Problem of the Month

POM

2022-2023

October 2022

MATHEMATICS

5 points:

Alice and Bob have been given a riddle by a conniving evil wizard. The wizard places a hat on each of their heads and writes a natural number on each hat. Alice and Bob can see each other's numbers but not their own. The wizard tells them that, without communicating, they must both write down a *finite* list of guesses for what they think the natural number on their own hat is. How can Alice and Bob guarantee that at least one of them will guess the number on their own hat among their list of guesses?

Hint: No hint

Answer:

Alice and Bob each guess all numbers less than or equal to the other's number.

Solution:

Either Alice or Bob's number is greater, or they are equal. If Alice's number is greater, then Bob's number will be on his list, and the opposite is true. If they are equal then both will have their number in their list.

10 points:

Now the wizard invites Alice and Bob's friends to rise to his challenge. He explains the rules of the challenge, and gives the n players some time to discuss a strategy. He erases Alice and Bob's numbers, places hats on the friends' heads, and writes natural numbers on all n of the hats. The players are allowed to look at the other hats but not discuss what they see. At the wizard's signal, they must all simultaneously guess the *number of players that have the same number as them on their hat (including themselves)*. How can the friends come up with a strategy that will guarantee that at least one person will guess correctly? (As an example, if the numbers on the hats are 17, 100, 17, 17, 100, and 51, then the six players want to guess the values 3, 2, 3, 3, 2, and 1, respectively. Note that only one of them has to get it right to solve the wizard's riddle.)

Hint: What if the group decides on a “leader”? What special role could this leader play to help them accomplish the task?

Answer:

The group decides on a leader. Every member of the group guesses the number of people assuming that they have the same number as the leader. The leader then guesses 1.

Solution:

There are two cases - either the leader is the only person with their number, or one or more people have the same number as the leader. If the leader is alone, then their own guess of 1 will be correct. If not, then all of the people who guess assuming that they have the same number as the leader will be correct. In either case, at least one player fulfills the condition set by the Wizard.

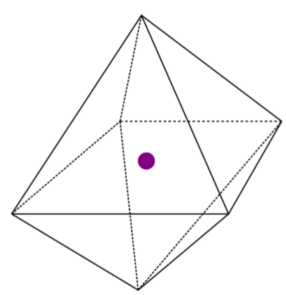
PHYSICS

October's topic is **Electromagnetism!**

Electromagnetic phenomena have been known to humanity for at least three thousand years, but the first (recorded) correct mathematical model didn't start taking shape until the 18th century. In 1785, Coulomb first published the inverse-square law of the electric force. Nearly a century later, the study of classical electromagnetism reached a crescendo with the crystallization of Maxwell's Equations - a set of twelve coupled linear partial differential equations (or 15 nonlinear ones if you want to include matter). Luckily, we will only need a small subset of them in their simplest forms. These equations tell you how changing electric fields create magnetic fields, changing magnetic fields create electric fields, and how charges and currents can generate electromagnetic fields.

5 points:

A point charge of charge q sits in the center of an octahedron of side length a. The octahedron has $total$ charge Q spread evenly around its surface.



(a) What is the total force due to the charge on $each$ face of the octahedron? You may find Gauss's law very useful.

(b) Suppose we allow the octahedron to deform, but we don't allow the material making it up to expand (i.e. to change in surface area). When the system achieves equilibrium, what is the total force with which the charge is pushing the sphere away from it?

Solution:

(a) The best way to tackle problems like these is to break the large (more precisely, extended) objects involved (in this case the octahedron) into tiny bits that we treat as particles. This is a very common technique used in all fields of physics. If A is the total area of the octahedron, then each little bit of the octahedron will have charge $\sigma = Q/A$. The force acting on this bit will be $\vec{F} = \sigma\vec{E}$, where $\vec{E}$ is the electric field where this bit is located.

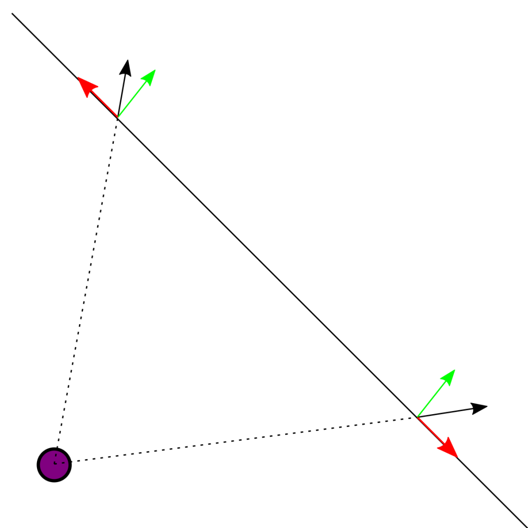


Figure Ph.3.1: The black arrows show the force due to the purple charge on one bit of a slice of the octahedral face. The green arrows show the part of the force perpendicular to the face, and the red arrows show the part parallel to the face. The green arrows will add, and the red arrows will cancel out for points that are related by reflection symmetry, so we only need to keep the perpendicular parts.

The force on a single face will be the sum of all the components *perpendicular* to the face, as the components parallel to the face will cancel out due to symmetry (see figure Ph.3.1). The total sum of the perpendicular component of the electric field acting on all the bits of the octahedron is precisely given by Gauss's law. Since there are 8 faces, we have

$$\vec{E}_{perp} = \frac{q}{\epsilon_0}$$

Summing all of the little bits on *one* of the eight faces results in

$$\vec{F} = \frac{qQ}{8\epsilon_0}$$

(b) The symmetry of the problem implies the octahedron will deform into a sphere. However, the solution to part (a) actually required no information about the precise shape of the octahedron, only its total area (this is the magic of Gauss's law). The total force on the sphere will therefore be

$$\vec{F} = \frac{qQ}{\epsilon_0}$$

10 points:

There's a neat trick we can employ sometimes when dealing with conductors. Conductors are defined as regions of space where the electric potential is required to be constant. For example, consider a point charge sitting next to an infinite conducting plane:

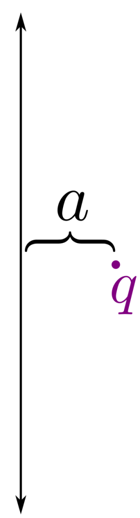


Figure Ph.2

We would like to calculate the electric field due to the charge everywhere to the right of the plane, meaning we would like to find a valid field configuration where everywhere along the plane we have $V(\vec{r}) = constant$. A cheeky hack to accomplishing this is to put a second charge of equal magnitude and opposite sign in the corresponding position on the other side of the plane:

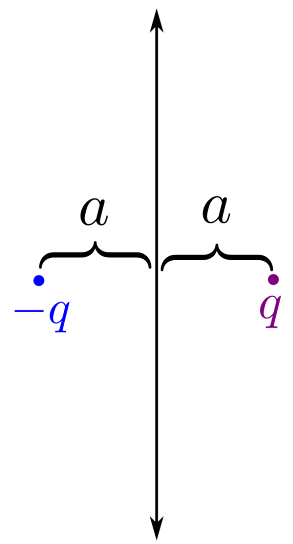


Figure Ph.3

With this charge in place, the condition that the electric potential is constant along the plane is automatically satisfied, and we can entirely forget about it! The correct answer for the electric field configuration to the right of the plane is the sum of the electric fields due to these two charges.

Part a) Prove that the configuration in Figure Ph.3 satisfies the correct boundary condition along the plane.

Part b) We instead place a charge halfway between two infinite conducting planes:

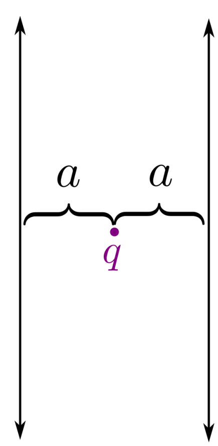


Figure Ph.5

Determine the electric potential everywhere between the two planes.

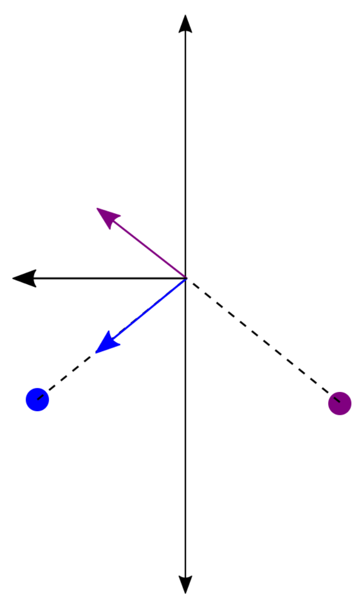
Solution:

(a) We remember that the electric field due to a single charge centered at the origin is given by

$$\vec{E} = \frac{q\,\hat{r}}{4\pi\epsilon_0 r^2} = \frac{q\,(x,y)}{4\pi\epsilon_0 (x^2+y^2)^{(3/2)}}$$

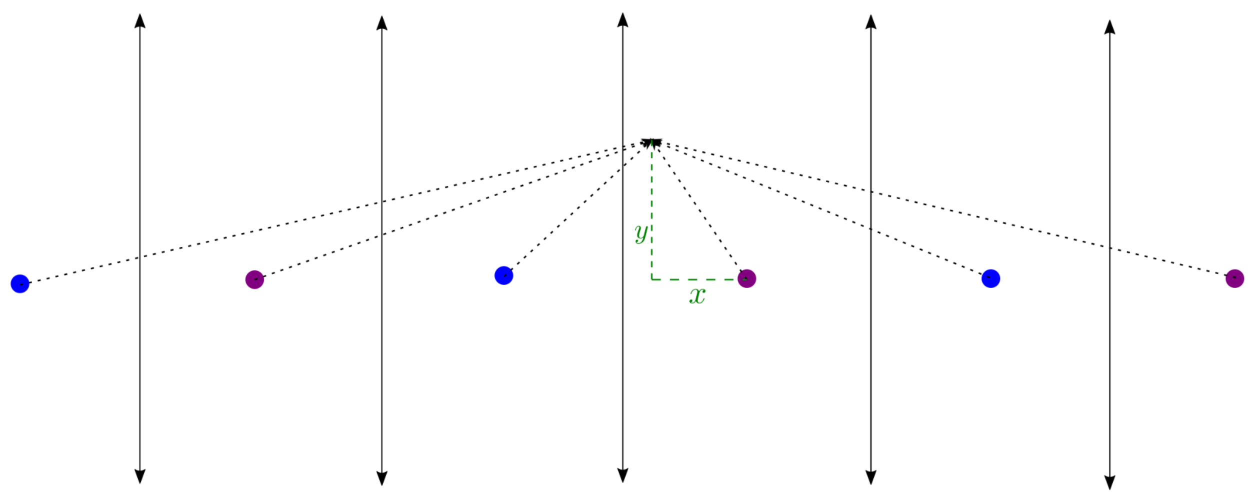
Where r is the distance to the charge and $\hat{r}$ is the unit vector in the radial direction. To translate this equation into human language, the electric field points away from the charge, and its strength falls off as one over the square of the distance (this is known as Coulomb's law). When q is positive, the electric field points away from the charge, and when it is negative, it points towards the charge. Here we assume (without loss of generality) that q is positive.

To show that the configuration in Ph. 3 precisely correctly reproduces the boundary conditions at the conducting plane in the midpoint, we have to show that the electric field in the $\hat{y}$ direction (point upward along the page in the figure) is zero. To assist in this, we choose a random point on the conductor and draw the electric fields at that point due to the positive charge on the right and the negative charge on the right (shown in the following figure).



The point chosen on the line, by symmetry, is equidistant from both charges, so the strength of the electric field will be the same. Also by symmetry, the angle formed between the electric field and the plane are congruent for the purple lines and the blue lines. From this image, it is clear that the $\hat{y}$ components will cancel out, and the remaining $\hat{x}$ components (here drawn in back) will add to form the sum of the two vectors. The field will be perpendicular to the plane at any point, so the boundary conditions are satisfied!

(b) To find the correct configuration, we have to mirror the charges around *both* planes. Every time we mirror a charge around one of the lines, we have to mirror the resulting image around the other one, resulting in the following infinite chain of alternating purple and blue charges:



All that is left is to evaluate the electric field, which is expressed as an infinite sum. We choose our coordinate system so that the middle charge (which we label q_0) is at location $(0, 0)$. Then, if the point we are interested in is at coordinates (x, y). We label charges to the left q_n with negative integer indices n, and charges to the right with positive integers. The value of charge q_n is $(-1)^n q$, and the vector pointing from charge q_n to the chosen point is $(x - na, y)$. We put all this together to determine that the value of the electric field at the point will be

$$\vec{E} = \frac{q}{4\pi\epsilon_0} \sum_{n=-\infty}^{\infty} \frac{(-1)^n (x - na, y)}{((x-na)^2 + y^2)^{(3/2)}}$$

Which is our final answer.

CHEMISTRY

5 points:

When Alice entered her lab, Bob was sitting before the laboratory bench and staring at three bottles. .

"Alice, I need help. I am studying the behaviour of laboratory mice, and I needed to make some incentive for them when a mouse behaves in a certain way. I used sweet water, which I prepared from glucose. I took this bottle (the bottle labelled “D-glucose”), and I prepared a 5% solution. My experiment went well, but, according to the plan of my study, the same series of experiments must be repeated three times, and they must be separated by a two week break. When I started the second series of experiments, I had to prepare a fresh solution of glucose. However, I found that the bottle of D-glucose was empty, so I took another one. The label on the new bottle was partially destroyed, but the letters "-lucose" were seen pretty clearly. To my big surprise, my mice refused to take this new solution, although I prepared the solution of exactly the same concentration. It looked like they didn't like it at all. I checked the melting point of this compounds, and it was exactly 150°. I asked Cecille to analyse it using her mass-spectrometer, and she told me that the molecular mass of this compound was 180 Da, so it would be natural to conclude it is glucose. Why don't my mice like it?”

“It seems I understand the reason, Bob”, - Alice replied.

“Wait, that is not the end of the story. Upon having realized that there is something wrong with that bottle, I took a fresh one. Unfortunately, the label on that bottle was partially destroyed too, but the letters “-ose” were clearly visible. Before making my experiment, I gave a sample to Cecile, and she told me that mass-spectrum showed a mass of 180 Da, so I decided to use this bottle. However, my mice refused to drink this solution either. To understand what is wrong with this solution, I tasted it, and it appeared to be not sweet, but slightly bitter. Can you please explain to me what is wrong with these two bottles?”

“Sure, I can do that, Bob”, - Alice replied. “But you must realize that you made TWO big mistakes, and you must promise me to *never ever* repeat them.”

What mistakes did Bob do, and what compounds were in the bottles #2 and #3?

Hint:

"How would you like to live in Looking-glass House, Kitty? I wonder if they'd give you milk in there? Perhaps Looking-glass milk isn't good to drink…”

Answer:

Glucose, as well as other sugars, is optically active, which means a mirror image of a glucose molecule is not identical to it: D-glucose and its mirror isomer (L-glucose) relate to each other like a right hand and a left hand. That means that D-glucose molecules interact with each other in the same way as L-glucose molecule interact with other L-glucose molecules. And all of them interact identically with other molecules that have mirror symmetry (for example with water).

Therefore, melting temperature (and other physical properties), solubility, molecular mass etc will be identical for D- and L-glucose, so, from chemist's perspective, they are the same compound. However, human enzymes and receptor proteins are, like glucose, chiral: they also resemble a hand, and, therefore, exist in a "D" and "L" forms. Our tongue receptors (and all enzymes of sugar metabolism) binds to D-glucose, but they are as incapable of binding to L-glucose as you are incapable of shaking other person's left hand with your right hand. Therefore, L-glucose is not a sugar for any living organism: it has no sugar taste, and it cannot be digested. In that sense, Alice was absolutely right: looking glass mirror is not good to drink.
The third bottle is a little bit more tricky. Most likely, it is another sugar, mannose. It is also naturally occurring sugar, and it has the same formula $C_6H_{12}O_6$. However, one carbon (C-2) has a different configuration: its hydrogen atom and its OH are swapped. That has a very strange effect: mannose is not sweet, but slightly bitter. It is not a surprise that Bob's mice didn't like it.

10 points:

Next day, when Alice came to the lab's office, she found Bob reading a thick foliant.
"I decided to educate myself, and I started from the beginning. Look, *Britannica* says":

The division of a sample of a substance into progressively smaller parts produces no change in either its composition or its chemical properties until parts consisting of single molecules are reached. Further subdivision of the substance leads to still smaller parts that usually differ from the original substance in composition and always differ from it in chemical properties. In this latter stage of fragmentation the chemical bonds that hold the atoms together in the molecule are broken.

"From that, I conclude that a molecule of some substance is a smallest piece where all atoms are held together by a network of covalent bonds. Am I right?"

"No exactly, Bob. What about DNA?"

"Oh, you are right, Alice. I forgot that DNA is a *double helix*, but the two strands are held together only by hydrogen bonds. Ok, I amend my statement a little bit: *a smallest piece where all atoms are held together by a network of covalent bonds, as well as hydrogen bonds or other weak interactions.* Is it correct now?"

"You are almost right, Bob. However, are you sure that "*a network of bonds*" is an absolutely necessary condition? Is it possible that some smallest piece of matter cannot be split apart without breakage of a chemical bond, but, nevertheless not all parts of that piece are connected to each other by some type of chemical bond? In other words, is it possible that some pieces of a molecule are not connected to each other by any chemical bond or a weak interaction, but the molecule doesn't fall apart? Try to think about that. If you give me an example of such a compound category, I'll give you a can of ginger ale (one can for each type of compound)".

What is a minimal number of cans Bob can earn? Explain your answer.

Hint:

In our macro world, there are many objects that consist of several parts, these parts are not connected to each other via gluing, soldering, welding, are not connected by screws, nails or bolts - but, nevertheless, they cannot be separated. Rings in a chain is one example. Think if that can be possible in the world or molecules.

Answer:

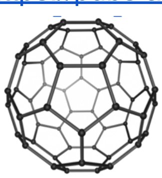
From our everyday's experience we know that two pieces of an object do not have to be glued, welded or soldered together to be inseparable from each other. Indeed, rings in a chain are not linked to each other by any glue, they are not welded together, but the chain still does not fall apart. At the end of the 20th century, chemists synthesized several new classes of compounds that are organized in the same way. Most common examples are cavitands (one part of the molecule forms a cavity where a smaller part is sitting), catenanes (several rings form a chain), and rotaxanes (a doughnut is sitting on a dumbbell). These are examples of mechanically interlocked molecular architectures, and a vast amount of literature is available about it.

Excellent list of examples of such molecules (from problem solution submitted by Emily Snyder):

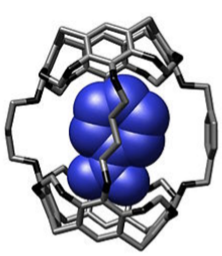
Hydrates:

- Water trapped in ionic lattice

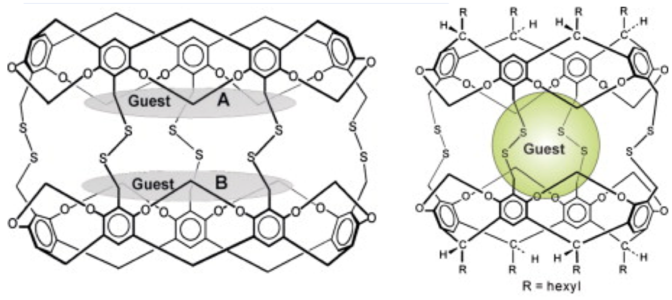
Fullerenes:

- https://pubs.acs.org/doi/10.1021/acs.jctc.1c00662
- 

Carcerands:

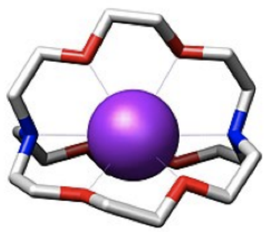
- https://en.wikipedia.org/wiki/Carcerand
- 

Carceplexes

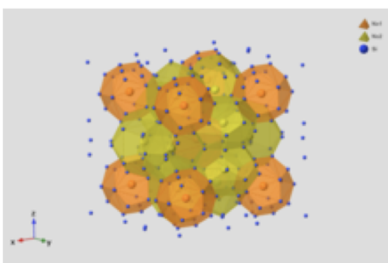
- https://www.nytimes.com/1989/03/21/science/chemists-create-a-new-state-of-matter-one-molecule-inside-another.html
- 

[5]Carceplex 1•Guests [4]Carceplex 2•Guest

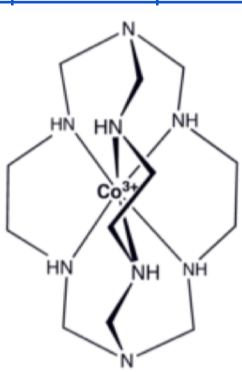
Cryptans

- https://en.wikipedia.org/wiki/Cryptand
- 

Clathrates

- https://en.wikipedia.org/wiki/Clathrate_compound
- 

Clathrochelates

- https://en.wikipedia.org/wiki/Clathrochelate
- 

Cryptophanes

- https://en.wikipedia.org/wiki/Cryptophane

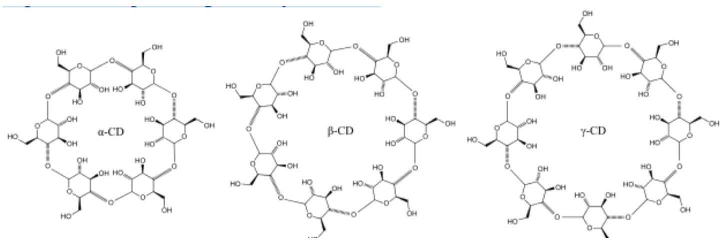
syn

anti

-

Cyclodextrins

- https://en.wikipedia.org/wiki/Cyclodextrin



-

BIOLOGY

Research aimed at uncovering relationships between the brain and behavior depends on an anatomical understanding of brain regions, neural pathways, and other tissues/organ systems. Neuroanatomy and methods used to study the structure of the nervous system have also informed the development of many targeted therapies to treat neurological conditions. Read through the paper here: https://www.cambridge.org/core/services/aop-cambridge-core/content/view/BC705F9216427F98212016A5B224EE7C/S2056469400003855a.pdf/div-class-title-psychiatric-neurosurgery-in-the-21st-century-overview-and-the-growth-of-deep-brain-stimulation-div.pdf on deep brain stimulation, and answer the following questions. The first question is worth 5 points, and the second is worth 10 points.

5 points:

Imagine you are a psychiatrist. Your long-time patient diagnosed with a chronic psychiatric disorder has not responded to pharmacological treatment, and the symptoms are worsening. Given your knowledge about DBS, its advantages and disadvantages, and the state of its current research in the field of psychiatry, what would be a scientific and ethical threshold for implementation of DBS as a potential treatment for your patient? Discuss your reasoning as compared to other available therapies and any ethical considerations.

Solution:

1) choose a side (i.e., yes or no to considering DBS as treatment) (1pt)
2) list at least two reasons for the decision that includes any of the advantages and disadvantages, OR cite previous research that supports their argument (2pts)
3) list at least one ethical consideration and explain it (e.g., potential "manipulation of the mind," misuse by authorities to "control antisocial behavior," mental state of patients to give informed consent for the procedure, lack of ability to conduct randomized controlled trials to identify most beneficial brain target areas, etc.) (2pts)

10 points:

You now have two more patients who are not responding to their respective pharmacological treatments and are curious about DBS. One patient is suffering from major depressive disorder, and the other one is suffering from Tourette's syndrome. Which patient would you expect to respond better to DBS and why? Also specify which areas of the brain would be most beneficial to target for each disorder and justify your answer. Taking into account that a brain operates as a circuit, what possible risks to target its specific components can be anticipated in each of those cases?

Solution:

1) You may pick either syndrome, as long as you justify it accordingly, e.g. Tourette's as it is a motor disorder, and stimulation of target areas can result in direct changes in motor function, and OCD commonly presents alongside Tourette's, and DBS can be used to treat both. Alternatively, for MDD: early clinical trials have shown better efficacy in treating MDD, more target brain areas have been identified for it, etc. (2pts)
2) (Choose any one or two of the following brain areas (there are other possible options, this isn't exhaustive) + explanation of the function of the area)
MDD: Anterior limb internal capsule, Subgenual Cortex, Habenula, Inferior thalamic peduncle (don't give any points for dorsolateral frontal or orbitofrontal cortex, as those are targets for surface stimulation, not deep brain)
Tourette's: Globus Pallidus, Thalamus centromedian nucleus. (Motor cortex is too general an answer here) (4pts)
3) The answer should demonstrate an understanding of the fact that DBS can cause both structural and functional changes. It should explain the functions of parts of the circuit + where the circuit breaks down in patients with these disorders. Possible risks include (for MDD) stimulating the reward system can cause there to be too much dopamine, which could lead to epilepsy, etc. For Tourette's: depressing the motor system can lead to lethargy, dykenesia, and development of obesity and depression, etc. (non exhaustive, any valid risk is okay here!) (4pts)

APPLIED SCIENCES

Surprise! This POM season, Linguistics will be sharing space with a new subject - **Applied Sciences!** Applied Sciences is a very large umbrella that covers all types of engineering (electrical, mechanical, civil) and more.

In honor of the SCREEM (Sigma Camp REcreational Engineering Marathon) event, this month's Applied Science POM is an engineering challenge. A Rube Goldberg is defined by Merriam-Webster dictionary as "accomplishing by complex means what seemingly could be done simply." For another definition, examples, and inspiration, check out rubegoldberg.org (they also hold their own Rube Goldberg contest!).

Your challenge this month is to build a Rube Goldberg that will result in creating a mark with a pencil on a piece of blank paper.
Up to 10 points points will be awarded for elements/technical parts, and up to 5 points will be awarded for creativity, for **a maximum of 15 points possible.**

For reference of how partial credit will be awarded:
5 technical points

- At least 4 distinct elements (in addition to the pencil element), including two different simple machines
- An element is defined as a process that transfers energy from a group of similar objects or a single object to another object
- You may not use a living being as an element

10 technical points

- At least 8 distinct elements
- Total of four simple machines

To submit your solution, record a video (no more than 5 minutes) of your Rube Goldberg running. Please take the video in one continuous shot, and say your name at the beginning of the video.
Submit a word document or pdf to the 10 point problem on the POM website: include a **link to your video** (upload it to youtube/google drive/your choice of filesharing service), **any explanation you would like to include** (no points will be deducted for not including it, but it may help you in earning points), and whether or not you would allow us to use your video in a compilation video (will not affect your score).

Hint: please remember to check the sharing settings on the video you submit!

Solution: Thank you everyone for your amazing submissions! Here are some of the fantastic Rube Goldberg machines we received:
https://drive.google.com/file/d/1NK_Q5Ysxm3Omd2xqIVGq4IvznkBGUepr/view

COMPUTER SCIENCE

Thank you to everyone who attended our lecture introducing basic computer science concepts! We hope you enjoyed it, and recordings will be posted soon if you weren't able to make it or you'd like to revisit a session. We plan to have another lecture next month. The CS POM team has also put together a list of their favorite CS resources, from beginner to advanced, which you can find here. As a fun note, you might've noticed that Alice and Bob pop up in a lot of POM problems, and their origin comes from a 1978 cryptography paper! If you're interested in learning more about them you can do so here.

- Your program should be written in Java or Python-3
- No GUI should be used in your program: eg., easy gui in Python
- All the input and output should be via files named as specified in the problem statement
- 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 incorrect format will not receive any points!

5 points:

After relentless begging and countless favors, Alice finally got access to her older brother's Minecraft account, and has begun making her own circuits with Redstone Dust, a conductive powder that can be placed on the ground and can transport an electric signal. Generating electricity in Minecraft is awfully easy; it can be accomplished with the single push of a button. Alice's objective is to light a lamp, which she places on another point in the grid. For a lamp to be lit, it must either be adjacent to a button, or adjacent to Redstone Dust that carries a signal. For Redstone Dust to carry a signal, it must be placed adjacent to a button, or to another Redstone Dust that carries a signal. Redstone Dust **cannot** transmit any signal diagonally. Each Redstone Dust can have **only two neighbors** in a valid circuit, counting the button, the lamp and adjacent Redstone Dusts (only horizontal and vertical adjacency counts; diagonal does not).

Alice has asked her older brother to analyze one of her circuits and determine whether the circuit will light when the button is pressed. If the button lights, the circuit is valid. On a grid with dimensions a x b, the button will be represented as "B", the lamp as "L", Redstone Dust as "#", and empty space as ".". **It is guaranteed that there will be exactly one button in the top left and one lamp in the bottom right on each grid.**

The following is a valid circuit example:

```
B...
##..
.#..
.#..
.##L
```

Note that a button can directly power a lamp if it directly beside it, therefore the following is also valid:

```
BL
```

Write a program that receives a, b, a circuit grid, and determines whether or not the circuit is valid. Your program should receive the input file **input.txt**, which will have a and b separated by a space on the first line, and the circuit grid on the following lines.
Example input file:

```
5 6
B....
###..
..#..
.....
..##.
...#L
```

Your program will produce the output file **output.txt**, which will contain “VALID” if the circuit is valid, and “INVALID” if the circuit is invalid.
Example output file:

```
INVALID
```

In the example above, there is no way for the signal to travel from the third line to the fifth, so the circuit is invalid.

Solution:

```
def check_north(i,j):
    '''Checks whether the cell in row i and column j has a # or a B
    above it.'''
    if i == 0 or (full_grid[i - 1][j] != "B" and full_grid[i - 1][j] != "#"):
        return 0
    else:
```

```
            return 1

def check_east(i,j):
    '''Checks whether the cell in row i and column j has a # or an L to
    the right of it.'''
    if j == width - 1 or (full_grid[i][j + 1] != "L" and full_grid[i][j + 1] != "#"):
        return 0
    else:
        return 1

def check_south(i,j):
    '''Checks whether the cell in row i and column j has a # or an L
    below it.'''
    if i == height - 1 or (full_grid[i + 1][j] != "L" and full_grid[i + 1][j] != "#"):
        return 0
    else:
        return 1

def check_west(i,j):
    '''Checks whether the cell in row i and column j has a # or a B to
    the left of it.'''
    if j == 0 or (full_grid[i][j - 1] != "B" and full_grid[i][j - 1] != "#"):
        return 0
    else:
        return 1

def is_valid(i,j):
    '''Checks whether the cell in row i and column j is valid.'''
    if i == 0 and j == 0:
        #If we're looking at the top left corner, then we should expect
        # to see a button that has exactly one # (or L) touching it
        if full_grid[i][j] != "B":
            return False
        elif check_east(0,0) + check_south(0,0) != 1:
            return False
        else:
            return True
    elif i == height - 1 and j == width - 1:
        #If we're looking at the bottom right corner, then we should expect
```

```
        # to see a lamp that has exactly one # (or B) touching it
        if full_grid[i][j] != "L":
            return False
        elif check_north(i,j) + check_west(i,j) != 1:
            return False
        else:
            return True
    else:
        if full_grid[i][j] == "#":
             if check_north(i,j) + check_east(i,j) + check_south(i,j) + check_west(i,j)
!= 2:
                return False
            else:
                return True
        elif full_grid[i][j] == ".":
            return True
        else:
            return False


def run():
    dimensions = input_file.readline().strip().split(" ")
    # Reads the first line of the input file, removes any
    # whitespace, and splits it
    if len(dimensions) != 2:
        return "INVALID"
    # If the number of dimensions given in the first line of the input
    # file is not two, the circuit grid is invalid
    if not(dimensions[0].isdigit() and dimensions[1].isdigit()):
        return "INVALID"
    # Checks to make sure both dimensions are integers
    global width
    width = int(dimensions[0])
    global height
    height = int(dimensions[1])
    # Hey, we know the width and the height now, so let's write them
    # down so that we can use them later
    global full_grid
    full_grid = []
```

```
    for i in range(height): # The number of lines in the file should be
                            # equal to the height we were given
        row = list(input_file.readline().strip())
        # Let's read the next row, strip off all that annoying
        # whitespace, and make it into a list of individual characters
        if len(row) != width:
            return "INVALID"
        # If the number of characters in the row isn't equal to the
        # width we were given, then we should riot (meaning say the
        # circuit is invalid)
        full_grid.append(row)
    if input_file.readline() != "":
        return "INVALID"
    for i in range(height):
        for j in range(width):
            if not is_valid(i,j):
                return "INVALID"
    return "VALID"
with open("input.txt", "r") as input_file:
    with open("output.txt","w") as output_file:
        output_file.write(run())
```

Hint:

No hint

10 points:

Bob is sitting in front of a red, a blue, a yellow, and a green buttons, and a screen displaying an integer. When a button is pressed, the integer on the screen is updated following these rules:

- When Bob presses the red button, the integer is subtracted by 1.
- When Bob presses the blue button, the integer is multiplied by 2.
- If Bob presses the yellow button and the integer is odd, nothing happens. If Bob presses the yellow button and the integer is even, the integer is divided by 2.
- When Bob presses the green button, the integer is incremented by the number of prior button presses. For instance, if Bob presses the green button first, nothing will happen. If Bob presses it after the 10 button presses, the integer will be incremented by 10.

Write a program to help Bob calculate the least number of button presses possible to go from N to M, where N is the initial integer on the screen and M is the target integer. Your program should receive the input file **input.txt**, which will consist of N and M separated by a space. Example input file:

3 7

Your program should produce the output file **output.txt**, which will contain a single integer with the minimal amount of button presses.
Example output file:

2

Bob can press the blue button followed by the green button to go from 3 to 7.

Solution:

```
with open("input.txt") as output_file:
   start, goal = [int(num) for num in output_file.readline().split()]

# tracks the number of presses for calculating the value if the green button is pressed
presses = 0
# haven't used sets before? Read up here:
# https://www.w3schools.com/python/python_sets.asp
# starts a set of possible options from button presses, at first it's just the start
# number
current = set([start])
while goal not in current:
   # makes a temporary set to hold all the possible numbers that can be reached from a
   # button press
   new_values = set()
   # iterates through all of the previous possibilities to find the numbers that can
   # be reached from them
   for state in current:
       new_values.add(state - 1)  # red
       new_values.add(state * 2)  # blue
       new_values.add(state // 2 if state % 2 == 0 else state)  # yellow
       new_values.add(state + presses)  # green
   current = new_values
   presses += 1

with open("output.txt", "w") as output_file:
   print(presses, file=output_file)
```

Hint:

Imagine layers of numbers, where the first layer of branches are the numbers that can be reached by one press, the second layer is the numbers that can be reached by two presses, etc. How are the layers connected? Remember that the program only needs to find the minimum number of presses, not the specific path to follow!