SigmaCamp Qualification Quiz 2019

Dear prospective SigmaCamper,

To complete your application to SigmaCamp, you need to:

1. Fill out the registration form on our website:

   sigmacamp.org/2019/apply

2. Submit your solutions to the Qualification Quiz, which you will find on the pages that follow. There are two problems from each of the five main disciplines at Sigma – math, physics, chemistry, biology and computer science. The second problem in each category was designed to be more challenging than the first problem. You are not expected to solve all the problems. We will evaluate your quiz submission based on your approach to the problems and the quality of reasoning. You can use the Internet, books and even help from someone else but **state precisely what sources you have used to solve each problem**. Also, please do not collaborate with other applicants. Your solutions can be handwritten or typed, and the file name must contain your own name (e.g. Chemistry.1.JohnSmith.pdf).
   Note: if you were accepted to Sigma through PoM, you do not need to submit the QQ.
3. Write an brief essay telling us about yourself and your math/science interests.
4. Obtain two letters of recommendation: one from a Mathematics or Science Teacher and one from an adult who knows you personally. If you are a returning camper, you are not required to submit letters of recommendation – we already know who you are! Recommendation letters can be submitted online via our website or returned to you in a sealed envelope. Detailed guidelines for recommendations are on the website.

We will email you a confirmation upon receiving your application.

The application deadline is April 18, 2019. We will notify applicants regarding an acceptance decision no later than **May 1st**.

Good luck with your application!

Math

Problem 1.

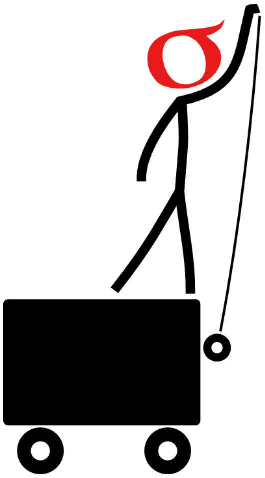
Calculate the following product:
$(1-\frac{1}{2^2})\bullet(1-\frac{1}{3^2})\bullet\ldots\bullet(1-\frac{1}{2019^2})$

Problem 2.

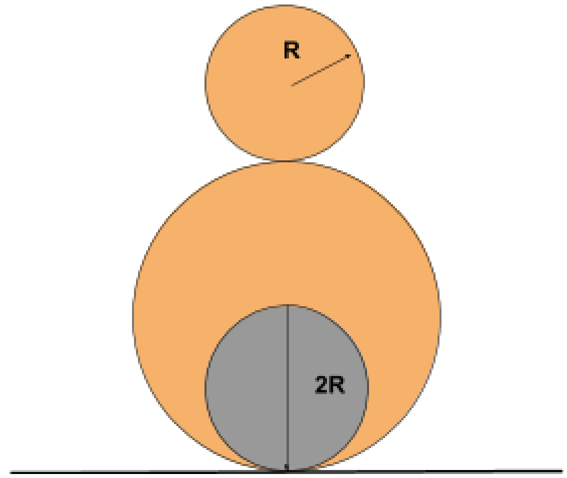
N = 10...01
^ 2^{2019} zeros total

Prove that N = 10...01 is composite and find one of its nontrivial (not 1 or N) factors.

Physics

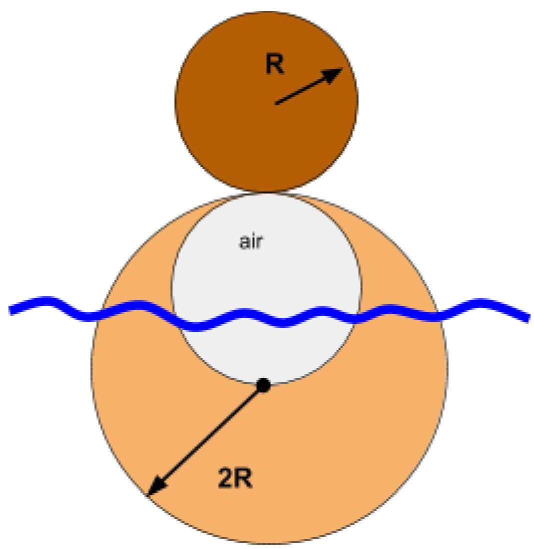


Problem 1.



A roly-poly toy known in Russia as the nevalyashka ("untopply") or van'ka-vstan'ka ("Ivan-get-up"), which is similar to Japanese okiagari-koboshi (okiagari means "to get up (oki) and arise (agari)") is made of two wooden spheres, a bigger one of radius 2R and a smaller one of radius R, glued together as shown in the Figure. A spherical cavity made at the bottom of the larger sphere (opposite to the "head", the smaller sphere) is filled with material that is denser than wood, so that the doll stands up when tilted. Find the minimum density of the filler material relative to the density of wood in order for the doll to work (stand up when tilted). In other words, how much denser should the material filling the cavity be compared to wood for the toy to be a true nevalyashka?

Problem 2.

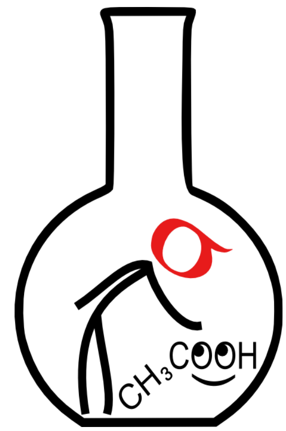


A floating toy consists of two spheres of radius 2R and R, respectively, glued together (See Figure). The bigger sphere is made of wood with density 0.6 g/cm^3.

In order for the toy to float in water with the smaller sphere pointing up, a spherical void of radius R is made in the upper half of the larger sphere, and filled with air. The density of water is 1g/cm^3, density of air can be neglected. What is the ***maximum*** density of the smaller sphere that will allow the toy to float with that sphere poinitng up?

Chemistry

Problem 1.

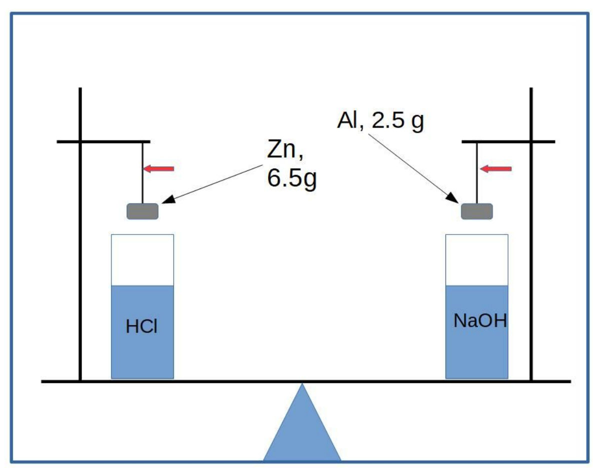


During the closing ceremony of the international Chemistry Olympiad, the organizers decided to announce the winning team in an unusual manner. They put on the table three glass beakers next to each other, each of which was half filled with some colourless and transparent liquid. The liquid in the right beaker was the aqueous solution of the sodium salt of dimethylglyoxime, the beaker in the middle contained the aqueous solution of barium chloride, and the beaker on the left had 5% ammonia. Then they added: to the right beaker - the nickel chloride solution, to the middle beaker - the sodium sulfate solution, to the left beaker - the copper sulfate solution. The liquid in each beaker have changed their appearance, and the three beakers represented the flag of the winning team's country.

Which country won the Olympiad?

Problem 2.

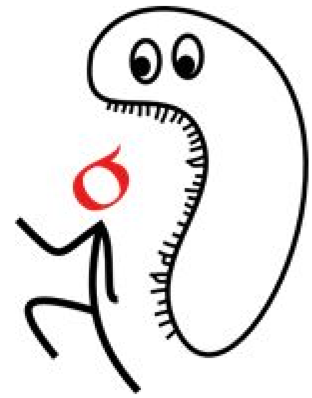
A balance is placed into a large gas-tight container filled with air (see the figure below). Two identical beakers are placed on each arm. The left beaker contains 200 mL of 1 M HCl, the right beaker contains 200 mL of 1 M sodium hydroxide. A piece of zinc (gray) with the mass of 6.5 g is suspended above the left beaker, and a piece of aluminium with the mass of 2.5 g is suspended above the left beaker. The scales are at equilibrium.



A robotic arm cuts the strings (the positions of the cut are shown with the red arrowa), and zinc and aluminium fall into the HCl and NaOH solutions, accordingly. Assuming that evaporation of water from the beakers is neglidgible, tell if the balance will tip after the system will come to a new equilibrium state, and if yes, in which direction.

Biology

Problem 1.



Five pots are filled with a white quartz sand. The same plants (for example, *Arabidopsis thaliana*) are planted in each of them. During next six months, the pots will be being watered with the following liquids:

- a pot #1: distilled water;
- a pot #2: dilute solution of the 1:1 mixture (by weight) of diammonium phosphate ($(NH_4)_2HPO_4$) and ammonium phosphate ($NH_4H_2PO_4$);
- a pot #3: dilute solution of potassium ammonium hydrogen phosphate ($K(NH_4)HPO_4$);
- a pot #4: dilute solution of the 1:1 mixture (by weight) of ammonium nitrate (NH_4NO_3) and potassium chloride (KCl);
- a pot #5: dilute solution of the 1:1:1 mixture (by weight) of sodium nitrate ($NaNO_3$), ammonium chloride (NH_4Cl) and potassium sulfate (K_2SO_4).

“Dilute solution” means that the concentration is in between 0.01-0.5% in each case.

In which pot the plant will grow better? What will happen in other pots? Describe the anticipated changes in as many details as possible.

Problem 2.

Many bacteria have special restriction enzymes to protect them against viruses. They cleave DNA at specific sequence motifs (usually, 6-8 nucleotide long) that are either absent in the bacterial DNA or protected from cleavage in some special way. These restristion enzymes are named after the first letters of the Latin name of the bacterial genus and species they came from; for example, *EcoRI* is a restriction enzyme (type I) from *Escherichia coli*. When such an enzyme acts on DNA, breaks occur in strictly defined places (for example, *EcoRI* cuts DNA only after the G nucleotide in the GAATTC context), and, as a result, DNA fragments of a certain length are produced. By comparing the length of cleavage products obtained after treatment of DNA by different restriction enzymes (and/or their combinations), one can determine the relative location of the cleavage sites and create a restriction "map" of this DNA (Figure 1 shows an example of such a map; *Alu, Xba, Not* are the names of the restriction enzymes, and the numbers denote the distance between cleavage sites, in thousands of base pairs).

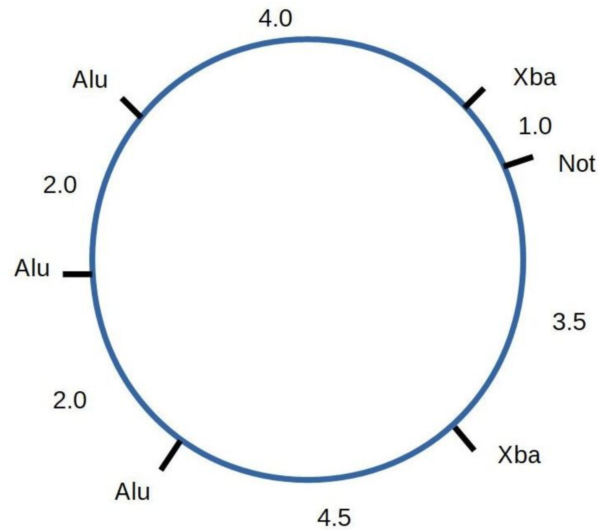


Fig. 1. An example of a restriction map of some circular plasmid that was subjected to a treatment by *Alu*, *Xba*, and *Not* restriction enzymes.

Professor Cutting isolated a small circular DNA (such circular DNA is called a "plasmid") from some bacteria. This plasmid (plasmid A) contained the gene that provided the bacteria with resistance against penicillin. Another plasmid (plasmid B) was isolated from a related bacteria; this second plasmid provided no penicillin resistance, although its length and nucleotide composition were exactly the same as those of the first one. To figure out the reason of such a difference, professor Cutting subjected both plasmids to restriction analysis using three restriction enzymes (*Hind*, *Sal*, and *Ava*), and the distribution of fragments is shown in the Table 1. Based on these data, Professor Cutting concluded that the plasmid B had a genetic defect: one of its segment had the opposite orientation as compared to that in the plasmid A. **Determine the approximate position of this segment relative to the restriction sites**. To do this, build restriction maps of both plasmids (as in the example shown in Fig. 1).

Table 1. The lengths of fragments obtained after cleavage of plasmids A and B with restriction enzymes. The length of fragments is shown in thousands of nucleotides (a.k.a. kilobases, kb)

Restriction enzyme	Plasmod A	Plasmid B
Sal	5 and 5	5 and 5
Hind	4 and 6	2.5 and 7.5
Ava	4.5 and 5.5	4 and 6
Sal + Hind	1, 2, 3, and 4	1, 1.5, 2.5, and 4
Sal + Ava	1.5, 2, 3, and 3.5	2, 2, 3, and 3
Hind + Ava	1.5, 2, 2.5, and 4	1.5, 2, 2.5, and 4

Computer Science

- You can write and compile your code here: http://www.tutorialspoint.com/codingground.htm
- Your program should be written in Java or Python
- No GUI should be used in your program: eg., easy gui in Python. All problems in POM require only text input and output. GUI usage complicates solution validation, for which we are also using *codingground* site. Solutions with GUI will have points deducted or won't receive any points at all.
- Please make sure that the code compiles and runs on http://www.tutorialspoint.com/codingground.htm before submitting it.
- Any input data specified in the problem should be supplied as user input, not hard-coded into the text of the program.
- Submit the problem in a plain text file, such as .txt, .dat, etc. **No .pdf, .doc, .docx, etc!**

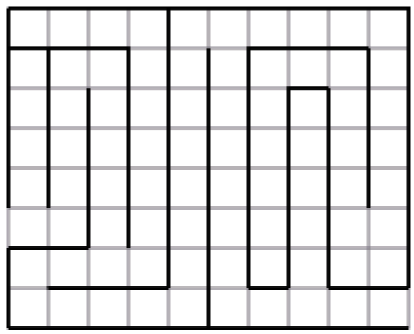
This year, your task will be to find an exit from a maze!
The mazes we will be working with will be rectangles of M*N square cells; some cells have walls around them. All this information is recorded in a file **maze.txt**; detailed description of the file format is below. For each problem, your program needs to read this file (which will be placed in the same folder as your program file) and then answer questions about the maze.

Format of maze.txt file:

First line of the file contains two numbers, M (number of rows) and N (number of columns), separated by a comma. You may assume both M,N are less or equal than 20.
After this, you have M lines, each consisting of N numbers, separated by commas. Each number describes one cell of the maze as follows:

- If there are no walls around this cell, the number is zero
- A wall at the TOP of the cell is represented by 1
- A wall on the RIGHT of the cell is represented by 2
- A wall at the BOTTOM of the cell is represented by 4
- A wall on the LEFT of the cell is represented by 8
- Two or more walls are represented by the sum of the corresponding numbers. E.g., if there are walls at the top, left, and right, the cell is described by number 1+2+8=11

For example, the maze shown below:



will be described by the following file:
8,10
13,5,5,3,9,1,5,5,5,3
11,9,3,10,10,10,9,5,3,10
10,10,10,10,10,10,10,11,10,10
10,10,10,10,10,10,10,10,10,10
10,10,10,10,10,10,10,10,10,10
4,6,10,10,10,10,10,10,8,2
9,5,4,6,10,10,14,10,12,6
12,5,5,5,6,12,5,4,5,5

We will also refer to cells of the maze by their coordinates:
x=column number (0..N-1), counting from left
y=row number (0..M-1), counting from the top
so that top left cell has coordinates (0,0), top right cell has coordinates (0,N-1), etc.

Problem 1 (5 pts)

Your program needs to read file maze.txt in the format described above and another file, path.txt, which contains a description of a path in the maze as specified below. You need to analyze the path and print the following:

- if the path is trying to go through a wall of the maze, you need to print "INVALID PATH: HIT THE WALL IN STEP ", followed by the step number. You can stop analyzing the path after that.
- if the path exits the maze after some number of steps without hitting any walls, you need to print "EXITED THE MAZE AFTER X STEPS" (replacing X by the actual number of steps). You can stop analyzing the path after that.
- otherwise, you need to print "PATH IS VALID; FINAL POSITION IS ", followed by X and Y-coordinates of the final position of the path.

Format of path.txt file:
first line: X and Y-coordinates of the starting position, separated by a comma. You can assume that the starting position is inside the maze.
second line: a sequence of instructions, separated by commas. Each instruction is one of the four words (all caps): UP, DOWN, LEFT, RIGHT
For example:
3,4
UP,UP,LEFT,DOWN,DOWN,LEFT,DOWN,RIGHT

Problem 2 (10 pts)

Your program needs to read file maze.txt, using the format described above, and then prompt the user to enter x- and y-coordinates of a cell in the maze. After reading these coordinates, your program should find and print a path from this starting position to an exit from the maze, as a series of instructions (UP, DOWN, LEFT, RIGHT) so that after executing these instructions, you would be immediately outside the exit from the maze. If such a path doesn't exist, your program should print "NO EXIT!". You are not required to find the shortest path.

To get full credit, your program should be able to find the exit from any maze and any starting position if such a path exists. Solutions utilizing simpler algorithms which are not guaranteed to find the exit (such as wall-following algorithm) will receive a partial credit.

Suggested reading: https://en.wikipedia.org/wiki/Maze_solving_algorithm

---

The application deadline is April 18, 2019. We will notify applicants regarding acceptance decision no later than **May 1st**.

Good luck with your application!