SigmaCamp's *Problem of the Month* Contest

November 2024

Starting from September 2024, we are requiring all submissions to be `.pdf` files (except for CS, which requires `.py` or `.java` files). If you are using Word, you may export to PDF by clicking File > Export > Create PDF/XPS Document.

Mathematics

For all mathematics problems, please provide full justification. **Do not include any code** in your submission — all code submissions will be awarded no points.

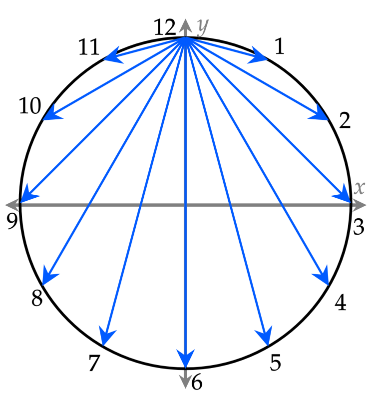
5 points:

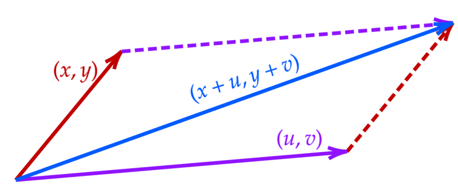
Consider a circular clock of radius 1 foot with 11 vectors* that originate at the 12 o'clock mark and point to the other 11 numbers, as shown in the diagram on the left. Calculate the exact sum of these 11 vectors.

(*) If you have never seen vectors before, this is an opportunity to learn about them. To solve this problem you need to know that vectors in a plane can be represented as arrows. They can be added using the parallelogram rule, as shown in the diagram on the right: place the two vectors at a common starting point and draw a parallelogram where they form adjacent sides — the diagonal from the starting point to the opposite corner is their sum.

Vectors can also be represented in coordinates, like (x, y), where x is the difference in the x-coordinate between the arrow's point and the tail, and y is the difference in y-coordinates. In this representation, vector addition works like this: $(x, y) + (u, v) = (x + u, y + v)$.

To learn more, we recommend Khan Academy (you don't need the whole unit) or an LLM, like Learn About.





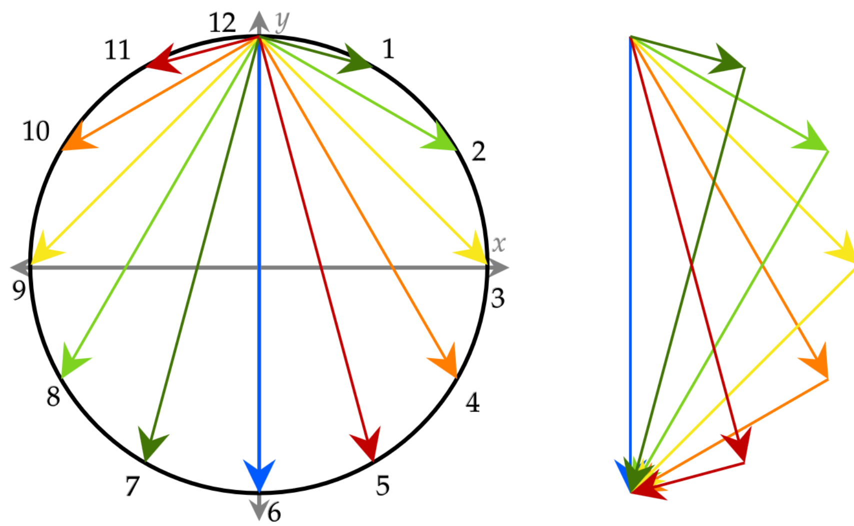
Parallelogram rule

Hint:

This problem is about geometry, not algebra.

Solution:

> **Answer**: $(0, -12)$ or 12 feet south/down.



All vectors except the one that points at 6 can be paired with another vector as shown in the diagram. Note that each pair of vectors is perpendicular and the sum of each pair will be equal to the vector that is pointing at 6, or $(0, -2)$. There are 5 such pairs, and together with the vector pointing at 6 the total is $(0, -12)$.

10 points:

Last Halloween, 5 campers decided to dress up as the original emotions from Inside Out (Joy, Sadness, Fear, Anger, and Disgust). Without coordinating with each other, they each randomly selected an emotion from the film to dress up as.

The campers do not want to go trick-or-treating as a group that contains more than one copy of any costume. They agreed that they would split off into several groups to make sure that all costumes in a group are different, but the campers want to split into as few groups as possible.

(a) What is the probability that the campers were all dressed as different emotions and were able to go trick-or-treating as one group?

(b) What is the probability that the campers had to split up into exactly 2 groups? To help you check your answer, please list all other situations that may occur (up to and including all 5 campers wearing the same costume and splitting into 5 groups), calculate their probabilities, and confirm that all the probabilities add up to 1.

Hint:

How many people can be dressed as Joy to guarantee that the campers will split into exactly 2 groups?

Solution:

Answer: (a) $5!/5^5 = 24/625$; (b) $2100/5^5 = 84/125$.

(a) There are 5^5 total options, and 5! ways to assign 5 costumes to 5 people, so the probability is $5!/5^5 = 120/3125 = 24/625 \approx 0.0384$.

(b) First of all, to have to split into 2 groups at least two people need too wear the same costume. To guarantee that splitting into two groups is enough, no more than two people can wear any one costume.

The total number of costume assignments is $5^5 = 3125$.

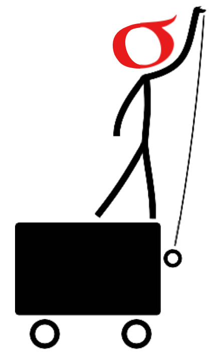
There are two different ways in which the "at least two people wear the same costume, and no more than two people can wear any one costume" condition can be fulfilled:

- 2 people wear one costume and 3 others each wear a different costume, corresponding to $5 = 3 + 1 + 1 + 1$, or
- 2 people wear one costume, 2 other people wear another costume, and the last person wears a third costume. This corresponds to $5 = 2 + 2 + 1$.

Overall, there are 7 such partitions of the 5 costumes, shown below with the count of how many ways there are for each partition to occur, and the (approximate) corresponding probability:

(1)	5	$\binom{5}{1} = 5$	0.0016
(2)	$4 + 1$	$5 \cdot \binom{5}{4} \cdot 4 = 100$	0.0320
(3)	$3 + 2$	$5 \cdot \binom{5}{3} \cdot 4 = 200$	0.0640
(4)	$3 + 1 + 1$	$5 \cdot \binom{5}{3} \cdot \binom{4}{2} \cdot 2 = 600$	0.1920
(5)	$2 + 2 + 1$	$5 \cdot \binom{5}{2} \cdot 4 \cdot \binom{3}{2} \cdot 3 \div 2 = 900$	0.2880
(6)	$2 + 1 + 1 + 1$	$5 \cdot \binom{5}{2} \cdot \binom{4}{3} \cdot 3! = 1200$	0.3840
(7)	$1 + 1 + 1 + 1 + 1$	$5! = 120$	0.0384

The total is indeed $5+100+200+600+900+1200+120 = 3125$, and the required probability is $\frac{900+1200}{3125} = \frac{84}{125}$.



Physics

Five astronauts are on a long-term mission in deep space.

5 points:

In their spaceship, there are two rooms. One of them is near the engine, which generates heat, but otherwise they have the same volume and are connected by an open door. Which room has the greater mass of air? There is no air flow out of or into the rooms.

Hint:

Air is well-modelled as an ideal gas.

Solution:

Air, at the temperatures and pressures comfortable for humans, behaves very closely to an ideal gas. Ideal gasses follow the equation $PV = Nk_BT$[1], where P is the pressure, V the volume, N the number of molecules, and T the temperature. k_B denotes Boltzmann's constant. Since the rooms are connected by an open door, the pressure in both rooms must be the same. We also know that the volume of both rooms is the same. Thus, we have $PV = N_1k_BT_1$ and $PV = N_2k_BT_2$, where N_1 and T_1 refer to the number of molecules and temperature in the warmer room and N_2 and T_2 refer to the number of molecules and temperature in the colder room. Then $N_1T_1 = N_2T_2$. Rearranging, $N_2/N_1 = T_1/T_2$. Since $T_1 > T_2$, $N_2/N_1 > 1$, so $N_2 > N_1$. In other words, there are more molecules of air in the colder room than in the warmer room. This means that the colder room has a greater mass of air.

10 points:

Their rocket breaks down and they must leave the ship to fix it. The spaceship has an airlock. Initially, the airlock, which has volume v, has no air inside it. When the astronauts leave the spaceship, air from the spaceship is used to slowly fill the airlock. The spaceship is initially at pressure p and has total volume V (including the airlock). Unfortunately due to malfunction, every time the astronauts leave the airlock all of the air inside it is expelled into outer space. What will be the pressure inside the spaceship after each of the 5 astronauts leaves the spaceship? Assume only one astronaut can use the airlock at a time and that the temperature in the spacecraft is kept at constant temperature T (due to a heating system powered by solar panels).

Hint:

No hint this month.

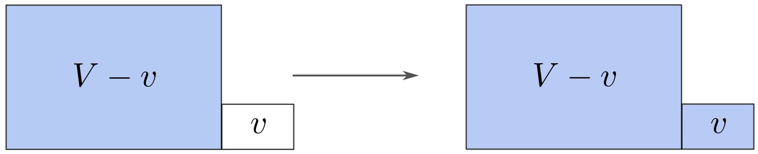
Solution:

First, we schematically separate the spaceship into two parts: an airlock with volume v and the rest of the spaceship with volume $V - v$. Initially, the air was only present in the bigger part with volume $V - v$ and pressure p. Applying an ideal gas law, we get

$$p\,(V - v) = n\,R\,T,$$

where n is the initial number of moles of the air. When the airlock is slowly filled with air, the same moles are spread over the volume V (see the picture below).

[1] You may also see it written as $PV = nRT$. Then n is the number of moles and R is the ideal gas constant.



Since the temperature in the spaceship is kept constant, we can apply the ideal gas law to the picture on the right:

$$p_1\,V = n\,R\,T.$$

This allows us to determine the pressure in the spaceship after the first astronaut leaves:

$$p_1\,V = p\,(V - v) \quad \Rightarrow \quad p_1 = p\left(\frac{V - v}{V}\right).$$

Since all the air is expelled from the airlock with the first astronaut, we return to the same picture on the left, but with fewer moles of air n_1:

$$p_1\,(V - v) = n_1\,R\,T.$$

After the airlock is filled with air for the second time, the pressure changes from p_1 to p_2 defined by

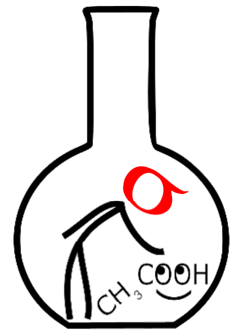
$$p_2\,V = n_1\,R\,T = p_1\,(V - v)\,.$$

Hence, p_2 is related to p via

$$p_2 = p_1\left(\frac{V - v}{V}\right) = p\left(\frac{V - v}{V}\right)^2.$$

Repeating the same steps for astronauts 3 to 5, we get

$$p_5 = p\left(\frac{V - v}{V}\right)^5.$$



Chemistry

5 points:

There is a hydrocarbon that is a gas under standard conditions. It quickly reacts with bromine (even in the absence of light) and with a $KMnO_4$ solution. This gas was mixed with oxygen in a 1:10 ratio, and the mixture was placed in a steel container and ignited. After the mixture cooled to 0°C, the water formed by the reaction was collected and measured to be 720 mg. The gas remaining in the vessel was pumped through an excess of a $Ca(OH)_2$ solution, and the precipitate formed was collected, dried, and its mass was measured. The mass of $CaCO_3$ formed was 4 g. What is the name and formula of the gas? Explain your answer.

Hint:

A double bond in alkenes reacts with strong oxidizers, including halogens and a permanganate anion.

Solution:

We know that the hydrocarbon contains multiple bonds, because it reacts with bromine. The combustion reaction will have two products: water and carbon dioxide. All the hydrogen from the hydrocarbon ends up as part of the water, and all the carbon becomes part of the carbon dioxide.

The reaction produces 720 mg of H_2O, which is 0.04 moles H_2O, which contains 0.08 moles of H.

The carbon dioxide from the reaction formed $CaCO_3$ according to this overall reaction: $Ca(OH)_2 + CO_2 \longrightarrow H_2O + CaCO_3$ 4 g of $CaCO_3$ were formed, which is 0.04 mol $CaCO_3$, which contains 0.04 mol C.

So the original hydrocarbon contained 0.04 mol C and 0.08 mol H, or a 1:2 ratio. Many molecules fit this formula such as C_2H_4 (ethylene), C_3H_6 (propylene), or C_6H_{12} (cyclohexane).

However, we know the molecule is gaseous and contains double bonds, so cyclohexane is ruled out.

10 points:

Alice found a bottle with some white powder in her lab. The label was partially peeled off, and she was able to read only

> ...lecular mass 380...
>
> ...rmula $Na_4C_{10}H_{12}$...

She took 4 g of this powder and dissolved it in 100 ml of water to give a colorless basic solution. Then Alice did the following steps, in order:

1. 10 ml of 1 M Na_2SO_4 is mixed with the solution. There is no visible change.
2. 10 ml of 1 M $CaCl_2$ is added to the previous mixture. There is still no visible change.
3. 20 ml of 1 M $CuCl_2$ is added to the previous mixture, forming a precipitate. When it is filtered out, the precipitate is white and the remaining solution is blue.

Give a possible identity for the mystery substance, or explain why the observations above are not possible.

Hint:

Look at the partial formula for the unknown. It contains sodium, carbon, and hydrogen, so it could be the sodium salt of some organic acid.

Look carefully at the salts that were added in steps 1, 2, and 3, and think about what would happen if any two of them were mixed together.

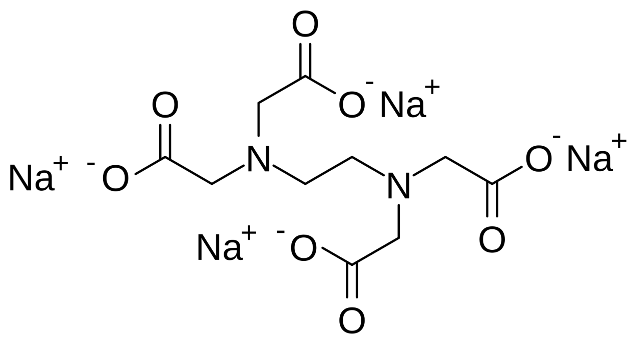


Figure 2: Structure of tetrasodium EDTA

Copper compounds are usually blue, but the precipitate observed in step 3 was white. What could it be?

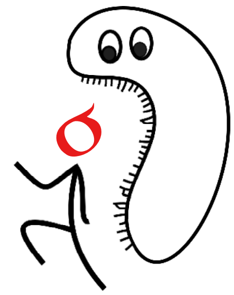
Solution:

The compound has molecular mass 380, and the partial formula $Na_4C_{10}H_{12}$... There are 4 sodium atoms in this formula. This suggests a sodium salt of some organic molecule with a -4 charge. The known fragment has mass 224, and so the remainder of the molecule has mass 156. This mass must be made up of atoms like oxygen, sulfur, nitrogen, or halogens.

The reactions give us more information. When sodium sulfate and calcium chloride are mixed, you would expect a precipitate of calcium sulfate, but no precipitate was observed. When copper chloride is mixed with either sodium sulfate or calcium chloride, no precipitate is expected, but one was observed anyway. Also, the precipitate appeared when we added copper, but it is a white powder and most copper compounds are blue.

From this we conclude that the unknown compound must somehow prevent calcium from precipitating out of solution in step 2, but it releases the calcium in step 3, when copper is added. One possibility is a chelating agent: molecules that can bind to metal ions, forming soluble complexes.

Looking at a list of chelating agents, we see that EDTA forms an anion with a -4 charge, and has a high affinity to bind with calcium, but an even higher affinity for binding copper. So EDTA would bind calcium at first but then release it once copper is available, just as observed. The tetrasodium salt of EDTA has a molar mass 380 and formula $Na_4C_{10}H_{12}N_2O_8$, which matches our partial formula. This is enough evidence to conclude that the unknown compound is tetrasodium EDTA.



Biology

5 points:

On August 14, 2065, Bob, a brilliant genetic engineer and Alice's colleague, was suffering from anxiety and depression, he was gradually losing interest in his work, and Alice often noticed him staring at a map of Connecticut hanging on the wall opposite his desk. A few days ago, he disappeared. He was not answering his phone. Alice found a mysterious vial on Bob's desk with the inscription "Please read me!!!" and two test tubes labeled "primers". In Bob's lab notebook, Alice found the sequence:

> 5'-GCCGTGCTCATCCTCTTCATGGAACACATCATGATCTCTTCCTCTATCGGTATGGCTTGTGC
> TATGCCCTAGCGCGCCTCGTGCCCATGTTATCTGCATCTGATACTGGCAAGCTTGAGTCTCGTA
> GAGGGGGGTAGAATTCCAGGTGTAGCGGTGAAATGCGTAGAGATCTGGAGGAATACCGGTGGC
> GAAGGCGGCCCCCTGGACGAAGACTGACGCTCAGGTGCGA-3'

and a note:

> Primers:
>
> 5'-GTGCTCATCCTCTTCATGG
>
> 5'-CACGAGGCGCGCTAGG

Alice concluded that this might be a message from Bob and decided to use polymerase chain reaction (PCR) to identify it. She ordered primers with the sequence specified in the note, mixed them with the contents of the mysterious vial, added other auxiliary reagents, and after 50 cycles of the PCR reaction, she obtained a significant amount of product (a fragment of double-stranded DNA).

Alice sequenced this DNA using a nanopore DNA sequencer and obtained a sequence that apparently encoded some kind of short protein (it began with a methionine codon, which is a standard start codon, and ended with a standard stop codon). And after Alice translated the nucleotide sequence into an amino acid sequence (using a standard table of single-letter codons), she smiled and breathed a sigh of relief.

"Oh, I know where Bob is!"

Explain how Alice knew where Bob was.

Hint:

The problem says that the sequence encodes some protein (not a protein fragment). That means it is supposed to contain a start codon and a stop codon. If you find them, you know the beginning and the end of the message.

Solution:

It appears that the answer lies within the sequence itself. If the sequence encodes a full-sized protein (rather than just a fragment of a peptide sequence), it must start with the start codon (which is the methionine codon) and end with one of the three stop codons. Identifying the "ATG" codon in the sequence is straightforward; this indicates the start of the coding region. Similarly, the stop codon, "TAG," can also be easily located downstream.

If Alice reads the sequence from the very first codon to the stop codon, it will be:

> ATG GAA CAC ATC ATG ATC TCT TCC TCT ATC GGT ATG GCT TGT GCT ATG CCC
> TAG

Up to this point, it appears to contain no significant information. However, considering that each codon corresponds to a specific amino acid and that each amino acid is represented by a single letter, this sequence can be converted to:

M E H I M I S S S I G M A C A M P

That provides Alice with a clue about where Bob might be.

10 points:

Alice, who was a researcher in Bob's lab, tried to enzymatically synthesize a 51-nucleotide-long fragment by mixing the template strand

5'-AGGTACACACCGTATATTCAACAAGTACTTCCTTCGACGCGTAGCTACGCG

with a 14-mer primer 5'- CGCGTAGCTACGCG, four deoxynucleotide triphosphates, and DNA polymerase.

To her big surprise, instead of a 51 nucleotide long product, she got mostly a longer DNA (about 88 nucleotide long), along with a small amount of the 51-mer. Alice was very disappointed, but Bob told her that the result was quite predictable and that the problem could be resolved. "By the way, if you repeat this reaction again, but add no XXXX, the major product will be the same", Bob said.

Explain why Alice's experiment failed and what Bob meant by XXXX. What is the sequence of the product obtained by Alice, and how can Alice fix the problem?

Hint:

Self-complementarity.

Solution:

Upon examining the template sequence, you will notice that it is partially self-complementary. This implies that it can serve as its own template for elongation, as illustrated in Fig. 1. Additionally, it is capable of forming a loop, which will yield the same result.

```
5'-AGGTACACACCGTATATTCAACAAGTACTTCCTTCGACGCGTAGCTACGCG-3'
                                        ||||||||||||||
                                     3'-GCGCATCGATGCGCAGCTTCCTTCATGAACAACTTATATGCCACACATGGA-5'

                                              |
                                              v

5'-AGGTACACACCGTATATTCAACAAGTACTTCCTTCGACGCGTAGCTACGCGTCGAAGGAAGTACTTGTTGAATATACGGTGTGTACCT-3'
                                        |||||||||||||||||||||||||||||||||||||||||||||||||||
                                     3'-GCGCATCGATGCGCAGCTTCCTTCATGAACAACTTATATGCCACACATGGA-5'
```

Figure 3: Self-complementary template and its elongation.

Typically, this type of problem can be resolved by adding a large excess of a primer, which competes with the template and prevents self-hybridization of the latter. However, as you can see, the primer is also self-complementary, meaning it binds to itself and cannot function effectively.

The issue can be resolved if Alice uses a different primer, such as one that is 5 to 6 nucleotides longer.

Linguistics & Applied Sciences

5 points:

The sample corpus below phonetically transcribes select words from an unknown language: (Note: /x/ is the sound in 'loch', /ɣ/ is /x/ but voiced. /ʃ/, /ʒ/, /θ/, /ʊ/, /ə/, 'y' are English 'sh', 'zh', 'th', 'f<u>oo</u>t', 'comm<u>a</u>', '<u>y</u>ell')

1. wöbʒäyʒä
2. tämä
3. ləlärniyän
4. xawzum
5. ləgzəwba
6. narn
7. trney
8. rmoksa
9. nöbʒi
10. kuponənawm
11. ʒüdɣü
12. stliɣi
13. tlofk
14. lworngəθämä
15. büsk
16. küpönenawm
17. blemdʒe
18. yiʃün
19. zudɣum
20. wödliɣlö
21. zutx
22. yiʃülelwornk
23. küpönö
24. närniyälle
25. sayəθmak
26. wʊsun
27. lwornk
28. gwomp
29. kupon
30. brmögʒö
31. smusəθrney
32. tlowgum
33. sayəθmagəvüsk
34. lmusəfʃafs
35. nyögʒef
36. ləzdlʊx
37. ksəwba
38. psafs
39. skfombəxupon
40. tmak
41. lmus
42. xəmts
43. dlʊx
44. wöɣemtʃe
45. tmyärnʃä

(a) You notice that certain transcribed sounds appear only in a certain place in the word, such as at the middle or the end of a word or before a vowel. You also notice that certain sounds may frequently be found together in the same word. These sounds can be referred to as "allophones": the actual pronunciation may change, but all of these sounds relate to a single *phoneme*, or distinguishable sound in the language. Make a <u>reasonable estimate</u> of how many phonemes exist in the language presented above and explain why.

(b) Take a look at the language's *phonotactics* — that is, how sounds fit together within a word and in what order. Group the above sounds into categories. How are the phonotactics of this language different from English? Are there sounds that are shared with English but are ordered in a way that isn't found in English?

(c) Write 3 words that obey the phonotactics of the language above and 3 that <u>do not</u>. Explain your reasoning.

Hint:

No hint this month.

Solution:

(a) In this problem, the featured language contains allophony, where consonants and vowels may change in response to other sounds around them. Even though allophones may be transcribed into English with different letters, these are not registered as 'different' sounds in the featured language.

One type of allophony in this problem is **palatalization**. There are two categories of vowels: category 1 (a ə ʊ o u), which have respective pairs in category 2 (ä e i ö ü). This causes certain consonants (s, z) become (ʃ, ʒ) before category 2 vowels. This leads to the question: which sounds are allophones? Are consonants causing the vowels to change, or vice versa? Based on the fact that some words contain no palatalized consonants but include category 2 vowels, it appears that the *consonants* are allophones because they change around category 2 vowels. Palatalization also changes 'w' to 'y'.

Another type of allophony in this problem is **voicing / devoicing**. Consonants are consistently *devoiced* at the end of words and after the 's-' prefix (p → b, w → f) Consonants are also consistently voiced after prefixes such as 'wö-' (note that this prefix does not appear to change the w consonant, meaning that 'w' and 'y' can technically be considered two separate consonants rather than allophones). Consonants can become both palatalized and voiced / devoiced at the same time. However, some voiced and devoiced versions of the consonants appear in contrastive positions (meaning, both voiced and devoiced consonants can appear at the beginning of words), which means that despite having allophony, they are still separate consonants.

One other type of allophony is that in compound words where two roots are connected with /ə - e/, stop consonants (b, k) may become fricative consonants (v, x) instead. This definitely means that 'v' is an allophone of 'b', not a full fledged consonant (same for t → θ).

Depending on your answers, above, the following consonants and vowels can be counted:

14-15 consonants (b, p, k, g, t/θ, d, x/ɣ, s/ʃ, z/ʒ, w/f(/)y, r, l, m, n)
10 vowels (a ə ʊ o u, ä e i ö ü)

(b) One key difference between this language and English is in its phonotactics — specifically the way that consonants can be combined. Let's define some categories: the stop consonants (b, p, k, g, t, d), fricatives (s, z) and sonorants (r, l, m, n, w). Note: because f is a devoiced allphone of w, it is not grouped the same as other fricatives, despite being one. Many syllable-initial clusters in the feature language (such as 'rm-', 'lw-', 'lm-', 'skf-', 'ks-', etc) are illegal in English. This usually means that consonants or categories of consonants are combined in an unusual order for English. Some things are more similar to English: the 's-' prefix can be added at the beginning of most consonant clusters, and semivowels (such as w) occur right before the vowel. The sound hierarchy can be written as follows:

(s) → (stop consonant) → (fricative consonant) → (r/l/n) → (m) → (w/f/y) → vowel
→ (r/w/y/f) → (m n) → (fricative / stop consonant)

Another aspect of the language's phonotactics is **vowel harmony**, where each word seems to be biased towards category 1 or category 2 vowels, described in the palatalization section of part A. In the case that two words with different vowel groups are compounded together (such as in küpönenawm), the /ə - e/ vowel aligns with the first term but prevents the vowel harmony from spreading to the second part of the word. English does not have vowel harmony, so this is another difference.

(c) Any words that demonstrate the qualities described above count towards this part of the problem.

10 points:

A tiny 4-wheel robot is placed on an airstrip. The airstrip consists of two parallel edges that extend for miles in both directions and is absolutely flat. Both edges are lined with a curb that is taller than the robot itself. The robot can rotate 360 degrees as it moves. It contains touch sensors around the perimeter and a microcontroller that can (1) control the direction and movement of the robot, and (2) store previous position data of the robot. However, it cannot tell its angle with respect to the edge of the airstrip, nor does it know how to turn perpendicularly to the curb. Assume that the robot is placed in a random position on the airstrip and that the width of the airstrip is unknown.

(a) Describe a strategy that will guarantee that the robot hits one of the edges. Try to be as efficient as possible in terms of distance travelled.

(b) If the width of the airstrip is 30 meters, what are the shortest and longest possible distances that a robot could travel using your strategy in part (a) before reaching the curb? What if the width is 300 meters?

(c) Once the robot hits the curb, explain how the robot can keep moving to identify the correct width of the airstrip by hitting the other curb. Try to be as efficient as possible in terms of distance traveled and analyze the best and worst case scenario as in part (b), but in terms of the hypothetical width W.

Hint:

No hint this month.

Solution:

(a) Two key components are important for the search algorithm to reach the wall by covering the *shortest travelled distance*: make sure that the robot doesn't get stuck parallel to the walls, and make sure that the longer the robot searches for the wall, the more likely it is to follow a trajectory that moves quickly away from its initial point.

Many students had points deducted for suggesting a zig-zag which travels a distance that increases by a constant increment after each 90 degree turn. The reason is because of the robot's poor worst case performance; if the robot starts at the middle of the tunnel and moves at a 45° / 135° angle to the tunnel walls, it gets stuck moving down the tunnel while approaching either wall very slowly. This is explained more closely in the Part B solution.

Some students identified a spiral (square or circular) as a potential search strategy, but increased each loop of the spiral by a constant increment. This results in a linear increase in radius. Rather than continuing in a single direction, the spiral can maintain a relatively close distance to the origin and can detect the wall in any direction from the origin. However, the worst-case performance of this spiral may actually match the previous zig-zag strategy; ignoring the given radii of the tunnels, it can be further optimized.

The correct intuition for this problem is to ensure that the search radius increases at a faster-than-linear rate (exponential, logarithmic, Archimedean, golden spirals satisfy this requirement). Some students also opted for concentric loops instead of spirals, which was similarly optimal.

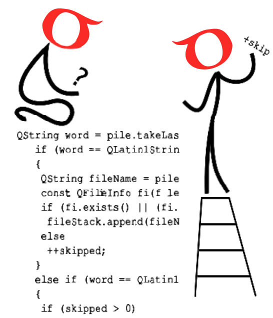
(b) Students were graded according to the accuracy of calculations performed for their chosen strategy. Students who chose spirals as their calculation methods should have received personalized comments on their calculations. For all strategies, the best-case is always very close to 0 meters, assuming the robot starts close against the wall.

For the suboptimal zig-zag strategy, here are the correct calculations for the worst case scenario assuming the robot moves 1 meter, then increases each next leg by 1 meter. When the robot begins at an equal distance between two tunnel walls and turns 45 degrees towards one of the walls before it starts moving, it will travel a total of $n = w * \sqrt{2}$, where w is the width of the tunnel. Since the final zig zag segment will likely hit the wall before the entire segment is completed, all but the last segment result in a total length of $1 + 2 + 3 + ... + n - 1 = n - 1(n)/2$. The length of the final segment can be calculated as the hypotenuse of a triangle whose length is equal to the half-width of the tunnel (w/2) plus the second-to-last segment's distance from the middle of the tunnel to its endpoint ($\frac{n\sqrt{2}}{4}$). This makes the total worst-case length to be: $\frac{1}{2}[w\sqrt{2} + n + n(n-1)] = \frac{1}{2}[w\sqrt{2} + n^2] = \frac{1}{2}[w\sqrt{2} + 2w^2]$.

(c) An important aspect of the problem was that the robot sensor is limited to touch, but not vision — it could not automatically detect whether it's perpendicular to the wall. Here is a summary of some successful strategies suggested by students that were accepted:

(1) After touching the wall, turn 180 degrees, travel a defined distance, turn 90 degrees in a way that reaches the wall again, and construct a right triangle. Use trigonometry to identify the angle of the robot with respect to the wall, then rotate accordingly to make sure the robot is perpendicular, then travel to the other wall.

(2) Travel in a circular arc away from the wall, then move the robot to the middle of the arc and use the arc length to detect the angle. Rotate the robot to be perpendicular, then travel to the other wall.



Computer Science

- Your program should be written in Java or Python-3.
- No GUI should be used in your program (e.g. `easygui` in Python). Other common libraries (e.g. `numpy`, `scipy`) can be used, but **not** in a way that trivializes the problems.
- All the input and output should be done through files named as specified in the problem statement to receive full points.
- Java programs should be submitted in a file with extension `.java`; Python-3 programs should be submitted in a file with extension `.py`. **No `.txt`, `.dat`, `.pdf`, `.doc`, `.docx`, etc. Programs submitted in the incorrect format will not receive any points!**

---

5 points:

The Silver Lake Camp and Retreat Center offers a breathtaking night sky perfect for stargazing. One evening, a group of n campers gathered to observe the Milky Way through a telescope. Each camper has identified a star with some brightness recorded as a positive integer (the higher this integer, the brighter the star).

The campers are competitive, and want to find the brightest star they can. Starting with the second camper, each camper i checks with the previous campers $(1, 2, \ldots, i-1)$ to see if any of them observed a star brighter than theirs. The following then occurs:

- If they find a camper who saw a brighter star, then they set their "goal brightness" to the brightness of the brightest star among the first $i-1$ campers, which they will try to attain at tomorrow's stargazing.
- If they check all the previous campers and find no brighter star, then they decide that their star is bright enough, and set their goal brightness to 0.

Write a program that determines the goal brightness for each camper.

Input specifications

Your program should read the input file `input.txt`, which has the following format:

- The first line contains a positive integer n representing the number of campers.
- The second line contains n space-separated positive integers, where the i'th integer represents the brightness of the star observed by camper i.

Output specifications

Your program should produce the file `output.txt`, which contains n integers corresponding to the goal brightness of each camper.

Examples

Sample Input 1:

```
5
8 3 9 10 5
```

Sample Output 1:

```
0 8 0 0 10
```

Sample Explanation 1:

Campers 1, 3, and 4 each have the brightest star when comparing only the campers before them. Camper 2 sees that camper 1's star is brighter, and camper 5 sees that the brightest star among the campers before him has brightness 10, which becomes their goal brightness.

Sample Input 2:

```
9
1 4 100 9 105 39 90 200 54
```

Sample Output 2:

```
0 0 0 100 0 105 105 0 200
```

Hint:

No hint this month.

Solution:

The 5pt solution is available on the SigmaCamp GitHub repository here:

https://github.com/SigmaCode/POM-QQ/tree/main/CS/2024-2025/November

10 points:

The campers have taken a picture of a small portion of the night sky, which contains n stars. Each star i is positioned at coordinates (x_i, y_i) where x_i and y_i are integers, and has a brightness b_i, which is a positive integer. The campers are trying to find the number of constellations in the picture that are strictly brighter than some threshold.

Constellations are groups of stars that are all connected to each other by a path of "nearby" stars. Two stars i and j, positioned at (x_i, y_i) and (x_j, y_j) respectively, are considered to be "nearby" if the Euclidean distance between their coordinates is less than d:

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} < d.$$

Note that **constellations must consist of at least two stars**. A constellation is brighter than a threshold t if every star in the constellation is **strictly** brighter than t.

The campers are interested in counting the number of constellations of various brightnesses. Write a program that takes in a list of stars and their brightnesses, a distance threshold and a list of brightness thresholds, and computes a list of the numbers of constellations that are brighter than each threshold.

> **! Clarification (November 24, 2024) !** In your program, you should count the number of largest possible constellations that can be formed (e.g. if there is a constellation of 3 stars, that should be counted as a single constellation, rather than four).

Input specifications

Your program should read the input file `input.txt`, which has the following format:

- The first line contains three space-separated positive integers: n, d, m, where:

- n is the number of stars,
- d is the distance threshold for stars to be considered "nearby", and
- m is the number of brightness thresholds.

- The second line contains m space-separated positive integers $t_1, \ldots, t_m$ representing the brightness thresholds.
- The next n lines contain three space-separated integers x_i, y_i, b_i, where:
  - x_i and y_i are the position of star i (**note that x_i and y_i could be negative**), and
  - b_i denotes the brightness of star i

Output specifications

Your program should produce the file `output.txt`, consisting of m space-separated integers, where the i'th integer is the number of constellations that is **strictly** brighter than the i'th brightness threshold t_i.

Examples

Sample Input:

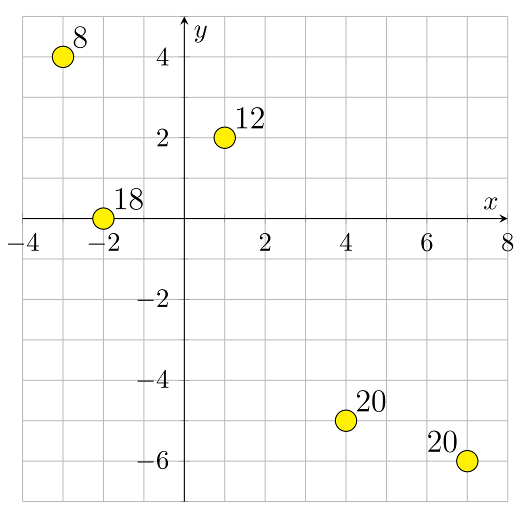
```
5 6 3
5 12 10
1 2 12
-3 4 8
7 -6 20
-2 0 18
4 -5 20
```

Sample Output:

```
2 1 2
```

Sample Explanation:

See a plot of the stars below:



First, disregarding any brightness thresholds, the two stars of brightness 20 cannot be in the same constellation as any of the stars of brightness $8, 12$ or 18, as the star at $(4, -5)$ is more than $d = 6$ away from the stars at $(-2, 0)$ and $(1, 2)$.

For $t_1 = 5$, the stars with brightness $8, 12, 18$ form **one constellation**, and the two stars with brightness 20 form another. Note that we do not count the three pairs of stars (with brightness 8 and 12, 12 and 18, and 8 and 18) as three more constellations. We only count the largest possible constellation of three stars.

For $t_2 = 12$, since the stars at $(1, 2)$ and $(-3, 4)$ do not exceed the threshold (their brightness must be **strictly** greater than 12), they cannot be counted as part of any constellations. Constellations must consist of at least two stars, so the star with brightness 18 is not counted. So, there is just one constellation brighter than $t_2 = 12$, consisting of the two stars of brightness 20.

For $t_3 = 10$, the star at $(-3, 4)$ does not exceed the threshold, but a constellation is still formed by the stars with brightnesses 12 and 18. So, there are two constellations brighter than $t_3 = 10$.

Hint:

Consider representing the stars as nodes in a graph.

Solution:

The 10pt solution is available on the SigmaCamp GitHub repository here:

https://github.com/SigmaCode/POM-QQ/tree/main/CS/2024-2025/November