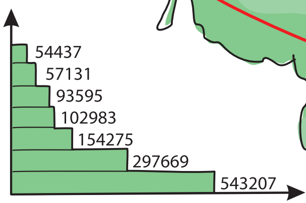
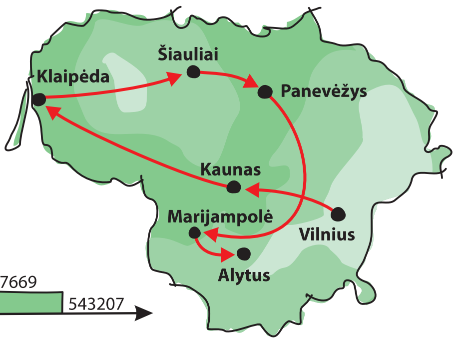
CITIES

In Lithuanian map, major cities are linked starting from the largest, Vilnius (inhabitants 543 207), descending (see map). The diagram shows number of inhabitants in the cities. Howeverm the names of cities are missing.





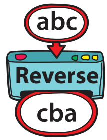
What is the population of Panevėžys?

Data and information processing is an important part of the computer science. Information can be presented in different ways. It is important to notice patterns and discover relationships between different data sets. In this case, we use the same data on a map and on a bar chart. It is important to notice the connection between the cities indicated by arrows on the map and sorted numbers on the chart.

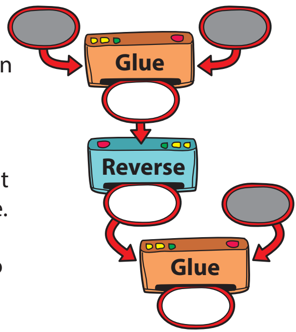
TEXT MACHINE

We have two kinds of text machines.

The Glue machine takes two pieces of text and glues them together



The Reverse machine takes one piece of text and reverses the order of the letters



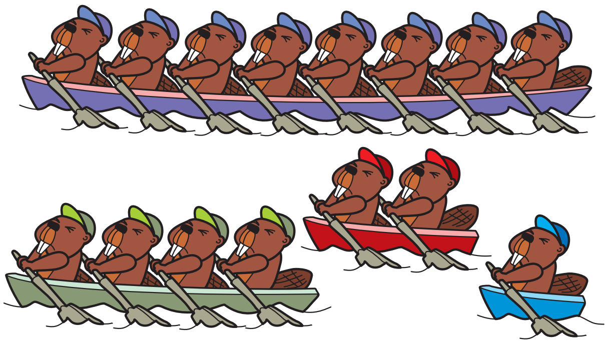
Combining two Glue machines and one Reverse machine in the way shown on the right picture, we obtain complex text machine. Our new complex machine needs three texts to work on (grey ellipses), processes them, and gives one text as the result of its work in the bottommost ellipse. **Which texts must be given to the machine, if we want the machine to produce the result QUESTION?**

1. EUQ TS NOI
2. TSE UQ INO
3. I TSEUQ ON
4. QU EST ION

The task is connected to formal languages and automata - one of the core parts of informatics. A language can be seen as a system suitable for expression of certain ideas, facts and concepts. For formalizing the notion of a language, one must cover all the varieties of languages such as natural (human) languages and programming languages. Automata theory is closely related to formal language theory. An automaton is a finite representation of a formal language that may be an infinite set.

ROWING TOURNAMENT

Some beavers would like to attend a rowing tournament. They have four boats available, one for eight beavers, one for four beavers, one for two beavers and one for a single beaver. However, the rules of the tournament state that every beaver may only participate in one contest.

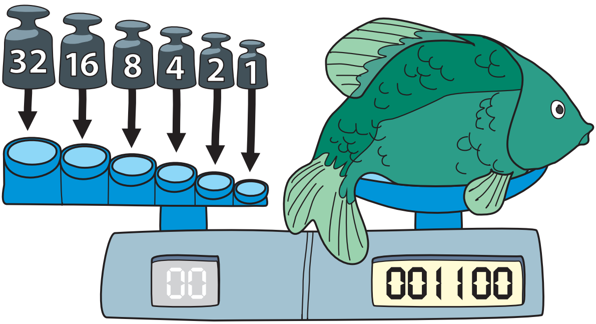


The trainer of the beavers is asked to write down for each type of boat whether they are going to participate (1) or not (0), starting from the largest boat. For instance, if ten students participate, he would have to write down 1010. This time thirteen beavers are going to participate.

What does the trainer have to write down?

The binary system is a numeral system like the standard decimal system. The difference is that instead of allowing ten different digits (0 to 9), only the digits 0 and 1 are used. The weight of each figure at position n is not 10^n, but 2^n. To convert this number to the decimal system, you have to multiply each digit by its position weight, so $1101=1*2^3+1*2^2+0*2^1+1*2^0=8+4+0+1=11$.

A Beaver scale shows weight both in decimal (left) and binary (right) numbers.
A fish weights 1100 kg in binary number system.



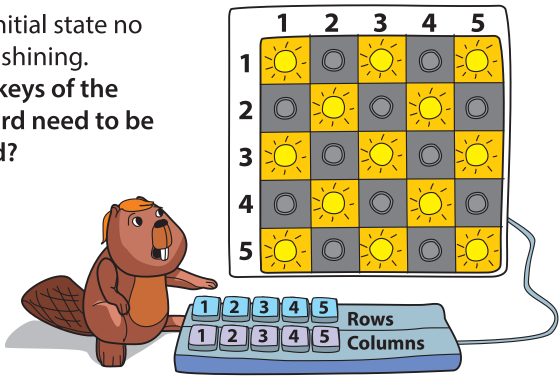
Which weights do you need to put on the scale plates so that you can see the weight of the fish in decimal numbers?

On your computer, everything is coded using binary digits: zero and one. How can we express a binary number by the, for us, usual decimal form? One of the simplest ways - imagine a scale with multiple in two weights (1, 2, 4, 8, 16, 32, and so on). Start from the end of the binary number and take only those weights, which are equal one (weights corresponding to zeros are ignored - leave empty places).

GRID OF LAMPS

A beaver controls a grid of lamps by using a special keyboard. The “row” buttons of the keyboard control the rows of lamps and the “column” buttons of the keyboard control the columns of lamps. By clicking any button all the lamps in respective column or row change their states: if a lamp is shining – it is turned off, if a lamp is not shining – it is turned on.
Beaver would like to get such pattern of shining bulbs:

In the initial state no lamp is shining.
Which keys of the keyboard need to be pressed?



Binary logic operates with bits - the smallest unit of information capable of acquiring two values. Electronic circuit operation is mainly based on binary logic or Boolean algebra (introduced by George Boole). Shining a lamp is one of simplest examples of binary logic, two states are easily noticed: the light is on (shining) - the light is off (not shining). Also a logical inversion operation can be introduced, the current status shall be reversed. What’s more - inversion applies to a whole single row (or column) - all lamps. Boolean algebra has been fundamental in the development of digital electronics, and is provided for in all modern programming languages.

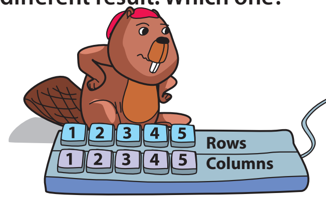
PROGRAMMING LAMPS

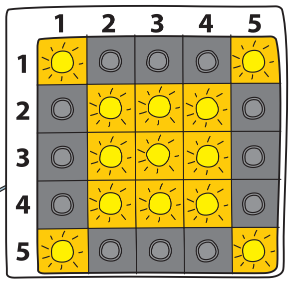
A beaver controls a grid of lamps by using a special keyboard. By one command he can change states of all lamps in respective column or row: if a lamp is shining – it is turned off, if a lamp is not shining – it is turned on.
Beaver wrote 4 programs:

1. 1C 5C 2R 3R 4R
2. 1R 5R 2C 3R 4R
3. 1R 5R 3R 4R 5R 1C 5C
4. 1C 2C 3C 4C 5C 1C 5C 1R 5R

C represents a column and R a row. For example, a command 1C turns on all lamps of the first column. If we repeat the same command (1C), all lamps of the first column will be turned off.

Three of the above programs give the result as shown in the picture.
One of the programs presents a different result. Which one?





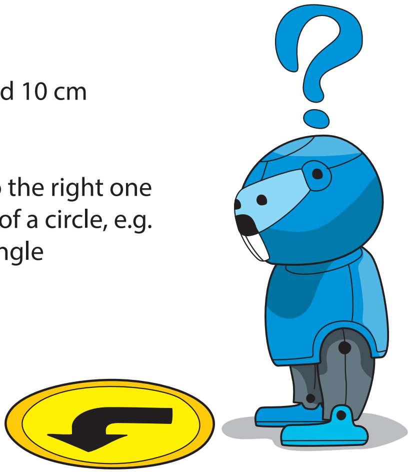
The binary number system plays a central role in how information of all kinds is stored on computers. Understanding binary logic can lift a lot of the mystery from computers, because at a fundamental level they're really just machines for flipping binary digits on and off. Binary representation is very much used in informatics, since all data are stored into bits, and each bit can take either 0 or 1 as value. Construction of lamp is based on binary logic, or Boolean algebra, when two states are used: light is on (shining) - light is off (not shining).

TO RIGHT

A little beaver has constructed a robot which can execute only two programs:

Forward 10 cm

Turn to the right one fourth of a circle, e.g. right angle

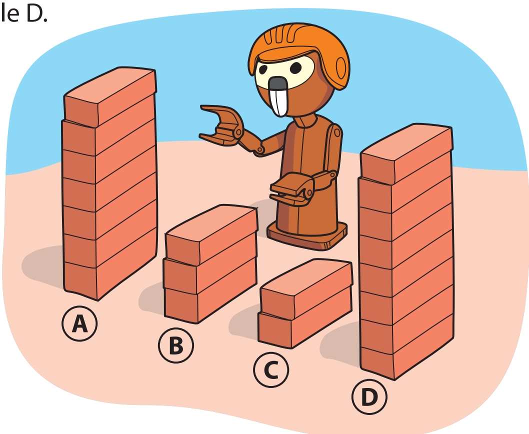


Is it possible to turn a robot a right angle to the left using these two commands? If yes, write down a set of commands.

While writing a program, the programmer has to think about actions, commands, and their status. Programmable computer system actions may be limited by technical possibilities. For example, a beaver robot cannot perform a turn to the left using given commands. Many programmable devices have limited possibilities of actions and states. Computer scientists are interested how to work with restrictions and to perform actions. Similar to programming languages - obviously not every language can have every feature.

BRICK TOWERS

There are four piles of bricks. A robot can move bricks from one pile to another one. The only command for the robot is a triplet of numbers in parentheses. For example, (A, 3, D) means: take 3 bricks from pile A and place them on pile D.



Write down the commands needed for the robot to change the piles shown in the picture so that after executing the commands, the number of bricks on every pile is the same?

Understanding instructions with parameters for robot and to program the robot using a simple algorithm. Algorithms and programs are important concepts. A command is an accurate and unambiguous instruction that can control computers, robots, or automatons computer, robot or automaton. The command is expressed by word abbreviations or letters. When we create an algorithm that should be understandable by a computer, all steps need to be very precise and detailed. Everything needs to be explicitly mentioned so that there is no source for confusion or interpretation – you cannot assume that the computer has some "background" knowledge as we humans have.

TAKING PHOTOS OF KANGAROOS

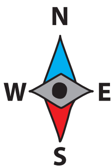
A beaver is flying a drone above a hall where kangaroos and beavers are sitting.

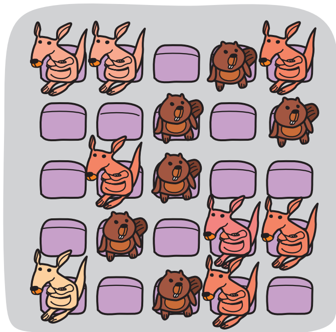
The drone is controlled with a command **FLY *direction step.***

– The parameter direction is defined as a letter, standing for either north (**N**), east (**E**), south (**S**), or west (**W**).

– The parameter step corresponds to a distance: there is one step between adjacent seats or adjacent rows.The drone starts flying from the center of the hall and is executing these commands:

FLY W 1
FLY N 2
FLY E 3
FLY S 3
FLY W 1
FLY S 1
FLY W 3





After the drone has executed a command, it takes a photo of exactly one animal below.

One of the kangaroos was left without a photo - you still need one more "command to photograph it". Write it down.

A command is a highly important concept in programming. Programmers must understand what is exactly done by a specific command. To make commands usable in various situations they are given parameters - quantities with variable values. In this case the command has two parameters. The parameters are given in specific order: the first parameter defines the direction (its value - one out of four letters), the second parameter defines the distance (a natural number).

A DRONE ABOVE THE HALL

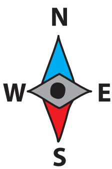
A beaver is flying a drone above a hall where kangaroos and beavers are sitting. The drone is controlled with a command **FLY** ***direction step.***

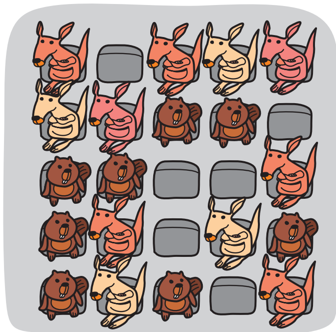
– The parameter direction is defined as letter, standing for either north (**N**), east, (**E**), south (**S**), west (**W**).

– The parameter step corresponds to a distance: there is one step between adjacent seats or adjacent rows.

The drone starts flying from the center of the hall and is executing these commands:

FLY W 2
FLY N 2
FLY E 3
FLY S 3
FLY E 1
FLY S 1
FLY W 4
FLY N 3





After the drone has executed a command, it takes a photo of exactly one animal below.

How many beavers and how many kangaroos have been photographed by the drone?

As you can see, the drone is controlled by one command, ordering it to fly. The direction and distance of flight are described by the parameters of this command. The parameters make the command universal, fit for many different uses. By changing the values of parameters, it seems, we are ending up with different commands. Note that not only the values of parameters are important, but also the order in which they are given. The commands are executed in the same order as they are written down.

A CITY STATUE

A little kangaroo is staying at a hotel in Beaver Town. She follows the directions below given by the hotel staff to get to the famous Beaver statue to take some pictures.

1. From the hotel's door, immediately turn to the left.

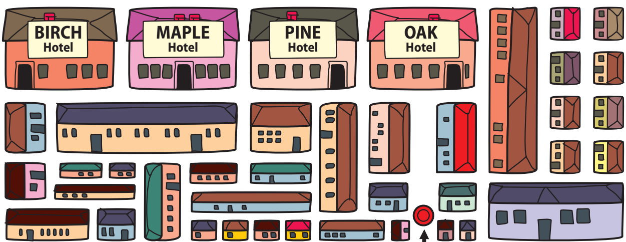
2. Go straight forward through two intersections.

3. At the third intersection, turn right.

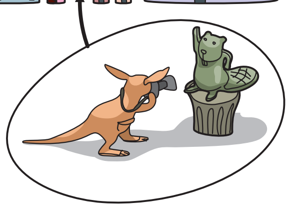
4. Go straight forward. At the first intersection, turn left.

5. Go straight forward. At the first intersection, turn right.

A little kangaroo found the statue and is taking a picture.



In which hotel is the kangaroo staying?

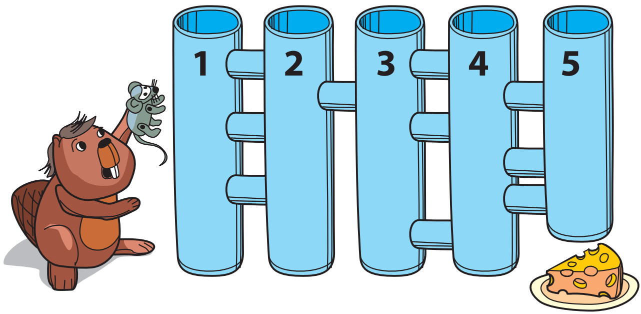


In this task a little kangaroo is given a set of instructions in order to reach a given goal. This is the basic idea of algorithms, which make up an essential part of informatics. An algorithm is a step-by-step solution to a problem. The steps involved should be as precise as possible and there needs to be a way to reach the solution. Algorithms can then be translated into computer programs by implementing them in a given programming language. As the task shows, we also deal with algorithms in our everyday life as well.

A TUBE SYSTEM

A beaver constructed a robo-mouse. The robo-mouse can move in the grid of smaller and larger tubes searching a delicious piece of cheese following these commands:

1. Go downwards until you see a smaller tube crossing the way.
2. Turn immediately to the smaller tube and continue to the neighboring larger tube. Then repeat the command 1 until you can get out.



In which tube should the mouse start so that it reaches the cheese at the end of tube 5?

Many automata are programmed in such a way that they have to follow exact commands. Also the mouse in this task follows exactly the commands "go downwards" and then "change at the next crossing" again and again. This kind of commands are deterministic in the sense that it depends on the choice of the tube entrance which way the mouse runs in the tube system. Most of the computer programs are deterministic. It means that every time the program starts with the same input data it performs exactly the same calculations and delivers always the same output.

THE CLINGING ROBOT

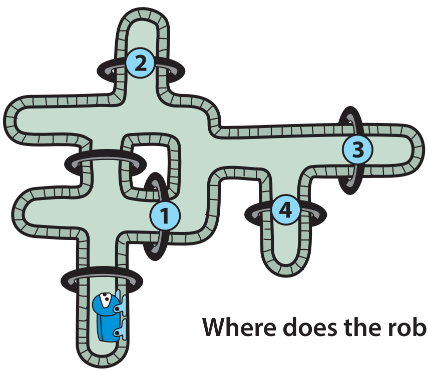
The clinging robot walks along the road clinging at one of the sides. The clinging robot knows four commands:

START: Start walking along the side you are clinging to.
STAY: Keep on walking at the side you are clinging to.
SWITCH: switch to the other side and keep on walking.
STOP: Stop walking.

The **START** command is executed at any place, wherever the robot happens to be. Each of the other commands is executed exactly when the robot walks over one of the dark magnetic control devices on the road.

The clinging robot executes this program:

START SWITCH STAY STAY STAY STOP

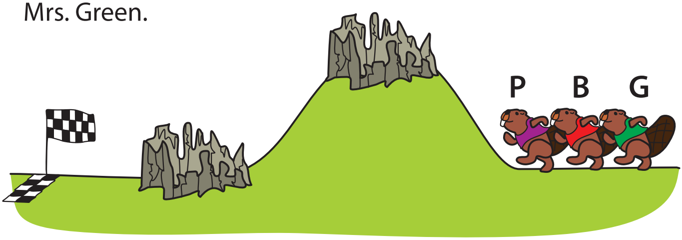


Where does the robot stop?

Automatically moving vehicles are found at many places including tunnel systems, airports and factories. These machines are controlled by computer programs. Basically a computer program is a sequence of commands. The commands are related to physical sensors and means of maneuvering available to the vehicle.

CROSS COUNTRY

Three fierce runners will compete in a cross-country run. The terrain is as shown in the picture: uphill, followed by some rocks, downhill and then again some rocks. Mrs. Pink starts in the first position, followed by Mr. Brown and Mrs. Green.

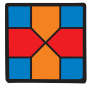


Mr. Brown will overtake one beaver when running uphill.	**B**	
Mrs. Pink will overtake one beaver when running downhill.	**P**	
Mrs. Green will overtake one beaver when running across rocks.	**G**	

In which order will they finish the race?

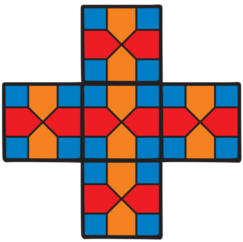
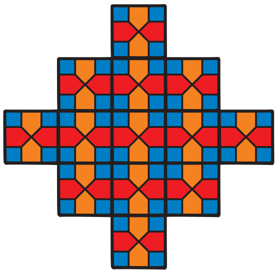
Programmers must often look up closely at how their programs execute. This is especially true when the programs do not work well: in this case, programmers carefully go through and check the effect of each line of the program. This task is similar. You are given some data – the sequence of running beavers. You have to "step through the program" - where the "steps" are uphill – rocks – downhill – rocks. You have to observe the effect of each step on the sequence and thus discover the "output" of the program, that is, the order at the end.

STAINED GLASS



A robot is decorating the windows with pieces of glass. The pieces can be of three different colors: blue, red, or orange.

Eight pieces of glass form the basic pattern. Using several basic patterns, the robot can create a nice regular symmetric decoration.

A three column ornament consists of 5 fragments.	A five column ornament looks like this:
	

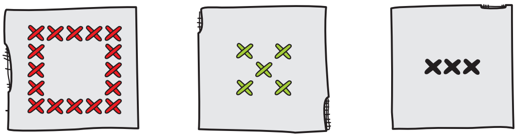
How many square pieces of blue glass will be needed to complete a stained glass decoration with seven columns?

The robot operates under a program. Here the functionality of the program is described by illustrating it with several examples. To write a program and turn this description into an algorithm of making stained glass ornaments, you need to understand the functional description of this program. This may be done by recognizing recurrent elements - in informatics they are called subroutines (or procedures, or functions). After breaking down the problem like that, it is not hard to calculate the answer.

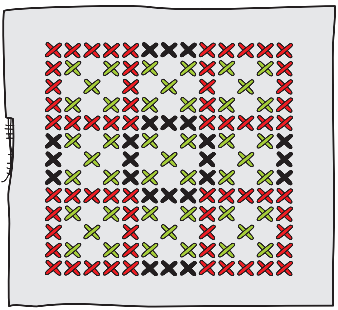
CROSS STITCH EMBROIDERY

Beaver has a cross stitch embroidery machine. One step results with one finished cross stitch.

Kangaroo has a newer model machine: with one step it can complete one of the following three figures:



One cross stitch cannot be made on top of another, already existing cross stitch. Beaver has made an ornament:



To produce the ornament, the beaver's machine made 133 steps. Kangaroo laughs: it's machine would take far less steps to make such an ornament.

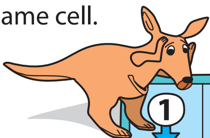
How many steps would the kangaroo's machine make?

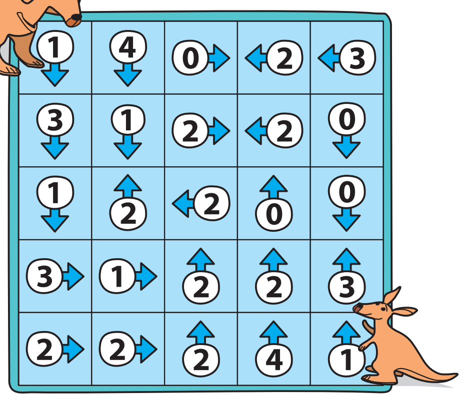
By recognizing recurrent patterns and assigning a form to them, the completion of further tasks can be facilitated. These recurrent patterns are called subroutines in programming. A subroutine constituting of several commands can be used as one complicated command and thus saves the time needed. This is being done by the kangaroo's machine - it stitches patterns, rather than single crosses. In programming, a subroutine is a sequence of program instructions that perform a specific task, packaged as a unit. Subroutines may be defined within programs, or separately in libraries that can be used by multiple programs. In different programming languages, a subroutine may be called a procedure, a function, a routine, a method or a subprogram.

KANGAROO AND BABY

A kangaroo wants to meet her baby.

- The first to jump is the mother kangaroo, then the baby kangaroo, and so on.
- When jumping from cell to cell, the kangaroos follow the arrows on the field and move a distance written by a number in the cell.
- The kangaroos do not care about the arrows and numbers in the cells they jump over.
- The mother and the baby meet when they are both in the same cell.





Will the kangaroo meet her baby?

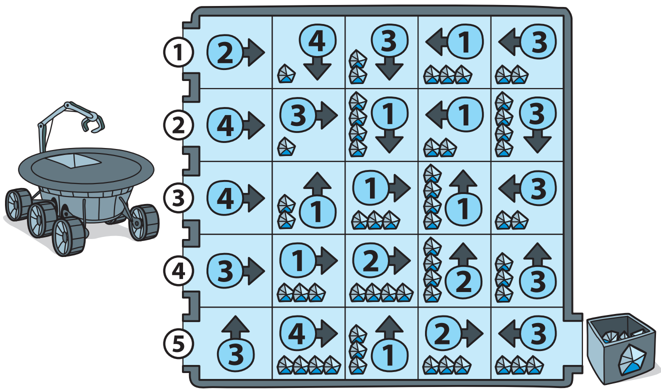
The field where the kangaroos hop can be seen as a simple program: there are commands telling what to do, a designated start and end of actions. The arrows and the numbers in the cells unambiguously describe the movement of kangaroos. Those are commands, despite being shown as arrows rather than written down as words. The program can be carried out automatically - the conditions are set; the actions are described unambiguously.

A LUNAR ROVER

B

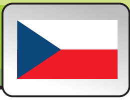
A beaver has built a lunar rover to move in a maze on the surface of the Moon and collect minerals.

1. The rover enters the maze through one of the entrances on the left side.
2. It follows the directions of arrows and moves a distance written by the number in each cell.
3. The rover does not care about the arrows and numbers in the cells it passes.



Which entrance or entrances will lead the lunar rover to collect the most of the minerals?

This lunar maze can be seen as a simple program: there are commands telling what to do, with a designated start and end of actions. The arrows and the numbers in the cells are genuine and unambiguous commands, despite being shown as arrows rather than written down as words. The number by each arrow can be seen as the parameter for the command: it indicates the distance of the next move. Another command could be one ordering to gather the minerals from the cell. The moon rover's program can run automatically - the commands are unambiguous.



A ROBOT-SQUIRREL

A robot-squirrel moves across a square grid according to given commands:

– she goes forward to the next square;

– she puts a nut on the square;

– she repeats the first command: „goes forward to the next square" four times.

There can be a squirrel and many nuts on a square.
Which set of commands tells the squirrel to put four nuts in a row next to each other?

Ⓐ 

Ⓑ 

Ⓒ 

Ⓓ 

Programming of a robot is a typical informatics task. To create a correctly behaving program, it is necessary to understand how to put commands together. A robot program is the set of commands or instructions that tells a robot what tasks to perform.

A ROBOT AND DIAMONDS

A magician robot moves across a square grid according to given commands:

– he goes forward to the next square;

– he puts a diamond on the square;

– he repeats the command: „goes forward to the next square" four times.

If more than one command should be repeated, then brackets must be used like this:

– the robot repeats four times: [move one step forward; move one step forward]. As a result, the robot moves 8 steps forward.

There can be a robot and many diamonds on a square. **Which set of commands tells the robot to put four diamonds in a row next to each other?**

Ⓐ 

Ⓑ 

Ⓒ 

Ⓓ 

Execution in a computer or other devices is the process by which a machine performs the instructions of a program. The instructions in the program trigger sequences of simple actions on the executing machine. Almost all the programming languages provide a concept called loop (repetition), which helps in executing one or more statements up to a desired number of times. Execution of more than one sentence is usually enclosed in brackets. All programming languages provide various forms of loops, which can be used to execute statements repeatedly.

JAPANESE CALENDAR

The old Japanese calendar contained a cycle of 60 years enumerated from 1 to 60 and grouped in pairs, as shown in the figure. Pairs are colored following the sequence: green, red, yellow, white and black:

1	2	11	12	21		51	52
3	4	13	14			53	54
5	6	15	16			55	56
7	8	17	18			57	58
9	10	19	20			59	60

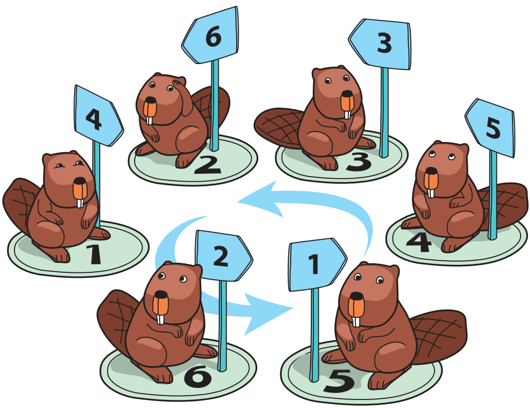
It is known that 1984 is year 1 in a new cycle of 60 years.
Which color is year 2017?

Digits of the years in the Japanese calendar are colored by certain rules, which can be called an algorithm. The algorithm can be constructed by analyzing the presented calendar and observing the regularities in year cycle repetition. It is important to understand that the key features of algorithms are sequence, decision and repetition.

ROUND DANCE

B

Six beavers play a game. Initially each beaver stays in one of the 6 different numbered rings (see the figure). At each ring there is a balloon with a number from 1 to 6 indicating a ring the beaver has to go to next (destination). There are different destinations for the different rings. After a signal, each beaver moves to the destination. This move is called round. Then the second round follows, then the third, and so on until all beavers happen to be on their initial places.



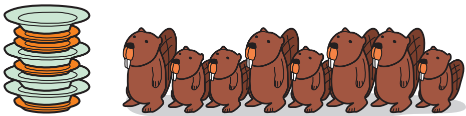
How many rounds will be needed to finish the game?

In Discrete mathematics, both the rule for exchanging the places of N different ordered (i.e. labeled with 1, 2, …, N) objects and the obtained arrangement are called permutation. Applying the same permutation (exchanging rule) on the arrangement obtained after the first exchanging give us a new arrangement, and so on. It is not difficult to prove that the number of permutations of N element is equal to 1 x 2 x ... x N, which is denoted by N! Solution by hand of the task uses an important characteristic of each permutation – its partition in cycles. The solution really builds the cycles of the given permutation: (1, 8, 14)(2, 10)(3, 12, 16, 5, 13, 4, 11)(6, 9, 15, 7), and measured the length of each cycle and finds the LCM (Least Common Multiple) of the cycle's lengths.

STACK OF PLATES

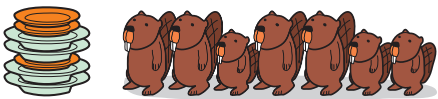
In the restaurant of the Beaver school, there are two different kinds of plates: the high ones for the small beavers, and the flat ones for the big beavers. Also the big and small beavers have separate queues to line-up, where one queue has only high plates stacked and the other only the flat ones.
One day, due to building activities, there is only room for one queue with a stack of mixed high and flat plates. The beaver kids are queuing for their lunch, and the kitchen beavers need to put the plates on the stack in the right order to make the stack match the queue.
For instance, have a look at this beaver queue:
Plates need to be put on the stack like this to match the above queue:
In one of the following pairs of plate stacks and beaver queues, there is a mismatch between queue and stack. In which one?

①

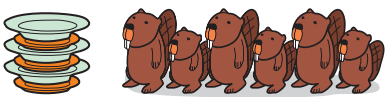


②

③



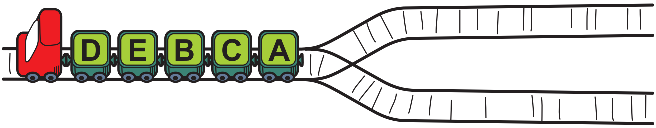
④



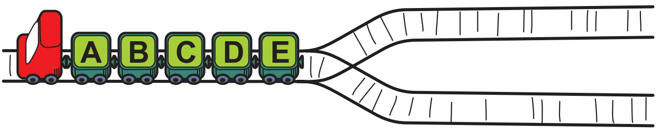
In computer programs, data needs to be well organized into so called data structures. Two of the most basic data structures are stacks and queues. A stack offers access only to the data that was added to the structure last (according to the LIFO principle: „last in, first out“). A queue offers access only to the data that was added to the structure first (the FIFO principle: „first in, first out“).

FREIGHT TRAIN

A freight train in the Bebras Railway was in the following configuration:



The locomotive can go forward or backward with any number of tail end freight cars can be connected to or detached from the train by one operation. The rail has the following layout. To reconfigure the train as in the following:

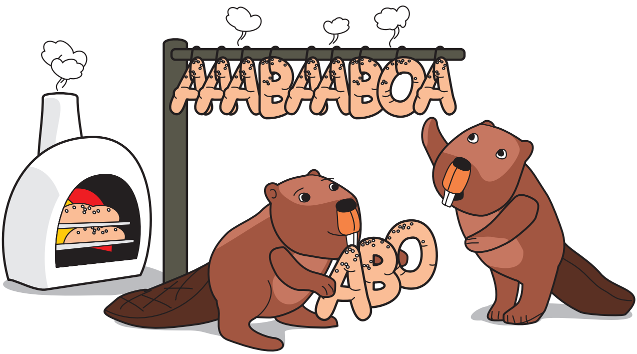


How many minimum operations (detach or connect) are necessary?

The data structure stack, which is a key concept for computer processing, is treated. In computer science, a stack is an abstract data type that serves as a collection of elements, with two principal operations: push, which adds an element to the collection, and pop, which removes the most recently added element that was not yet removed. The order in which elements come off a stack gives rise to its alternative name, LIFO (for last in, first out). Two or more data can be moved simultaneously in this question (different from only one data for the similar task in Tower of Hanoi).

BAGELS

Two friends have opened a bakery. Sue bakes three bagels (one of each shape A, B and O) and hangs them together on a stick, placing A on first, then B on second, and O third. She then repeats this process. Peter is selling the bagels and takes always the right-most bagel from the stick. Sue is baking faster than Peter can sell the bagels.

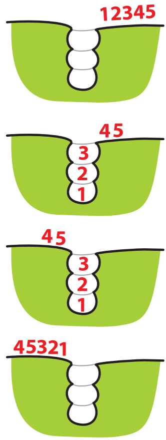


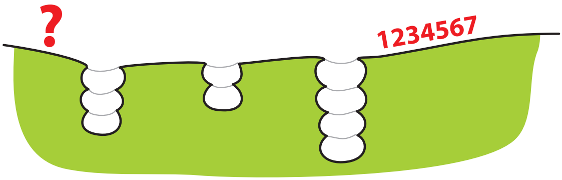
What is the fewest number of bagels sold by Peter if the bakery looks like the above picture?

The management of a data structure, namely a stack, is shown. When using a stack, elements can be stored only at the uppermost position and can be taken only from the uppermost position. That is, a stack is a LIFO data structure: last-in, first-out, meaning the most recently placed item into the stack will be the first item removed from the stack.

RABBIT HOLE

A line of beavers is going for a walk in the woods. They follow each other, one beaver after another. However, the nasty rabbits have dug lots of holes along the path that the beavers are walking on. Beavers fall in holes over top of each other, until the last beaver in the line crosses through and pulls up the beavers out of the hole, from the top to the bottom. An example of 5 beavers walking along through a hole holding 3 beavers is shown in the pictures.



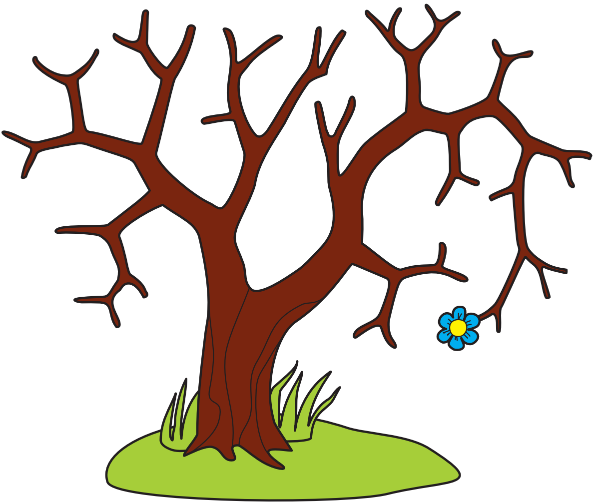


If there are 7 beavers, what is the order of the beavers after all beavers have passed over three holes in the last picture?

Organizing data in a structured way is important in informatics, and there are many different data structures that can be used for this purpose. This task shows an example of a structure called a stack, which works similarly to stacking plates on top of each other. You always add new plates on top of the stack and have to remove them from the top one at a time. This type of structure is commonly referred to as a LIFO-structure – the objects that have been added last are the first to be removed.

A BINARY TREE

In informatics one of the coding ways is binary tree: we start from the stem (S) and then go forward turning left (L) or right (R).

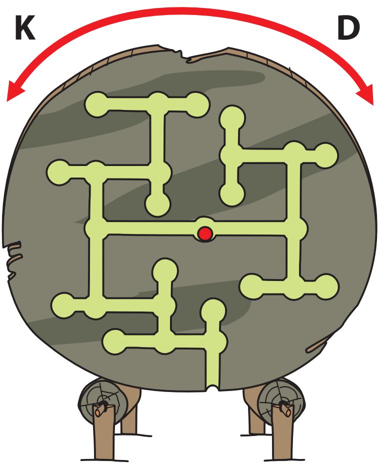


Write a sequence of the letters which lead us to the blossom.

Binary trees are commonly used to be able to access data very fast. One can specify a big number of branches with only relatively few letters. In fact, if one uses for example 10 letters one can describe the position of 1024 (2 to the power of 10) different branches. With 20 letters, one can describe over one million branches.

A SPINNING TOY

Beavers discovered a piece of wood into which worms made a system of tunnels. A handy father used it to make a toy. In the beginning we put a marble in the middle. The goal is to get the marble out by turning the spinning toy to the left (L) and right (R). By each turn the marble runs to the next pit or at the end out of the toy.

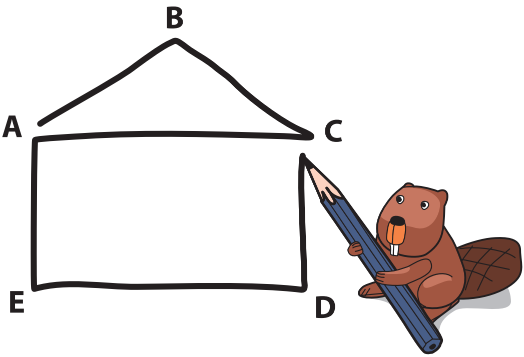


Help the marble to reach the exit: **write down the turning sequence for the toy.**

The problem is difficult to solve if one has to imagine turning the toy left and right (unless you can turn the monitor or print the image on a sheet of paper). From the toy we need to go to an abstract model. We can notice that each tunnel goes two ways: to the left and to the right. Thus, it is possible to apply a binary tree. Describing paths in binary trees is a common operation in computer science. It is, however, quite easy if we recognize that we are actually searching through a path in a tree.

DRAW A HOUSE

Draw a house by holding a pencil against paper sheet (without any lifting) and drawing the same line only once:



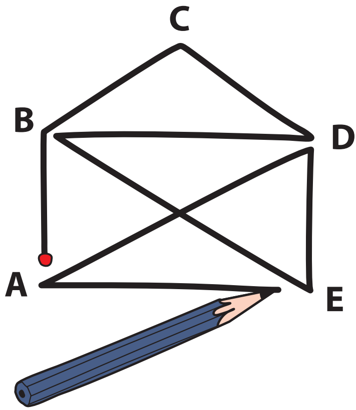
The picture shows a way starting from point A.
Can you start from any of the other points? From which one?

A graph is a structure consisting of vertices or nodes (points) which are connected by edges (lines). An odd node is a point where an odd number of lines meet. A graph containing exactly two odd nodes can be drawn without taking the pen off the paper and without repeating the lines, if one starts at one of the odd nodes and ends at the second odd node. This is, for example, the optimal route for visiting all famous places, to clean the streets and so on.

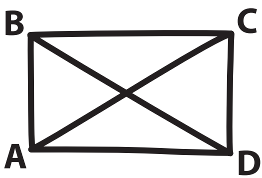
DRAW AN ENVELOP

B

An open envelope can be drawn without taking the pencil off the paper and drawing the same line only once



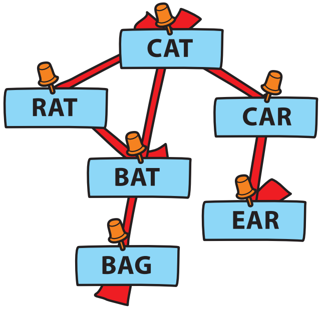
Now try drawing a closed envelope.

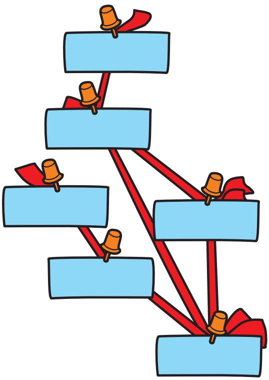


Why is the first case possible, but the second is not?

In informatics, a graph is a set of vertices or nodes (points) and edges (lines). A graph containing exactly two odd nodes (an odd-numbered node is one with an odd number of edges connected to it) can be drawn without taking the pen off the paper and without repeating the lines, if one starts at one of the odd nodes and ends at the second odd node. However, a graph with all nodes odd cannot be drawn in the described way. One has to count how many edges are connected to each node.

A beaver girl wrote words on cards. Then she connects any two words that differ in exactly one letter with rubber bands (right picture).

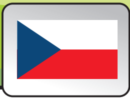




When she was done, her little brother started to play and erased the words. Also the cards were completely mixed up (left picture).
Little beaver realized that he had done something wrong but the girl calmed him down: "Don't worry, I know where to put the right words and you can help me do it".

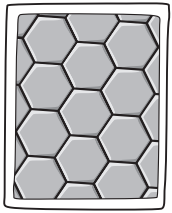
What are the correct words for the cards?

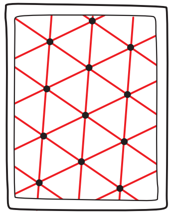
This is a problem about graphs: a graph is a set of objects, where some pairs of objects are connected. This task asks us to build a graph. We can proceed by counting the edges going from each node. There are two nodes with three edges, two nodes have two edges and two nodes have one edge. There is only one node with one edge connected to a node that has two edges. So we can continue with this method.



A PAVEMENT

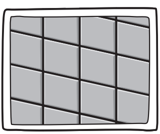
Peter took a photo of a pavement in front of his house and then created a graph which describes the paving (see pictures).



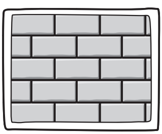


A point on the graph represents a tile. A line joining two points represents any two tiles bordering.
Later Peter was walking in the town and was photographing pavements. When he returned home he realized that all pavements (except of one) were suitable to fit his graph.
Can you recognize which of them was not?

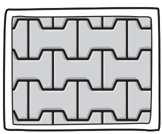
1



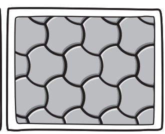
2



3



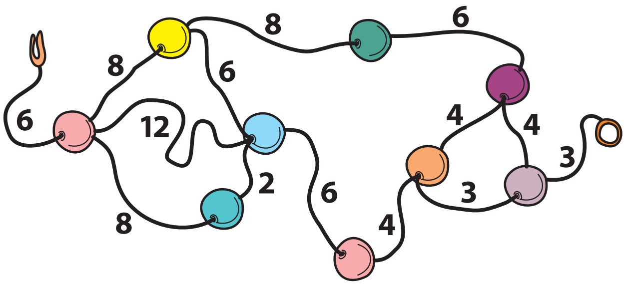
4



The solver has to discover the relationship between the graph and the shape of the tiles. The task concerns the comprehension of graphic information representation. This task demonstrates how graphs can be used to represent dependencies between items. A graph is a data structure that is used a lot in informatics to demonstrate relationships. Graphs also make it easier to visualize a task compared to just reading the descriptions of the relationships in text.

NECKLACE

A beaver lady made a necklace for herself. Now that it's finished, she's not sure that it will fit around her neck. The numbers tell the lengths of the threads between pearls. Clasps are on the left and right.



How long is the necklace?

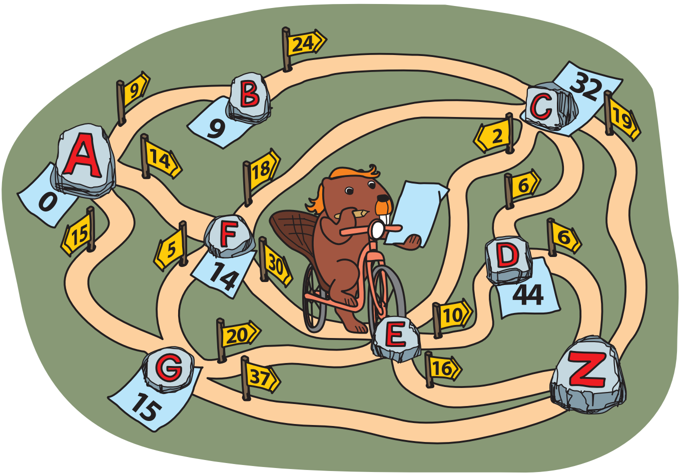
The task is about searching the shortest path to reach the end. The shortest path problem is one of the fundamental computing (informatics) task with everyday applications. The problem of finding the shortest path between two intersections on a road map (the graph's vertices correspond to intersections and the edges correspond to road segments, each weighted by the length of its road segment) may be modeled by a special case of the shortest path problem in graphs.

A BIKER

A beaver biker is choosing the shortest route from A to Z. There are only one-way cycle paths. She knows a clever approach (an algorithm) how to find the route and put hints on sheets of paper at crossings.

What is she writing at the moment?

Write an integer which the biker has counted for the current crossing E.



The shortest path problem is one of the fundamental computing (informatics) task with everyday applications. Shortest path algorithms are applied to automatically find directions between physical locations, such as driving directions on web mapping websites like MapQuest or Google Maps. Dijkstra's algorithm is one of the more popular algorithms used to find the shortest route. Dijkstra's algorithm will assign some initial distance values and will try to improve them step by step.

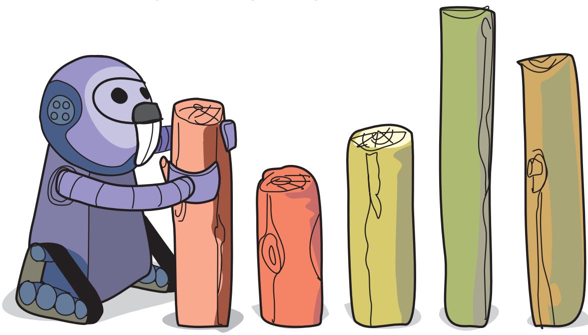
COMPARE AND SWAP

A beaver robot compares a row of poles and swaps poles one by one: the first one with the second one, then the second one with third one, etc. Reaching the end, the robot starts again from the left.

The beaver robot could execute two commands:

Command 1: Compare two adjacent poles.

Command 2: If the second pole is the shortest of this pair, swap the poles.

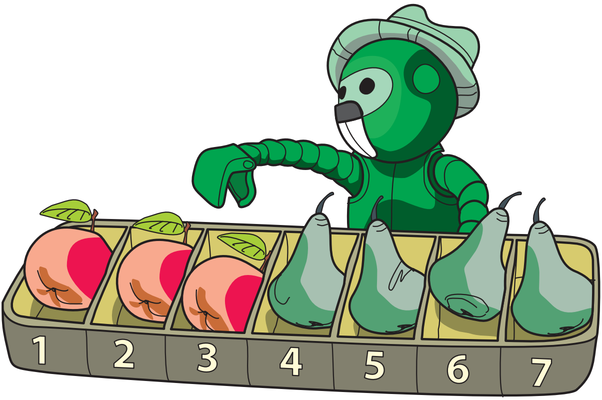


How many commands need to be run until he sorted all poles?

Sorting refers to arranging data in a particular format. Sorting algorithm specifies the way to arrange data in a particular order. The importance of sorting lies in the fact that data searching can be optimized to a very high level, if data is stored in a sorted manner. So a sorting algorithm is an algorithm that puts elements of a list in a certain order. Since the dawn of computing, the sorting problem has attracted a great deal of research, perhaps due to the complexity of solving it efficiently despite its simple, familiar statement.

SWAPING FRUITS

A beaver robot sorts apples and pears.
The robot can swap two adjacent fruits.



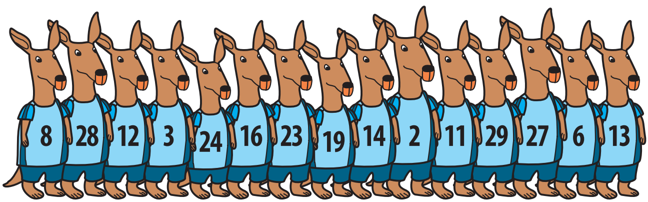
How many times does he need to swap at least before all the pears occur on the right side and all the apples on the left side?

This sorting algorithm has the name “bubble sort”. It is a simple sorting algorithm that repeatedly steps through the list to be sorted, compares each pair of adjacent items and swaps them if they are in the wrong order. The pass through the list is repeated until no swaps are needed, which indicates that the list is sorted. Although the algorithm is simple, it is too slow and practically very rarely used.

ORDERED JERSEY NUMBERS

Two teams of 15 players are shown below, with numbers printed on their jerseys. The players of the first team are ordered by jersey number.

The players of the second team are ordered by player height.

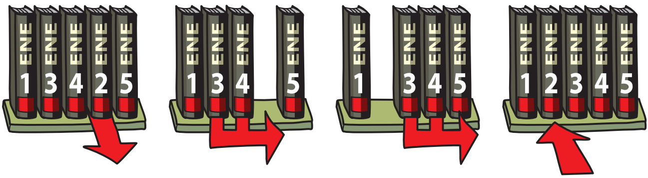


How many jersey numbers are in the first team that are also in the second team?

Searching an ordered list can be much faster than searching an unordered list. This task demonstrates that it is much quicker to search Team 1 for individual jersey numbers from Team 2, than it is the other way around. If we imagine that it takes one second to compare two numbers, then it will take approximately 45 seconds to solve this task by searching Team 1 for individual jersey numbers from Team 2, and it will take approximately 112 seconds to solve this task by searching Team 2 for individual jersey numbers from Team 1. Searching lists is a fundamental problem in computer science. Ordering lists is a fundamental tool in a computational thinkers' tool belt. It can be a good strategy to order an unordered list before searching, particularly if we have to search a list multiple times.

BOOKSHELF

There is an encyclopedia in the school library, but the volumes are not in the correct order. The librarian wants to arrange the volumes with as few movements as possible. He proceeds as follows (shown from left to right on the figures below): he takes a volume out of the shelf, shifts some of the remaining ones to left or right and puts the volume in his hand to the new free space.

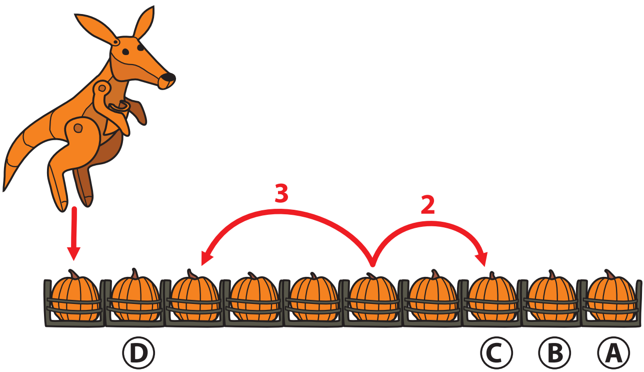


What is the smallest number of the take-shift-put sequences needed to order all volumes, starting from the situation shown below?

The task is about sorting, a fundamental sub-task in many more complex problems, and explores order statistics and sorting efficiency. Since the volumes not removed from the shelf stay in their original relative order, only those can remain in the shelf that already are in increasing order. The longest such sequence contains five volumes (1, 6, 7, 8, 9) and the remaining four volumes (4, 5, 3, 2) have to be removed. The order in which they are extracted from the shelf and inserted into their correct position among the remaining ones affects the number of volumes that have to be shifted, but not the number that have to be taken out from the shelf.

JUMPING KANGAROO

There are 10 boxes in a row with one melon in each. Beaver has built a jumping kangaroo robot. Each time when the kangaroo jumps, she can jump two boxes ahead or three boxes back. She can jump only into a box where there is a melon and then takes it.
Kangaroo starts on left side of the row as it shown:



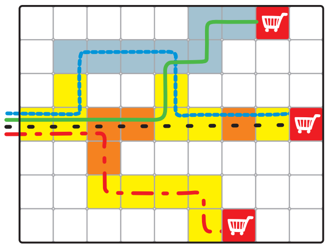
The kangaroo robot has finally collected all melons. Which one of the melons is the last taken?

One of the possible solutions to this problem - modelling the kangaroo jumps and the analysis of all possible sequences of boxes taken. We are looking for a sequence which matches the possible jumps. Another approach is to build a permutation one plate at a time. Once you figure out that a permutation is not valid (such as determining that the second plate cannot be anything but plate 3), you can remove the last plate(s) and continue building new permutations. This is called backtracking and if you can rule out many permutations early in your search, you can find a valid permutation much faster.

A HUNGRY BEAVER

Beaver felt hungry and wanted to go to a shop to buy some nuts. He decided to go with a bike to get there faster. Nearby there are three shops and a few roads to get to them. Beaver knows that there are easier and heavier road sections, as marked by different colored squares. The plan and explanations:

- Shop
- 1 minute per section
- 2 minutes per section
- 5 minutes per section



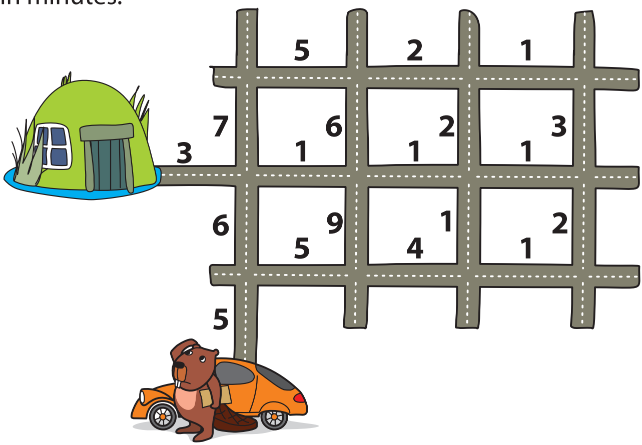
Which road should the little beaver take if he is in a hurry?

A. Blue road
B. Green road
C. Black road
D. Red road

One of the main tasks in computer science is the search for so called feasible solutions, i.e. for solutions that satisfy some given constraints. In our case, In our case, we look at fastest road sections. In some constraint problems, the fundamental question is whether there exists at least one feasible solution, i.e., whether all constraints could be satisfied at once. Also the task is related to so the called scheduling problems. Scheduling involves determining the correct or optimal order to perform a set of actions. Scheduling occurs in industrial applications, such as assembling a car from parts in a particular way, as well as in the management of machine instructions executed on CPUs.

NO TURNING LEFT!

Traffic jam in a city! It is impossible to turn left in such a traffic density. My father is hurrying up home by car from the hotel he works at. In the picture, the travel duration for each street in minutes is given.
Father uses his GPS navigation to find shortest way home in minutes.

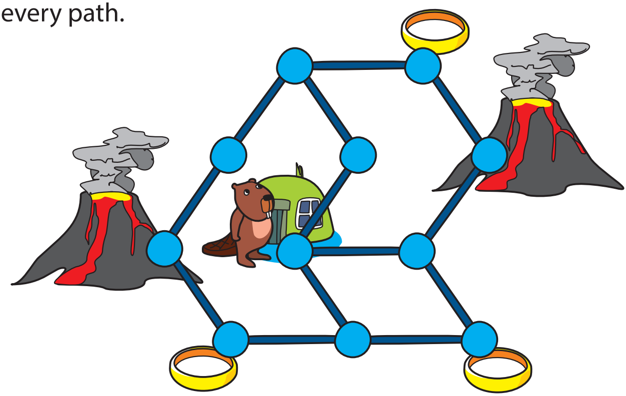


How long will it take at least to go from the hotel to his home when it is not possible to turn left?

In Informatics we often try to determine a path with minimum effort which is in compliance with given conditions. The decision which path to choose is based on certain criteria (e.g. time and float damage) in every decision point (for our beavers this is at the crossing). The number of all possible paths is usually too high for considering all of them to identify the path with minimum effort. Therefore, fast algorithms try to reduce the number of inspected paths.

BEAVER THE HOBBIT

Beaver goes on an adventure. He has to pick up three rings and then throw them in the lava of a volcano. After this he wants to return back home.
Beaver has a map. Every path on the map can be walked in exactly one day. Beaver is allowed to take the same path more than once, and he does not have to walk on every path.



What is the minimum number of days that Beaver has to walk to pick up the three rings, then throw them in one of the volcanoes and then return home?

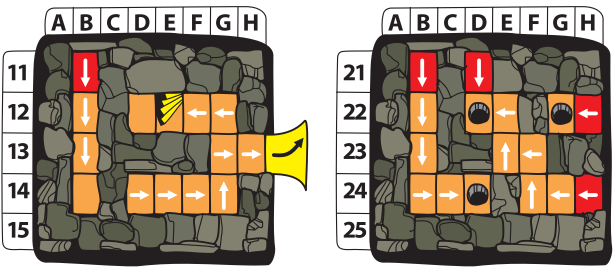
The goal of this problem is to find a shortest path that satisfies several constraints. The path must go through three particular nodes of the map, and it must then go through at least one of two particular nodes of the map. One way to be sure that there is no better solution consists in trying all possible paths, in a brute force way.
The brute force approach is always an effective way to solve a problem using a computer, even though it only gives results in a reasonable amount of time when the number of possibilities to enumerate is not too large (e.g., less than a billion). Otherwise, cleverer algorithms may need to be devised.

BEAVER MAZE

Hazard Beaver decides to better protect his home by creating a maze. He makes a model of the maze on two floors and creates a robot ant which can move the planes of the model as follows:

1. when on an arrow it goes ahead to the next cell
2. when on a circle it goes down to the lower floor on the corresponding cell
3. when on stairs it goes up to the upper floor on the corresponding cell
4. when on a empty cell it stops
5. dark cells make walls and cannot be crossed by the ant

In the first floor there is one entrance cell (B11) and one exit cell (H13). In the second floor there are only entrance cells (B21, D21, H22, H24).

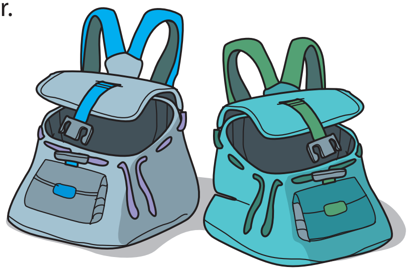


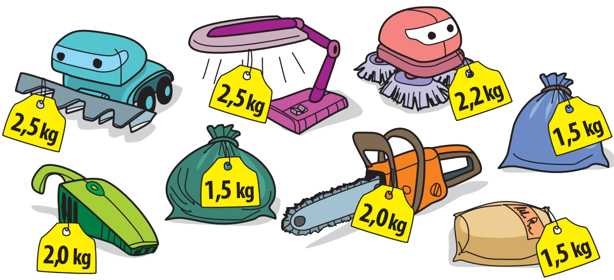
What entrance cell lets the ant robot exit from the maze?

One has to live with the constraints given to find a possible solution. In this case reaching certain goals give you additional resources that allow you to solve the problem. Besides, this task has strong connections to video game problems which students are familiar with.

TWO BACKPACS

Two beavers are getting ready for a trip. They are packing their gear.



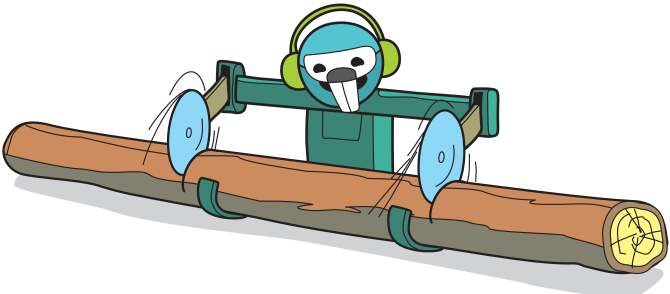


The weight of one backpack cannot exceed 8 kg.

How to distribute the things between the backpacks, so that the beavers could take as many things as possible?

Many real life problems require an optimal choice of actions or decisions. To this day, many different optimization algorithms have been created. One of them, called the greedy algorithm, requires the chooser to take the best (largest, heaviest, most profitable) component each step. However, in many cases this principle fails to provide the optimal solution and one is left to look for a better strategy.

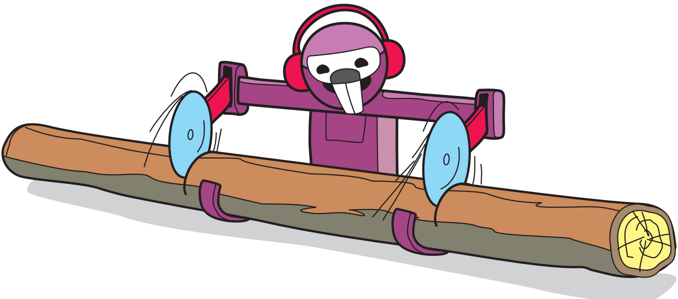
A robo-beaver cuts the log into 3 pieces in one second.



How many robo-beavers are needed to cut the log into 9 pieces in 1 second?

Parallel computing is a type of computation in which many calculations or the execution of processes are carried out simultaneously. Large problems can often be divided into smaller ones, which can then be solved at the same time. If a device can perform several operations at one and the same time, we say that it works in parallel. In this way, a computer processor works, it performs a multiple action. Parallelism is a feature of many objects, usually multiple tasks are performed simultaneously using different resources, for example each robot has its own saw, despite all of them cut the same log.

A robo-beaver cuts the log into 3 pieces in one second.



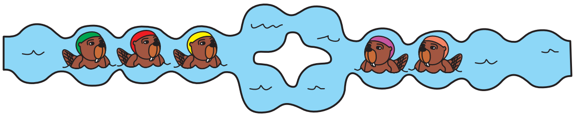
How many robo-beavers are needed to cut the log into 9 pieces in 2 seconds?

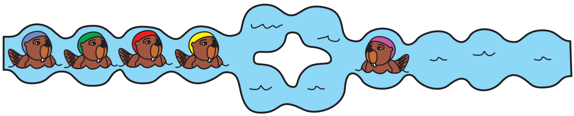
A computer's processor does some tasks by serial computing - one after another; while most of the other tasks are completed in parallel - all at the same time. The tasks are completed more rapidly when the processes are parallelized. However, parallel computing requires more resources (for example, more sawblades). This particular problem combines both serial and parallel processes.

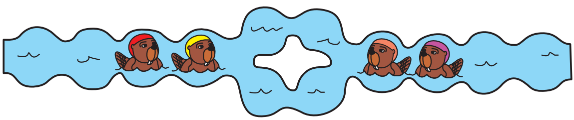
BEAVER RIVER

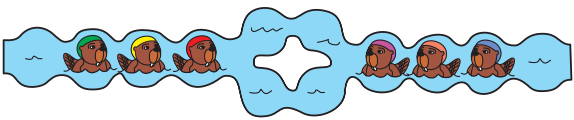
There flows a narrow stream in the Beaver Land called Beaver River. A beaver does not want to swim backwards, so it made two channels in the Beaver River for passing other teams of beavers in the opposite direction. One channel can only take up as many as two individual beavers at one time.

In which case will it be impossible for the teams of beavers to pass through?

A


B


C


D


This task demonstrates a deadlock situation. Deadlock is a situation in computing where two processes are each waiting for the other to finish. In computing, a deadlock is a state in which each member of a group of actions, is waiting for some other member to release a lock. Deadlock is a common problem in multiprocessing systems, parallel computing, and distributed systems, where software and hardware locks are used to handle shared resources and implement process synchronization.

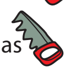
SHARING TOOLS

Beavers Bitaro and Bibako sit at a river building wooden toys. They share one hammer, one pair of scissors and one saw. They take a tool from a sandbank when they need it and they return it to the sandbank when they are done using it. If a tool they need is not on the sandbank, they wait until it is returned. Sometimes both beavers need concurrently the tool which is not on the sandbank. It this happens, beavers go to swim.

In which situation do they take a swim?

A. Bitaro has  and requires . Bibako has  and requires .

B. Bitaro has  and requires . Bibako has  and requires .

C. Bitaro has  and requires . Bibako has  and requires .

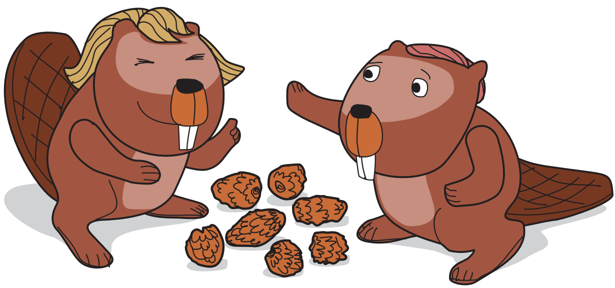
D. Bitaro has  and requires . Bibako has  and requires .

This situation is called 'deadlock'. When programs access sharing resources, the order of locking is important to prevent the deadlock. Many programs are running concurrently in networks and in computers. Sharing resources and preventing deadlock are both important. In an operating system, a deadlock occurs when a process enters a waiting state because a requested system resource is held by another waiting process, which in turn is waiting for another resource held by another waiting process. If a process is unable to change its state indefinitely because the resources requested by it are being used by another waiting process, then the system is said to be in a deadlock.

THE LAST CONE

Beavers Anna and Ben are the cone game champions. They are both playing well. Rules of the game: players take the cones in turns and pick either 1 or 2 cones on their turn. The player who takes the last cone loses the game.

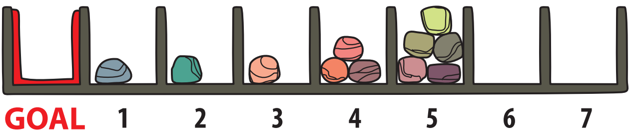
Ana starts a game. Could she win the game?



A strategy game is a game in which the players' unsourced, and often autonomous decision-making skills have a high significance in determining the outcome. Almost all strategy games require internal decision tree style thinking, and typically very high situational awareness. Strategy is any of the options players can choose in a setting where the outcome depends not only on their own actions but on the action of others. A player's strategy will determine the action the player will take at any stage of the game. A strategy on the other hand is a complete algorithm for playing the game, telling a player what to do for every possible situation throughout the game.

A STONE GAME

The goal of this game is to move all the stones to the **GOAL** box by following one simple rule: a box numbered from 1 to 7 can be emptied only if the number of stones in the box is equal to the box number itself; its stones are distributed equally on all the boxes to its left, one stone per box. The game is won when all the stones are successfully moved into the **GOAL** box.



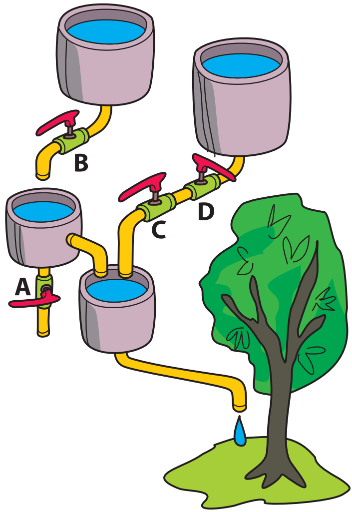
The game given in picture can be won.

Please give the order to follow in emptying the boxes.

In this game the order of operations is key for the solution: this is often the case in several informatics problems. The given winning strategy is a list of algorithmic steps that when followed, ensured the game can be won if it is winnable. The correctness of the algorithm can be proved or reasoned by mathematical induction. For only 1 box, it's clear the algorithm ensured the game can be won if there is only one stone in the box, else the game is not winnable. Assume the above algorithm is correct for up to N boxes. For N+1 boxes, there are 3 cases to consider (do that). Since we have assumed the algorithm works for up to N boxes, then by induction, the algorithm works for up to N+1 boxes.

WATER SUPPLY

Beaver has constructed a pipeline system to water his apple tree. The gates are marked by variables A, B, C, D, and can assume only one of two values: ***true*** or ***false***.
If the gate is open, then the value of its variable is ***true***. If the gate is closed, then the value of its variable is ***false***.

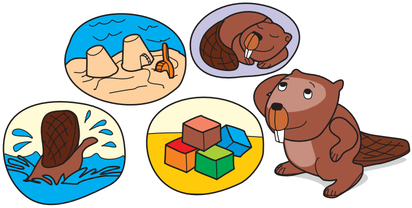


In which case does the apple tree get water?

1. A = ***true***, B = ***true***, C = ***false***, D = ***false***
2. A = ***true***, B = ***false***, C = ***false***, D = ***true***
3. A = ***false***, B = ***true***, C = ***false***, D = ***false***
4. A = ***false***, B = ***false***, C = ***false***, D = ***true***

Computer programs process data structures that model real things. A model is an abstraction, a simplified image of some real scenario. In this case the gates are represented by variables that contain the values open or closed. This is an abstraction, since all other properties of gates are ignored (for example, half-closed gates). Water flow through pipes can be calculated using mathematical logic operations. For example, a consecutive two-gate arrangement corresponds to the logical AND operation (it assumes that the true is only when both values are true), and parallel two-gate arrangement corresponds to the logical OR operation.

BEAVER'S PLAYING PLACES



A beaver decides where to play according to the weather.
The rules are as follows:

- If it is sunny today, but if it was rainy yesterday, he will swim in the river.
- If it is sunny today and if it was sunny also yesterday; he will play in the sand on the bank of the river.
- If it is rainy today, but if it was sunny yesterday, he will play with toy blocks in his house.
- If it is rainy today, and if it was rainy also yesterday, he will not play.

The following table is the weather history during November 1st - 8th

Date	1st	2nd	3rd	4th	5th	6th	7th	8th
Weather								

Where does a beaver play on 7th November?

This is a problem for questioning the thinking logical. As the condition 3 and 4 are exclusive to each other, the playing place is clearly determined. However, condition 1 and 2 are not exclusive. For example, if it was fine yesterday, both condition 1 and 2 are satisfied. This is a Finite-State Automata problem. In this problem, even if today is rainy, playing place depends on the weather "state" of the previous day. This means that two certain rainy days may be in different states. By this problem, the idea of "state transition" will be noticed. The approaches of the state transition are an important concept that is used in the design of programs, such as a vending machine.

A SPECIAL CODE

Two beavers communicate by sending messages. For safety, they encode the messages by a code developed by themself: find a letter's number in the table and multiply the number by two.

A	1	H	8	O	25	V	37
B	2	I	9	P	26	W	38
C	3	J	15	U	27	X	39
D	4	K	16	R	28	Y	45
E	5	L	17	S	29	Z	46
F	6	M	18	T	35	!	47
G	7	N	19	U	36	?	48

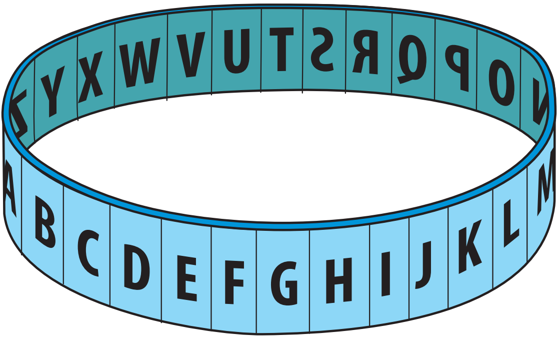
So they change message text by sequence of numbers, for example, a word BEBRAS will be coded as 410456258

How could we encode the word INFORMATICS?

The process of encoding converts information from a source into symbols for communication or storage. Decoding is the reverse process, converting code symbols back into a form that the recipient understands. One reason for coding is to enable communication in places where ordinary plain language, spoken or written, is difficult or impossible. Encoding of information is one of the basic problems in informatics. Moreover, the codes have to be designed in such a way that they are from a rich enough domain that one can distinguish between different information.

A CIPHER

Beavers are sending messages encoded with a specific cipher: every alphabet letter is shifted by two letters. For example: A->C, B->D…, Y->A, Z->B.



The following message was received:
EQOG CPF XKUKV MCPICTQQU

What did the beaver write?

Being able to keep messages secret has always been important, for instance to send sensitive information between different cities and countries. Today, being able to protect digital information is even more important, as more and more data is transmitted using the Internet and other networks. Cryptography is a field of informatics, which deals with techniques for encrypting messages so that only the sender and the intended receiver can read the content. Clearly, we do not want for instance our e-mail messages, credit card information or other private data to be readable to anyone. The encryption technique used in this task is called a shift cipher, since it shifts every letter in the alphabet a given number of positions. The method is also called Caesar cipher, as Julius Caesar used this particular approach for protecting his communication.

INFORMATICS CONCEPTS IN THE TASKS

1	Representation of information
2	Automaton
3	Binary numbers
4	Binary numbers
5	Binary logics, states
6	Binary logics, commands
7	Command, state, limitation
8	A command, a state, limitation
9	Command, parameters
10	Command, parameters
11	Algorithm, execution
12	Deterministic algorithm
13	Algorithm, program
14	Program, debugging
15	Subroutine, procedure
16	Subroutine, procedure
17	Program, testing
18	Program, automation
19	Command, repetition, loop
20	Sequence of commands, loop
21	Algorithm, repetition, loop
22	Repetition, loop, permutation
23	Data structure, stack
24	Stack, operation
25	Stack, parallel processes
26	Stack, in and out operations
27	Stack, in and out operations
28	Model, binary tree

29	Graph, number of vertexes
30	Graph, number of vertexes
31	Graph, abstraction
32	Graph, abstraction
33	Graph, the shortest path
34	The shortest path, Dijkstra's algorithm
35	Sorting algorithms
36	Bubble sort
37	Sorting, sequential search
38	Longest increasing subsequence
39	Permutation
40	Brute force search
41	Optimisation
42	Optimisation
43	Optimization, finite search
44	Greed algorithm
45	Parallel actions
46	Parallel actions
47	Parallel processes, deadlock
48	Parallel processes, deadlock
49	Game, winning situation
50	Winning strategy, induction
51	Logic operation
52	Logic operations, automaton
53	Coding
54	Ciphering
55	Informatics concepts
56	Example of the rules

RULES EXAMPLE

''Bebras'' Cards are for informatics study. There are various studying methods. One of them is given below, others are on the website www.bebras.lt.

1. Students gather together with 2-3 people in one group. Teacher explains game rules. (5 min)

2. Every group gets one card. (2 min)

3. Game for two (or three): students together solve both tasks from a card (10 min), solutions should be written on a paper.

4. Game for a group: two couples next to each other make a group. Both couples explain their tasks to each other. Group chooses the most interesting task and prepares a solution to present in front of the class. (10 min)

5. Every group presents one chosen task and its solution. (6 x 2 min = 12min)

6. Questions and additional explanations. (6 min)
Time: 45 min.

Material: 12 ''Bebras'' cards, sheet of paper, writing tool.