participants are given a schedule of the different artists and bands, the times they will be performing, and the stage where they will be performing. This schedule is given as a list, with performance $i$ beginning at time $b_i$, ending at time $e_i$, and playing on stage $s_i$. Assume that it does not take you any time to travel between stages, and that you will stay for the entire duration of a performance that you start watching.

(a.) (4 points) Describe how to model this situation using a directed graph, where paths in your graph should represent possible sequences of performances you can see. What are the vertices, and when are two vertices connected with an edge?

**Answer:** The vertices are the performances. Connect show $i$ to show $j$ with a directed edge when if $e_i \leq b_j$.

(b.) (2 points) How can this graph be used to determine the maximum number of performances you can see at Coachella?

**Answer:** To find the maximum number of performances you can watch at the event, you would need to find the longest path in your graph.

(c.) (2 points) Is your graph a DAG? Explain why or why not.

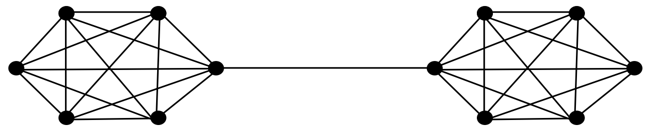
**Answer:** It is a DAG since the graph will never include cycles. The edge rule implies that if there is a path from show $i$ to show $j$, this means $b_i < e_i \leq b_j < e_j$, so there can't be a path from show $j$ back to show $i$ because we don't have $e_j \leq b_i$. A more informal way to say this is that time moves forward, so we can't go back to watch a show again.

(d.) (2 points) In terms of the performances, what would a source in your graph represent?

**Answer:** Sources in this DAG are the performances that we can watch at the event. In other words, a show $i$ is a source if there is no other show that ends at the same time or before the start time of $i$.

## 2. Graphs, Trees, Tours, and Counting (9 points)

For $n$ an integer with $n \geq 3$, let $Barbell(n)$ be the graph that is made by attaching with a single edge two complete graphs with $n$ vertices each. A complete graph with $n$ vertices is a graph in which every vertex is adjacent to every other vertex besides itself. As an example, the graph $Barbell(6)$ is pictured below.



The following questions are about $Barbell(n)$, for an integer $n \geq 3$ :

(a.) (3 points) How many **edges** are in the graph $Barbell(n)$? Briefly justify your answer.

**Answer:** $\textit{number of edges} = 2|K_n| + 1 = 2\binom{n}{2} + 1 = n(n-1) + 1 = n^2 - n + 1.$

(b.) (3 points) When $n$ is even, how many **Eulerian circuits** does the graph $Barbell(n)$ have? Briefly justify your answer.

**Answer:** There is **no** Eulerian circuits since there is an odd number of odd-degree vertices.

(c.) (3 points) When $n$ is odd, how many **Hamiltonian circuits** does the graph $Barbell(n)$ have? Briefly justify your answer.

**Answer:** There is **no** Hamiltonian circuit since there is a bridge in the graph. This bridge prevent us from going back to a vertex on either side after crossing it.

## 3. Counting (16 points)

**Note:** There is no partial credit for this problem, and so you do not need to show any work. You may leave your answers as unsimplified expressions with factorials, exponents, binomial coefficients, etc.

In a futuristic dystopian Chicago, society is divided into five factions: Abnegation, Amity, Candor, Dauntless and Erudite. At the age of 16, each person is allowed to choose any faction as their permanent social group at the Choosing Ceremony.

This year, there are 25 candidates at the ceremony, including Tris, Caleb, Christina, and Peter. All 25 candidates get called up one at a time to pledge themselves to the faction that they want to join.

(a.) (2 points) How many possible call orders of all 25 candidates have Tris as the last person called?

**Answer:** $24!$. Ignoring the last person, there are 24! ways to rearrange the order of the first 24 candidates.

(b.) (2 points) How many possible call orders of all 25 candidates have Tris, Caleb, Christina, and Peter (in **any** order) as the first four candidates called up to pledge their allegiance?

**Answer:** $4! \times 21!$. Since the first four can be called up in any order, there are 4! to choose an order for them. For each arrangement of the first four people, we now have 21! to rearrange the remaining 21.

(c.) (2 points) How many ways are there to select 6 candidates to join Dauntless?

**Answer:** $\binom{25}{6}$

(d.) (2 points) How many ways are there to distribute all 25 candidates into factions such that Tris, Christina, and Peter all join Dauntless?

**Answer:** $5^{22}$. This is equivalent to distributing 22 distinguishable balls into 5 distinguishable boxes.

(e.) (2 points) How many ways are there to distribute all 25 candidates into factions such that **exactly** 5 candidates belong to Erudite?

**Answer:** $\binom{25}{5} \times 4^{20}$. First choose five candidates for Erudite in $\binom{25}{5}$ ways. Then there are $4^{20}$ ways to distribute the remaining candidates into the other four factions.

(f.) (2 points) How many ways are there to distribute all 25 candidates into factions if each of the five factions has room for only five incoming candidates?

**Answer:** $\binom{25}{5}\binom{20}{5}\binom{15}{5}\binom{10}{5}\binom{5}{5} = \frac{25!}{5!5!5!5!5!}$. This is equivalent to the number of rearrangements of an alphabet of $5A's, 5B's, 5C's, 5D's$ and $5E's$.

(g.) (2 points) For every fixed call order, how many ways are there to distribute all 25 candidates into factions such that nobody joins the same faction as the person called up just before them?

**Answer:** $5 \times 4^{24}$. There are 5 ways to pick a faction for the first person. For each of the remaining candidates, there are 4 ways to pick a faction that is not the same as the one chosen by the person before them.

(h.) (2 points) How many ways are there to distribute all 25 candidates into factions such that Tris and Caleb belong to different factions?

**Answer:** $5 \times 4 \times 5^{23} = 4 \times 5^{24}$. There are 5 ways to pick a faction for Tris and 4 ways to pick a faction for Caleb from the remaining four. Then we can distribute the rest into factions in $5^{23}$ ways.

(i.) (2 points) How many ways are there to distribute all 25 candidates into factions if there is at least one candidate in Dauntless and at least one candidate in Erudite?

**Answer:** $5^{25} - 2 \times 4^{25} + 3^{25}$. We shall use inclusion-exclusion.

Let $A$ be the collection of distributions of 25 candidates into factions such that no one is in Dauntless. Then $|A| = 4^{25}$.

Let $B$ be the collection of distributions of 25 candidates into factions such that no one is in Erudite. Similarly, $|B| = 4^{25}$.

$A \cap B$ is the set of distributions of 25 candidates into factions such that no one is in Dauntless **and** no one is in Erudite. Here, $|A \cap B| = 3^{25}$.

$A \cup B$ is the set of distributions of 25 candidates into factions such that no one is in Dauntless **or** no one is in Erudite. By inclusion-exclusion,

$$|A \cup B| = |A| + |B| - |A \cap B| = 2 \times 4^{25} - 3^{25}.$$

Finally, let $U$ be the total number of ways to in order to distribute all 25 candidates into factions with $|U| = 5^{25}$.

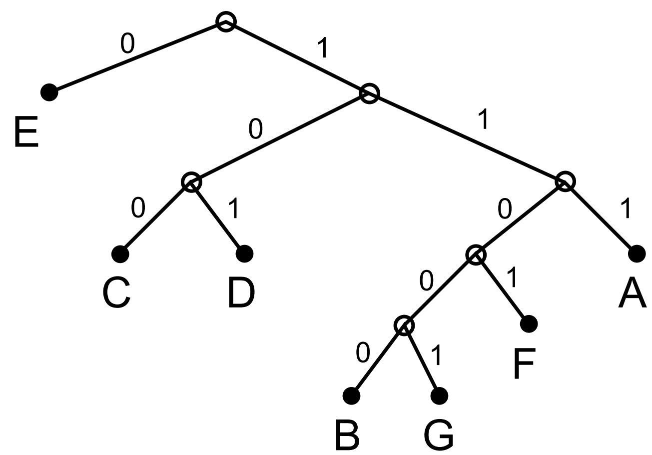
To obtain the number of ways to distribute all 25 candidates into factions if there is at least one candidate in Dauntless and at least one candidate in Erudite, we need to remove from $U$ all distributions that has no one in Dauntless **or** no one is in Erudite. Thus, the required number of distributions is given by

$$5^{25} - 2 \times 4^{25} + 3^{25}.$$

## 4. Encoding and Decoding (10 points)

Suppose that we want to encode a message that only contains the certain letters of the alphabet, each with a different frequency. The **Huffman code** is a nice algorithm that can solve this problem and produce an optimal code. For this algorithm, we first construct a binary tree where every left edge is labeled 0 and every right edge is labeled 1. Each leaf is labeled with one of the letters that appears in the message. This creates a 1-1 correspondence between the leaves and the letters, and the path from the root to the leaf is the codeword for the corresponding letter.

(a.) (5 points) The following tree gives is a Huffman encoding for our message that only contains the letters from $A$ to $G$. In this case, the letter $A$ is encoded as 111 while the letter $E$ is encoded as 0. in general, a letter that occurs with higher frequency in the message will require less bits to encode.



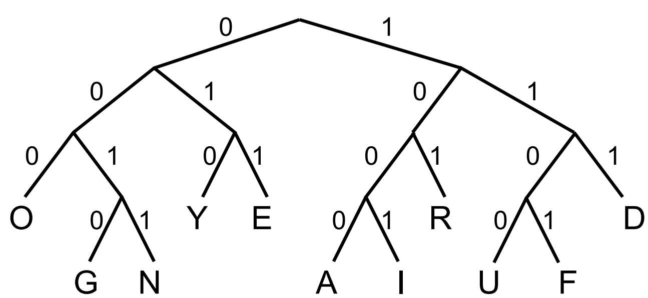

Finish the following table to obtain the codeword for each of the remaining letters.

| Letter | $A$ | $B$ | $C$ | $D$ | $E$ | $F$ | $G$ |
|---|---|---|---|---|---|---|---|
| Codeword | 111 | 11000 | 100 | 101 | 0 | 1101 | 11001 |

(b.) (5 points) A file contains the following sequence of binary digits

**01000011001000101011111000100100110010110110010011011**

Decode the message knowing it was encoded using the Huffman code given by the tree below




**Answer:** "YOUAREDOINGFINE" or "YOU ARE DOING FINE"