SigmaCamp's *Problem of the Month* Contest

# OCTOBER 2025

**IMPORTANT:** For the 2025-2026 season, POM is experimenting with two format changes:

- **Video solutions** must be submitted for each problem.
- Each month will include a **project** (worth 30 points) in place of one or two subjects.
- We now have monthly **POM office hours** where you can ask questions about this month's POM problems! See sigmacamp.org/ pom/office-hours for details.

## Video Submissions

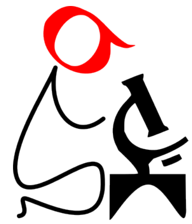
Starting this year, **all solutions must be accompanied by short videos explaining your work.**

- Videos must be at most **2 minutes long** for 5pt/10pt problems, and at most **6 minutes long** for projects.
- **Solutions must be narrated**, but you do not need to show your face. Acceptable formats include:
  - Screensharing slides or a drawing app (e.g., MS Paint) with narration.
  - Recording a whiteboard, paper, or easel with narration (contents may be pre-written).
  - Speaking directly to the camera.
- **Submit videos as links** (Google Drive, Youtube, Dropbox, etc.). Extra requested files may be submitted as a single PDF file per problem.
- For coding problems, submit your code along with a video explaining your submission. Only **Python-3** (`.py`) and **Java** (`.java`) code submissions are accepted.

Please see sigmacamp.org/pom for full details on the 2025 POM format change.

# Project (30 points)

# Biology and Chemistry



## Introduction

In the history of scientific experimentation, some of the most transformative discoveries arose from experiments whose designs were strikingly simple yet methodologically rigorous. Two well-known examples are Gregor Mendel's pea plant crosses and Robert Boyle's investigation of gas pressure and volume.

Mendel used garden peas to study inheritance. By isolating traits such as seed shape or flower color, he controlled fertilization between true-breeding plants and meticulously counted the traits of thousands of offspring across generations. What made his work rigorous was not the complexity of tools, but the precision in experimental design. From this, Mendel derived statistical ratios that revealed the particulate nature of heredity, earning him the title of the "father of genetics."

Similarly, Boyle's 17th-century gas experiments relied on a J-shaped glass tube and mercury. He trapped a pocket of air at one end and varied the pressure on it by adding mercury, while measuring the corresponding volume change. Through repeated trials, Boyle demonstrated a precise mathematical relationship: pressure multiplied by volume is constant when temperature is held steady. This conclusion, known as Boyle's Law, emerged from a methodology built on simple apparatus, systematic variation, and quantitative recording.

Often, experiments are designed using the "scientific method." The scientific method is defined as a "dynamic process that involves objectively investigating questions through observation and experimentation."[1] Additional resources about the scientific method can be found here.

## Instructions

There is an experimental and theoretical section to this project. The experimental section is worth 18 points, and the theoretical section is worth 12 points. There are opportunities for partial credit in all parts of the project.

The solution will be graded based on the quality of the experiment, experimental design, and the quality of the work itself.

If your experiment fails, do not worry. There is still an opportunity to receive points. Refer to the section after the theoretical questions for questions about your experiment.

---

## Experimental Portion

In this project, you will design and conduct a controlled investigation to test how effectively yeast produces carbon dioxide when provided with different beverages as its primary sugar source. Your task is to test at least three different liquids of your choice and determine which one allows yeast to ferment most effectively. You are required to collect quantitative data and establish a result based on your data.

You are encouraged to be creative in how you design and conduct a controlled investigation to test how effectively yeast produces $CO_2$ during the fermentation of different beverages as its only sugar source. We will be looking for scientific rigor, optimization, and implementation. We encourage using modern scientific approaches based on realistic experimental designs.

[1] https://extension.unr.edu/publication.aspx?PubID=4239

## Theoretical Questions

Each question below is worth 3 points.

1. Different drinks with the same sugar content could produce different volumes of $CO_2$. If that occurs, how could you explain this?
2. What are the ideal parameters for fermentation, and how do sugary carbonated beverages deviate from them?
3. Equate the chemical reaction occurring.
4. Among the processes of glycolysis, fermentation, and respiration, which results in the greatest change in volume, and why?

## If your experiment fails

There is still the opportunity to receive points, even if something goes wrong with your experiment. Recall that a significant part of scientific experiments fail, and drawing correct conclusions from the experiments that went wrong is a part of a scientist's job. If your experiment fails, analyze possible reasons and, if you can, propose solutions: you may also get some points for this. Your analysis may include the answers to the following questions:

1. What could you have done to improve your experiment?
2. Describe what you did and some reasons why it might have failed.
3. If you decide to redesign your experiment at any time, please submit all your results and describe what you changed.

# Mathematics

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

## 5 points:

The number $20! = 2\,432\,902\,008\,176\,640\,000$ has 41040 divisors. How many of them are odd?

## 10 points:

(a) A stash of several identical watermelons was stored at SigmaCamp. At lunch, the SigmaCampers collectively ate 10 of the watermelons. Everyone present received the same amount of watermelon.

At snack time later that day, some campers were still full from lunch, but a team consisting of only 11 of the campers wanted more watermelon. Each of them ate half as much watermelon as they did at lunch. By the end of snack time, all the remaining watermelons were gone. How many watermelons were stored initially?

(b) Write your own problem that uses the same idea as the problem above to create a solvable problem with exactly one solution. Your problem cannot use any of the three numbers from the original problem (10, 11, half). Creative scenarios that do not involve consuming food or drink are encouraged.

# Physics

## 5 points:

There is a plate attached to the bottom of a long tube by a Hookean spring of spring constant $k$ (if you don't know what this means, have a look at the Additional Information section below). A ball of mass $M$ rests on the plate, as shown in Figure 1A. For this problem you may neglect friction and air resistance, and you may assume the mass of both the plate and the spring to be 0. Let the height of the spring at rest with no mass on it correspond to $y = 0$.

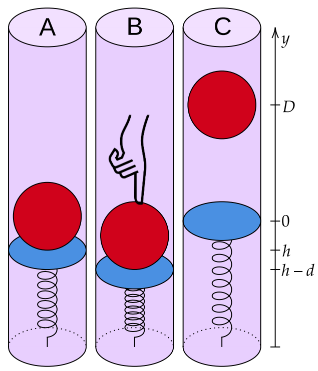



Figure 1: Diagram for the Physics 5 pt problem

(a) (0 points) List the forces on the ball, their magnitudes, and their directions. Draw a free body diagram showing all of them. What is the total force? What condition must hold for the ball to be at rest?

*This question is worth 0 points because it is purely here to get you warmed up. The answer is presented at the bottom of the problem, but try to do it yourself before looking.*

(b) (1 point) At what height $h$ does the ball rest? What is the gravitational potential energy of the ball relative to $y = 0$? What is the energy stored in the spring? What is the kinetic energy of the ball?

Now imagine that someone comes along and pushes the ball down by an additional distance $d$. The ball is then released and shoots up into the air.

(c) (3 points) What is the maximum height $D$ the ball reaches?

However, in no real spring does the force perfectly scale linearly with the displacement. Assume instead that

$$F(y) = -ky - ly^2.$$

In this case the potential energy due to this kind of spring is given by the usual quadratic Hookean term, plus a non-Hookean correction:

$$V = \underbrace{\frac{1}{2}ky^2}_{\text{Hookean}} + \underbrace{\frac{1}{3}ly^3}_{\text{non-Hookean}}$$

For realistic springs, $l$ is usually very small and the non-Hookean terms can be ignored.

(d) (1 point) What is the maximum height the ball now reaches?

**Answer to (a)** There are two forces on the ball: the gravitational force pointing downwards and the spring force pointing up. The magnitude of the spring force will be $F_s = -hk$, by Hooke's Law (remember that $h$ is negative). It will be pointing up. The magnitude of the gravitational force will be $F_g = mg$, and it will be pointing down.

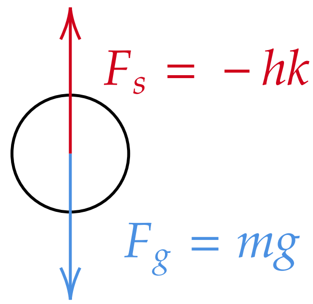



Figure 2: Free body diagram for part (a) of the 5pt physics problem.

The total force is $\vec{F}_{\text{net}} = \vec{F}_s + \vec{F}_g$. The horizontal component will be $F_{\text{net},x} = 0$ and the vertical component will be $F_{\text{net},y} = F_s - F_g = -hk - mg$. In order for the ball to be at rest, the acceleration must be 0, so the net force must also be 0. Thus, $-hk - mg = 0$.

**Additional Information.** Springs and objects that can be modeled like springs, can be frequently found in many physical systems. The simplest type of spring, called a "Hookian spring", exerts a force if it is stretched or compressed via the following equation

$$F = -k\Delta x,$$

where $k$ is the spring constant and $\Delta x$ is how much the spring is stretched/compressed[2] from its equilibrium position.

It is often useful to think about these quantities called the kinetic and potential energy of the mass on a spring, denoted by $T$ and $V_s$ respectively:

$$T = \frac{1}{2}mv^2, \quad V_s = k(\Delta x)^2,$$

where $m$ is the mass of the item on spring, $v$ is its speed and $\Delta x$ is once again how much the spring is stretched from its equilibrium position.

It turns out, that for systems where force only depends on the position of the particle, the total energy of the system is conserved over time[3]:

$$E_{tot} = T + V_{tot}.$$

For example, in the case of a mass on a spring, if at every point in time we measure the speed $v$, how much the spring is stretched $\Delta x$ and use it to compute $E_{tot}$ as above, we always get the same number [4].

The potential energy a one particle system has due to gravity is given by

$$V_g = mgh,$$

where $m$ is the mass of the particle, $g$ is the gravitational constant and $h$ is the elevation of the particle above ground (or any fixed reference point). In the case of multiple forces acting on one particle, $V_{tot}$ is the sum of potential energies due to each force involved.

[2] We assume the convention that positive $\Delta x$ refers to the spring being stretched and negative to the string being compressed by $\Delta x$.

[3] You can check this for the spring by considering some small time interval $\Delta t$, and comparing the energy of the system at time $t$ with its energy at time $\Delta t$. It is useful to make the approximations that $v(t+\Delta t) = v(t)+\Delta t \cdot a(t)$ and $x(t+\Delta t) = x(t)+v(t)\Delta t$ and seeing that the energy essentially does not change over this arbitrarily small time change $\Delta t$.

[4] Note that this number is the same as the same system evolves over time. Energy can be different, for a system with a different starting $v$ and $\Delta x$.

## 10 points:

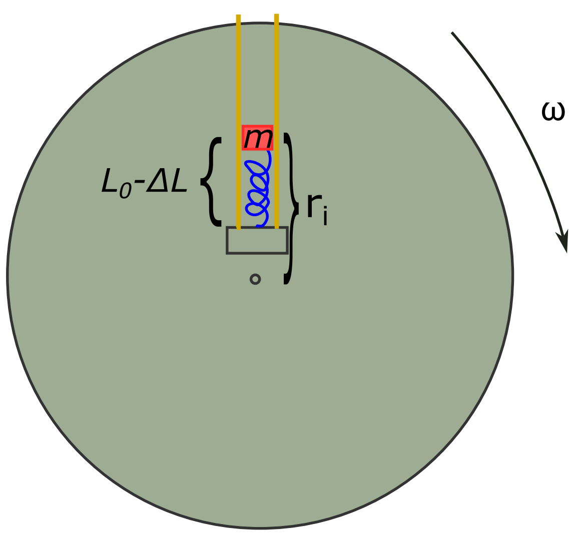
Consider the system shown in Fig. 3, consisting of a merry-go-round with a mass-spring setup. The surface has a static and and kinetic coefficient of friction denoted by $\mu_s$ and $\mu_k$, respectively. A mass $m$ is attached to the spring (which has spring constant $k$) that is anchored to a short vertical wall. The mass can only slide radially along frictionless rails on the merry-go-round.

Initially, the spring is compressed by a distance $\Delta L$ relative to its rest length $L_0$. The only force preventing the spring from uncompressing is the force of static friction on the mass. Taking into account the distance of the wall from the center of the merry-go-round and the spring's compression, the mass starts at an initial radius $r_i$.

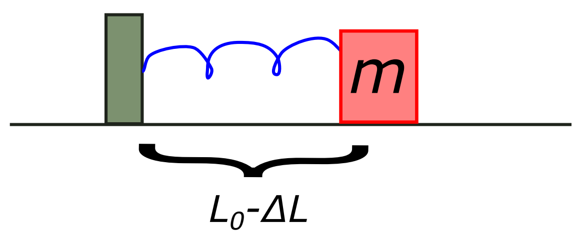
The merry-go-round starts at rest, and gradually accelerates.

**Part (a)**: Find the angular velocity $\omega$ the merry-go-round must accelerate to for the static friction to be overcome, causing the mass to start slipping.

**Part (b)**: The merry-go-round is set to spin at the fixed angular velocity $\omega$ you solved for in the previous part. As the mass begins to move, the spring decompresses, but the mass remains permanently affixed to the spring's end. Find the maximum radius $r_f$ that the mass reaches.




(a) A top-down view of the system. A mass $m$ is loaded onto a spring of constant $k$ compressed a length $\Delta L$ below its rest length $L_0$. The mass sits at a distance $r_i$ from the center. The orange lines are frictionless rails attached to the merry-go-round that keep the mass only moving in the radial direction.




(b) A side view of the system in figure (a). The system starts off compressed by length $\Delta L$ from its equilibrium length. The surface is rough, with a coefficient of static friction $\mu_s$ and kinetic friction $\mu_k$

Figure 3

## Linguistics & Applied Sciences

### 5 points:

Using up to 24 of 3” by 5” lined paper index cards, and one standard 40 mm ping-pong ball, construct a “boat” that will carry as much weight as possible while floating in water. Other conditions:

- Your boat does not need to resemble an actual boat. It does need to float in water.
- No other materials can be part of your boat, including no glues or adhesives.
- You can cut or tear the cards and the ball as you want, if you want.
- As weights, you should use any coins you have access to. As part of your solution, include the denominations and quantities of coins, look up the weights on the internet and figure out the total weight to the nearest 0.1 gram.
- The weights need to rest on top of the boat and be visible.
- The weights can be placed on the boat before or after the boat is placed in water. After the final weight is placed and the boat is floating (whichever is later), you have to show in your video that your boat stays floating and upright and supports the weights for at least 30 seconds.
- Make sure that the boat is seen to not touch the bottom of your water container.
- Document as much of the construction process as you can in your video.

### 10 points:

Here are 7 sentences from a language and their corresponding English translations. **These are ALREADY MATCHED, you do not need to match them.**

| | | |
|---|---|---|
| 1. Jae soaurbaedahlbvi? | ⟷ | 1. *They are teachers?* |
| 2. Sejebeirmaer ae sohrgosohrgosohrg. | ⟷ | 2. *The girl is most likely still snoring.* |
| 3. Le lipa pihmaer. | ⟷ | 3. *You see the woman.* |
| 4. Sae saeverever. | ⟷ | 4. *We are most likely not dancing.* |
| 5. Soaurdaedoahlmaer e pap. | ⟷ | 5. *The worker talks.* |
| 6. Je saebaedahl. | ⟷ | 6. *They don’t teach.* |
| 7. Serihbvi ae saesohrg. | ⟷ | 7. *The children most likely don’t snore.* |

Translate the following sentences into English and explain your reasoning:

(a) Pihbvi ae soaurdaedoahlbvi?

(b) Se saelipa soaurbaedahlmaer.

Translate the following sentences into the language and explain your reasoning:

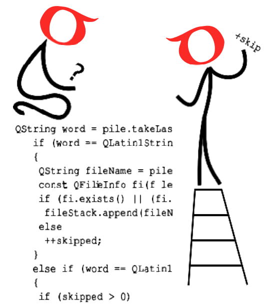
(c) *The girls are still not dancing.*

(d) *They most likely work.*

Explain how to form the following English words in the language:

(e) *“talking”*

(f) *“dancers”*




# 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).
- All the input and output should be done through 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 the incorrect format will not receive any points!**

---

## 5 points:

The Sigma Fresh rap battle is run as a round robin: every team competes against every other team once, and every such match has a winner and a loser (no ties).

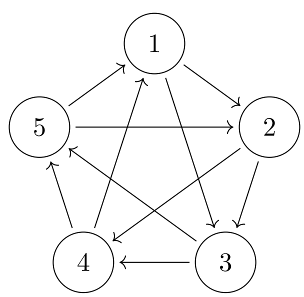
The results can be pretty confusing to interpret. However, you notice the following funny pattern: no matter what the results are, there always seems to be an ordering of the teams so that *every team in the ordering beat the team immediately before it*. That is, one can order the teams in a list $t_1, ..., t_n$ so that each team appears once in the list and, for every $i$ from 1 to $n-1$, team $t_i$ lost to team $t_{i+1}$. Such an ordering is called *Hamiltonian*, because Alexander Hamilton loved rap battles[5].

For example, suppose $n = 5$ and the ten match results are as follows, where a pair $(i, j)$ means that team $i$ lost to team $j$.

$$\Big((1,2),(1,3),(2,3),(2,4),(3,4),(3,5),(4,5),(4,1),(5,1),(5,2)\Big).$$

These results are pictured below (where an arrow $i \to j$ indicates that $i$ lost to $j$). They admit several valid Hamiltonian orderings, including

$$(1,2,3,4,5) \qquad \text{and} \qquad (4,5,1,2,3) \qquad \text{and} \qquad (2,4,5,1,3).$$




**(5 points)** Write a program that takes a set of results and an ordering of the teams, and checks if the results are complete and the order is Hamiltonian.

---

[5] This detail may not be strictly true; see instead William Rowan Hamilton.

**Input** Your program should read the file `input.txt`. Teams are labeled $1, \ldots, n$.

- *Line 1:* Two integers $n$ and $m$, where $n \geq 2$ is the number of teams and $m \geq 0$ is the number of reported match results.
- *Lines 2 to $m+1$:* Each line has two integers $u$ $v$, separated by a space, with $1 \leq u, v \leq n$ and $u \neq v$, indicating that team $u$ lost to team $v$.
- *Final line:* $n$ space-separated integers $t_1\ t_2\ \cdots\ t_n$ — a proposed ordering of the team.

**What to Check**

- *Complete?* The results are *complete* if and only if every unordered pair $\{i, j\}$ with $1 \leq i \neq j \leq n$, one of $(i, j)$ or $(j, i)$ appears among the $m$ lines, but not both.
- *Hamiltonian?* The ordering $t_1, \ldots, t_n$ is *Hamiltonian* if it includes all $n$ teams without repetitions and, for every $i = 1, \ldots, n-1$, the pair $(t_i, t_{i+1})$ appears among the results (that is, $t_i$ lost to $t_{i+1}$).

**Output Format** Write exactly two lines to the file `output.txt`

```
complete YES|NO
hamiltonian YES|NO
```

**Examples**

*Example 1 — complete and Hamiltonian*

```
5 10
1 2
1 3
2 3
2 4
3 4
3 5
4 5
4 1
5 1
5 2
1 2 3 4 5
```

*Correct output*

```
complete YES
hamiltonian YES
```

*Example 2 — complete but not Hamiltonian*

```
5 10
1 2
1 3
2 3
2 4
3 4
3 5
4 5
4 1
5 1
5 2
1 3 2 4 5
```

*Correct output*

```
complete YES
hamiltonian NO
```

*Example 3 — incomplete but Hamiltonian ordering still holds*

```
5 9
1 2
1 3
2 3
2 4
3 4
3 5
4 5
4 1
5 2
1 2 3 4 5
```

*Correct output*

```
complete NO
hamiltonian YES
```

*Example 4 — invalid permutation for the ordering*

```
4 6
1 2
1 3
1 4
2 3
3 4
4 2
1 2 2 4
```

*Correct output*

```
complete YES
hamiltonian NO
```

## 10 points:

After reflecting on it, you realize this is not a fluke: a Hamiltonian ordering will exist for any set of tournament results! One way to prove this is by via an algorithm. Specifically, write a program that, given a set of match results, *always* finds a Hamiltonian ordering of the teams.

**Input** Your program should read the file `input.txt`. It has exactly the same format as in the 5pt, except that there is no final line describing an ordering. **You may assume these results are complete.**

**Output** Your program should write a space-separated Hamiltonian ordering of the teams to the file `output.txt`. (There may be several correct Hamiltonian orderings, you may output any of them.)

*Example:*

```
5 10
1 2
1 3
2 3
2 4
3 4
3 5
```

```
4 5
4 1
5 1
5 2
```

*Correct output*

```
1 2 3 4 5
```