Bebras Australia Computational Thinking Challenge

2022 Solutions Guide
Round 2

Primary School
Grades 3–6

bebras.edu.au

Bebras Australia Computational Thinking Challenge

Bebras is an international initiative aiming to promote Computational Thinking skills among students.

Started in 2004 by Professor Valentina Dagiene from the University of Vilnius, 'Bebras' is Lithuanian for beaver. This refers to their collaborative nature and strong work ethic.

The International Bebras Committee meets annually to assess potential questions and share resources. Questions are submitted by member countries and undergo a vetting process.

The Bebras international community has now grown to 60 countries with over 2.9 million students participating worldwide!

Bebras Australia began in 2014 and is now administered through CSIRO Digital Careers.

In Australia, the Bebras Challenge takes place in March and August–September each year. As of 2020, two separate challenges are offered for each round.

To find out more and register for the next challenge, visit **bebras.edu.au**

Engaging young minds for Australia's digital future

423
Australian schools participated in Round 2 2022

25,498
Australian students participated in Round 2 2022

2.9 million
Students participate worldwide

CSIRO Digital Careers supports teachers and encourages students' understanding of digital technologies and the foundational skills they require in an ever-changing workforce. Growing demand for digital skills isn't just limited to the ICT sector. All jobs of the future will require them, from marketing and multimedia through to agriculture, finance and health. Digital Careers prepares students with the knowledge and skills they need to thrive in the workforce of tomorrow.

csiro.au/digital-careers

CSIRO digital careers

What is a Solutions Guide?

Computational Thinking skills underpin the careers of the future. Creating opportunities for students to engage in activities that utilise their critical and creative thinking along with problem solving skills is essential to further learning. The Bebras Challenge is an engaging way for students to learn and practice these skills.

Within this Solutions Guide you will find all of the questions and tasks from Round 2 of the Bebras Australia Computational Thinking Challenge 2022. On each page above the question you will find the age group, level of difficulty, country of origin and key Computational Thinking skills.

After each question you will find the answer, an explanation, the Computational Thinking skills most commonly used, and the Australian Digital Technologies curriculum key concepts featured.

Contents

What is Computational Thinking?

Computational Thinking is a set of skills that underpin learning within the Digital Technologies classroom. These skills allow students to engage with processes, techniques and digital systems to create improved solutions to address specific problems, opportunities or needs. Computational Thinking uses a number of skills, including:

DECOMPOSITION
Breaking down problems into smaller, easier parts.

PATTERN RECOGNITION
Using patterns in information to solve problems.

ABSTRACTION
Finding information that is useful and taking away any information that is unhelpful.

MODELLING AND SIMULATION
Trying out different solutions or tracing the path of information to solve problems.

ALGORITHMS
Creating a set of instructions for solving a problem or completing a task

EVALUATION
Assessing a solution to a problem and using that information again on new problems.

More Computational Thinking resources

Visit digitalcareers.csiro.au/CTIA to download the Computational Thinking in Action worksheets. These can be used as discussion prompts, extension activities or a framework to build a class project.

Each resource was designed to develop teamwork; critical and creative thinking; problem solving; and Computational Thinking skills.

Computational Thinking skills alignment

2022 Round 2 Questions	Grade level	Decomposition	Pattern Recognition	Abstraction	Modelling & Simulation	Algorithms	Evaluation
Years 3+4							
Exam Results	Easy	■	■	■		■	
Funny Filter	Easy	■	■	■		■	■
Password	Easy	■	■	■		■	■
Dancing Dress	Easy	■		■	■	■	■
Find Animal B	Easy		■		■	■	■
Chat Bot	Medium	■	■	■		■	■
Longest Sequence A	Medium	■	■	■	■	■	■
Birds Song	Medium	■	■	■	■	■	■
Between Dots A	Medium	■	■	■	■	■	■
Fruit Road	Medium	■	■	■	■	■	■
Napping Together A	Hard	■		■	■	■	
Flower Growth Phases	Hard	■		■	■	■	■
Kangaroo C	Hard	■	■	■	■	■	■
Arranging Shapes	Hard	■	■		■	■	■
Volcanoes	Hard	■		■	■	■	■
Years 5+6							
Garden of Eden, Hotel California	Easy	■	■	■	■	■	■
Cat Pictures	Easy	■	■	■		■	■
Between Dots B	Easy	■	■	■	■	■	■
Grocery Shopping A	Easy	■			■	■	■
Robot	Easy	■		■	■	■	■
Animal Sorting	Medium	■			■	■	■
Presents Program	Medium	■	■	■		■	■
FIFO Restaurant	Medium	■		■	■	■	■
Beaverley's Food Bags	Medium	■	■	■		■	■
Cupcakes B	Medium	■	■			■	■
Guess Who?	Hard	■		■	■	■	■
Three Beavers	Hard	■			■	■	■
Glass Cabinet A	Hard	■	■			■	■
Robot Drawing	Hard	■		■	■	■	■
Bank Lock	Hard	■	■			■	■

Australian Digital Technologies curriculum key concepts

Abstraction

Hiding details of an idea, problem or solution that are not relevant, to focus on a manageable number of aspects.

Data Collection

Numerical, categorical, or structured values collected or calculated to create information, e.g. the Census.

Data Representation

How data is represented and structured symbolically for storage and communication, by people and in digital systems.

Data Interpretation

The process of extracting meaning from data. Methods include modelling, statistical analysis, and visualisation.

Specification

Defining a problem precisely and clearly, identifying the requirements, and breaking it down into manageable pieces.

Algorithms

The precise sequence of steps and decisions needed to solve a problem. They often involve iterative (repeated) processes.

Implementation

The automation of an algorithm, typically by writing a computer program (coding) or using appropriate software.

Digital Systems

A system that processes data in binary, made up of hardware, controlled by software, and connected to form networks.

Interactions

Human-Human Interactions: How users use digital systems to communicate and collaborate.
Human-Computer Interactions: How users experience and interface with digital systems.

Impact

Analysing and predicting how existing and created systems meet needs, affect people, and change society and the world.

For more information on the Digital Technologies curriculum, please visit the Australian Curriculum, Assessment and Reporting Authority (ACARA) website: australiancurriculum.edu.au/f-10-curriculum/technologies/digital-technologies

Digital Technologies key concepts alignment

2022 Round 2 Questions	Grade level	Abstraction	Data Collection	Data Representation	Data Interpretation	Specification	Algorithms	Implementation	Digital Systems	Interactions	Impacts
Years 3+4											
Exam Results	Easy										
Funny Filter	Easy										
Password	Easy										
Dancing Dress	Easy										
Find Animal B	Easy										
Chat Bot	Medium										
Longest Sequence A	Medium										
Birds Song	Medium										
Between Dots A	Medium										
Fruit Road	Medium										
Napping Together A	Hard										
Flower Growth Phases	Hard										
Kangaroo C	Hard										
Arranging Shapes	Hard										
Volcanoes	Hard										
Years 5+6											
Garden of Eden, Hotel California	Easy										
Cat Pictures	Easy										
Between Dots B	Easy										
Grocery Shopping A	Easy										
Robot	Easy										
Animal Sorting	Medium										
Presents Program	Medium										
FIFO Restaurant	Medium										
Beaverley's Food Bags	Medium										
Cupcakes B	Medium										
Guess Who?	Hard										
Three Beavers	Hard										
Glass Cabinet A	Hard										
Robot Drawing	Hard										
Bank Lock	Hard										

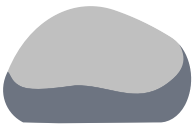
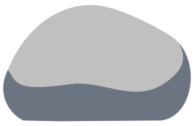
Bebras Challenge 2022 Round 2

Years 3+4

This question comes from
Egypt

Exam Results

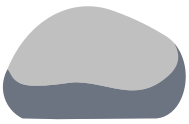
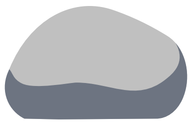
A teacher at Beaver Primary School sends their students a secret number using pictures of rocks and trees. The student beavers are given this picture:

Cell 1	Cell 2	Cell 3	Cell 4		Total Score
8 Points	4 Points	2 Points	1 Point		
				=	9

The students find the number by adding the points of the cells that contain a tree. So they add 8+1=9.

Question

The teacher sends the students a new picture. What is the new secret number?

Cell 1	Cell 2	Cell 3	Cell 4
8 Points	4 Points	2 Points	1 Point
			

3 6 9 12

EXPLANATION

Answer

Correct answer is 6.

Explanation

The student beavers should add the 2nd and 3rd cells which include the trees.

So the secret number will be 4+2=6.

Continued on next page

Exam Results – continued

BACKGROUND INFORMATION

Computers use *binary* – the digits 0 and 1 – to store data. A binary digit, or bit, is the smallest unit of data in computing. It is represented by a 0 or a 1. Binary numbers are made up of binary digits (bits), eg the binary number 1001 is equal to 9.

The circuits in a computer's processor are made up of billions of *transistors*. A transistor is a tiny switch that is activated by the electronic signals it receives. The digits 1 and 0 used in binary reflect the on and off states of a transistor.

Computer programs are sets of instructions. Each instruction is translated into machine code - simple binary codes that activate the CPU. Programmers write computer code and this is converted by a translator into binary instructions that the processor can execute.

All software, music, documents, and any other information that is processed by a computer, is also stored using binary.

To help you develop a better understanding of the binary system and how it relates to the decimal system you're familiar with, here's how the decimal numbers 1-10 look in binary:

1 = 0001

2 = 0010

3 = 0011

4 = 0100

5 = 0101

6 = 0110

7 = 0111

8 = 1000

9 = 1001

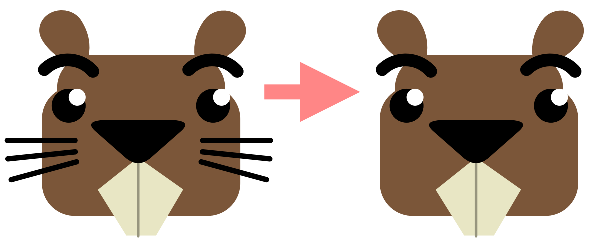
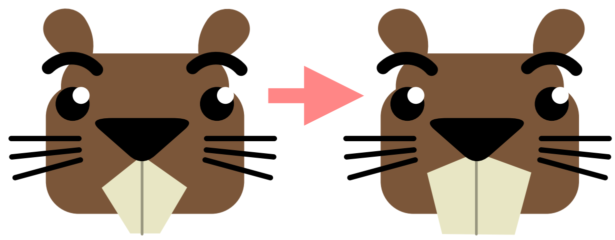
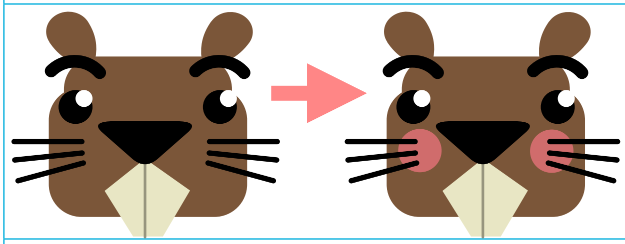
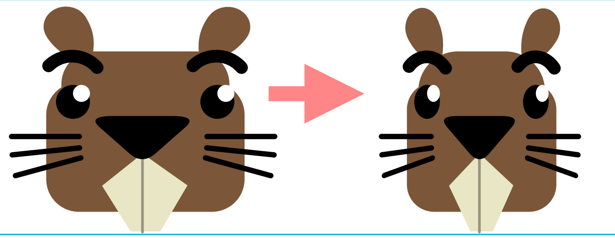
10 = 1010

This question comes from
Taiwan

Years 3+4 Easy
Years 5+6
Years 7+8
Years 9+10
Years 11+12

Funny Filter

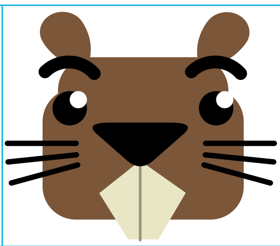
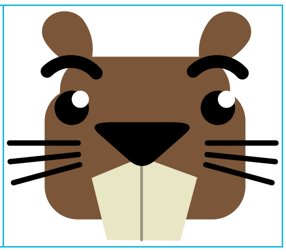
A photo app has four funny filters; each filter has a different effect shown below:

	
Remove Whiskers	**Enlarge Teeth**
	
Apply Blush	**Reshape Face**

After Little Beaver applied the two filters “apply blush” and “reshape face” on a photo, the photo looks like this:

Question

What might the original photo look like?

Continued on next page

Funny Filter continued

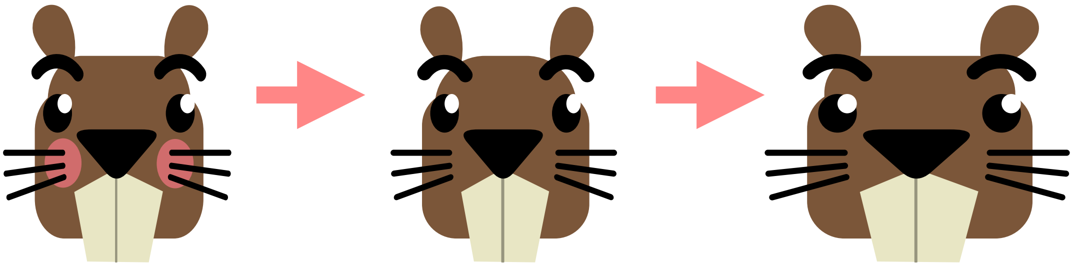
EXPLANATION

Answer

Explanation

Since the “enlarge teeth” filter is not used, we can infer that the teeth in the original photo are already large; therefore, option A and C, both showing relatively small teeth, are not correct. Also, little beaver has whiskers on the final photo, and no filter that adds whiskers was applied, so the original photo has to have whiskers. Hence, options A and B are not correct.

We can also find the answer by reversing the effect that the two filters had on the photo, as shown in the photo below. After removing the “apply blush” filter, we will get the photo in the middle. After removing the “reshape face” filter from the middle photo, we will get the photo on the right, which is the original photo.



BACKGROUND INFORMATION

In this task, we are describing the face of beavers using a set of properties, sometimes called attributes.

Each property of a beaver can have different values:

- Teeth can be short or long
- Whiskers can be present or absent
- Head shape can be round or thin
- Cheeks can be bare or have blush

Each funny filter changes the value of one of these properties.

When computers manipulate objects of the real world, they often represent them by such lists of properties. People in a computer database have properties such as their firstname, lastname, genre, birth date, birth place, etc. We can then perform searches based on the value of some of these properties, such as finding all the people called Bob who were born in 2004 in Lithuania.

This question comes from
Egypt

Years 3+4 Easy
Years 5+6
Years 7+8
Years 9+10
Years 11+12

Password

Beavers have a special way to send messages. They use this table to change the letters into numbers, and then they send the numbers to each other:

A	B	C	D	E	F	G	H	I	J
65	66	67	68	69	70	71	72	73	74

If a beaver would like to send “HI” to their friend, they will send “72.73”.

Question

If a beaver sends “72.73 74.69.70.70” to their friend, what is their friend’s name?

Jace | Jeff | Jada | Jedi

EXPLANATION

Answer

The correct answer is “JEFF”.

Explanation

From the table, the beaver could change every number to its corresponding letter. So, 74.69.70.70 will be JEFF.

You can also solve this in a quicker way. The encoded message ends with 70.70, so the last two letters are the same. There is only one answer where the last two letters are the same.

BACKGROUND INFORMATION

Computers can only store numbers internally. This is why when using letters you need what is called an ‘encoding’, a way to translate numbers into letters and vice versa.

ASCII is a table of characters for computers. It is binary code used by electronic equipment to handle text using the English alphabet, numbers, and other common symbols. ASCII is an abbreviation for American Standard Code for Information Interchange, and it was developed in the 1960s based on earlier codes used by telegraph systems.

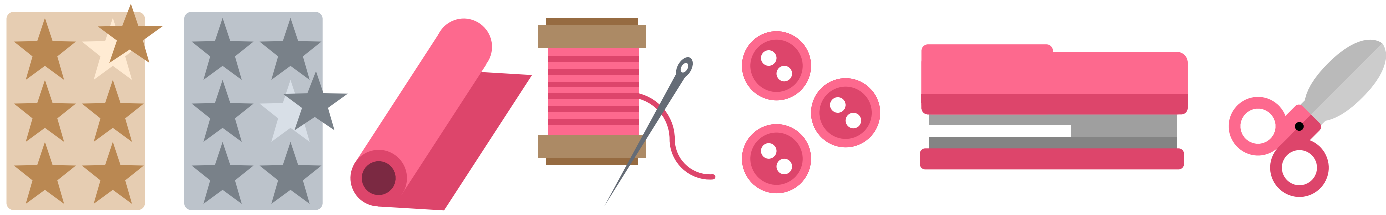
The code includes definitions for 128 characters: most of these are the printable characters of the alphabet such as abc, ABC, 123, and ?&!. There are also control characters that cannot be printed but instead control how text is processed, to start a new line, for example. Most of the control characters are no longer used for their original purpose.

ASCII uses 8 binary digits (bits) to represent characters: 1000001 (65 in standard base-10 numbers) represents the upper-case letter A; 1000010 represents B (66); 1000011 represents C (67); and so on in sequence. Eight bits allow a parity bit to be included in each byte sent over a serial port or modems. *Parity bits* are used to detect errors - each character has a number which if changed, alerts the person who is receiving the message that there was an error in transit. This was more important years ago, when connections were often unreliable.

This question comes from
Ireland

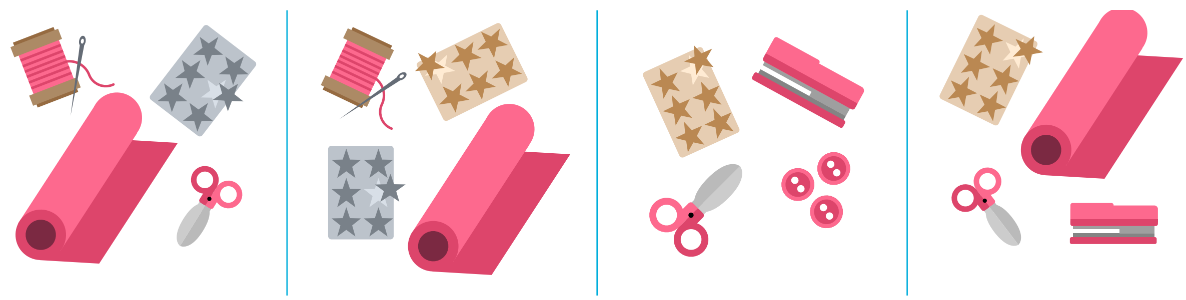
Dancing Dress

Zuri needs to make a costume for a dancing competition. Zuri goes to a shop that sells craft bags containing a selection of the items shown in the picture below.



Question

Zuri needs pink fabric, scissors, and gold stars to make the costume.
Which of these craft bags should Zuri buy?



EXPLANATION

Answer

Explanation

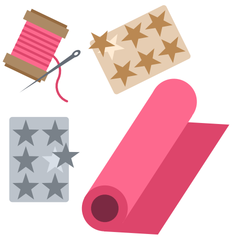
Zuri needs to choose a bag that contains pink fabric, scissors, and gold stars.

Craft Bag D (shown above) is correct because it contains pink fabric, gold stars, and scissors.

Continued on next page

Dancing Dress – continued

Craft Bag A,  , is not correct because it does not contain gold stars.

Craft Bag B,  , is not correct because it does not contain scissors.

Craft Bag C,  , is not correct because it does not contain pink fabric.

Each bag also contains at least one item that is not required, but this does not affect the answer.

BACKGROUND INFORMATION

This task involves *conditional statements*. A conditional statement tells a computer program to do different actions depending on whether a condition is **true** or **false**.

In this task we have three conditions that all have to be true in order for Zuri to buy that bag. Conditions and their evaluation are an important part of computer programming and algorithmic thinking.

This task also teaches elements of *abstraction*, in that the students must abstract away from the individual items, ignoring distractor items, to obtain true/false values whether each bag satisfies particular properties or not.

This task can also be used as an easy example to introduce pattern recognition to young students, if the students solve it by constructing their own bag of correct items from the selection in the question, and then visually comparing bags to find the closest matching bag.

This question comes from Lithuania

Years 3+4 Easy
Years 5+6
Years 7+8
Years 9+10
Years 11+12

Find Animal

Beaver Bindi can find animal names on the board in class by using arrows. For example, Bindi puts the arrow below on the letters to spell out the word FOX:

Question

In the picture below, find the name of the animal represented by the arrow next to the table.

MOUSE | HORSE | BEAVER | PANDA | BEAR

EXPLANATION

Answer

The correct answer is PANDA.

With the given arrow, we can go through the table and find where we have to put it so that it spells out the name of an animal.. According to the shape of the arrow (height is 3 cells and width is 2 cells) we can see that we have to check all the possibilities on the first line, counting from the top (3 options) and all the possibilities on the second line, there are also 3 options. In third line our arrow doesn't fit, because it's height is 3 cells. So during this process in the first line, starting with second cell we found the word PANDA.

This can also be done by process of elimination: there is no M or H in the picture, and there is no R next to an A, so Mouse, Horse, and Bear can be ruled out. Looking at the letter B, the word BEAVER can be made using an arrow, but it would have to go immediately down from B to E, and this is the wrong shape, so Beaver can be ruled out. Finally, moving to the letter P, PANDA can be spelled and the arrow is the right shape.

BACKGROUND INFORMATION

The task is about rows and columns of letters, a type of *two-dimensional array*, and searching continuous sequence of positions according to a pattern. Using programming patterns is very common when creating various programs.

Chat Bot

Dr. Beaver has made an app called “Botty”, a virtual assistant on mobile phones. Every time you ask a question, Botty opens an app based on the “keyword” your question starts with:

When the question starts with...	then Botty opens a...
Where	Map app
What	Search Engine App
How	Video App

If the question does not start with the keywords listed above, Botty **will not respond**.

For example, when you ask “How to learn to code”, Botty opens a video app; when you ask “Will it rain tomorrow”, Botty **will** not respond.

Question

Which of these statements are correct? You are able to choose more than one.

- **When asked “When is Easter”, Botty activates a search engine app**
- **When asked “What is a decimal”, Botty activates a search engine app**
- **When asked “When is Easter”, Botty activates a search engine app**
- **When asked “How to get to the train station”, Botty activates a map app**

Continued on next page

Chat Bot – continued

EXPLANATION

Answer

B. When asked "What is a decimal", Botty activates a search engine app.

C. When asked "Where is my eraser", Botty activates a map app.

Explanation

A. When asked "When is Easter", Botty activates a search engine app.
B. When asked "What is a decimal", Botty activates a search engine app.
C. When asked "Where is my eraser", Botty activates a map app.
D. When asked "How to get to the train station", Botty activates a map app.

- Although option A seems logical for a human (activating a search engine app based on the keyword "When"), Botty should not respond, since the keyword "When" is not in the keyword list.
- For option B, Botty activates a search engine app based on the keyword "what". This option fits not only the Dr. Beaver rules, but also human logic.
- For option C, Botty activates a map app based on the keyword "where". Although Botty's response seems illogical for a human, Botty indeed follows the rules.
- In option D, Botty should activate a video app based on the keyword "how". Although activating a map app fits human reaction, it does not fit the rules set by Dr. Beaver.

So the correct descriptions are B and C.

BACKGROUND INFORMATION

This task introduces basic concepts of *virtual assistants*. A *chatbot* is one example of virtual assistants. Some chatbots are equipped with *natural language processing systems*. Natural language processing explores how to program computers to process and analyze natural language data.

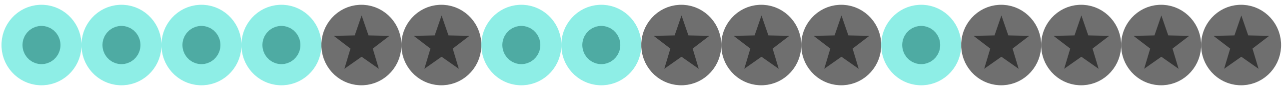
Basic natural language process system only retrieve some keywords and find an appropriate response from the *corpus* (a collection of language materials), where complex natural language process system can deal with large amount of langauge data. Since apps behaviour depends only on programmer work, it is important to provide an appropriate logic that would be similar to human one while programming.

This question comes from Ukraine

Years 3+4 Medium
Years 5+6
Years 7+8
Years 9+10
Years 11+12

Longest Sequence

Your friend Connie made you a chain using 16 beads. Connie used two types of beads:



You notice the longest sequence of the same colour beads in a row is 4 - the blue beads on the left, and the black beads on the right.

Connie allows you to change any two beads in the chain, to the colour of the other bead.

Question

After changing two beads, what is the longest possible sequence of the same colour beads in a row inside the chain?

7 | 8 | 9 | 10

EXPLANATION

Answer

The correct answer is 9.

Explanation

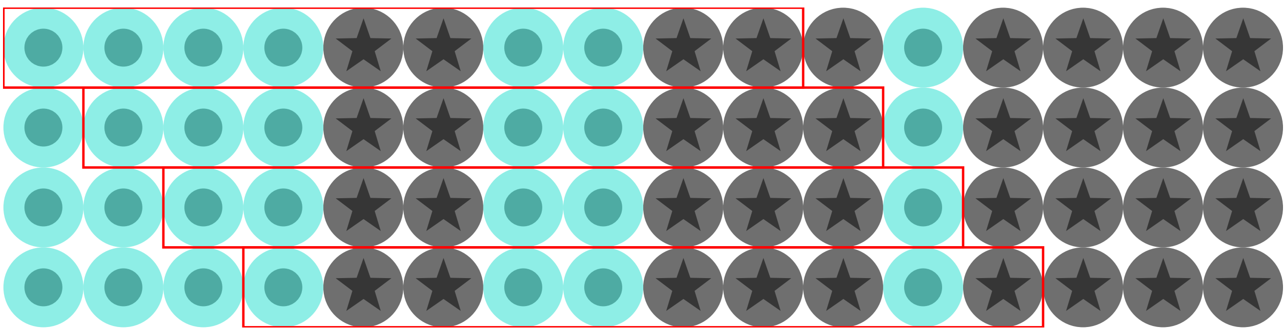
To show this, we need to prove two things:

1. that an unbroken chain of length 9 is possible, and
2. that an unbroken chain of length greater than 9 is not possible.

The first part is easy to prove. Here is how an unbroken chain of 9 stars can be made:

To prove that an unbroken chain of length greater than 9 is not possible, consider any chain of length 10. Since we are only allowed to change two shapes, any chain of length 10 in the original sequence must already have eight identical shapes in it.

There are seven chains of length 10 in the original sequence, a few of which are shown below. In no chain can eight identical shapes be found. Convince yourself of this for the remaining unshown three chains.



Continued on next page

Longest Sequence – continued

Since it is not possible to have an unbroken chain of length 10, it is certainly not possible to have an unbroken chain of length greater than 10.

Thus, we have shown that the length of the longest unbroken chain of identical shapes possible is 9.

BACKGROUND INFORMATION

This task is related to finding the *longest substring* that matches some given criteria.

There are many instances in informatics where finding the longest substring is useful, in particular, finding the longest *common* substring given two strings.

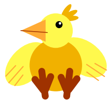
Finding the longest common substring can help detect plagiarism, and help compress data by data reduplication (removing redundant copies of data).

Some techniques that can be used to find the longest sequences include the two pointer method, and the sliding window.

This question comes from
Pakistan

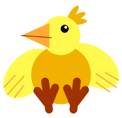
Birds Song

Years 3+4 Medium
Years 5+6
Years 7+8
Years 9+10
Years 11+12

A bear living in a cave in Beaverland predicts weather each morning by listening to the sounds of three birds (bird  Ollie, bird  Sandy and bird  Felix) living nearby.

If the sounds follow all of the 3 rules below, the bear believes that the weather will be good and will come out of the cave. Otherwise the bear will stay in the cave.

Conditions for good weather:

1. All three birds sing at least one sound.
2. Ollie  sings exactly one sound.
3. Sandy  sings at least two sounds in a row.

Question

Which of the following sequences of bird sounds will make the bear come out of the cave?

S-F-O-F-F-S-F | **S-S-S-O-S-S-S** | **F-S-O-S-S-F-F** | **S-F-O-S-S-O-F**

Continued on next page

Birds Song – continued

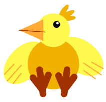
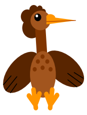
EXPLANATION

Answer

The correct answer is F-S-O-S-S-F-F, because all three birds sing,  (S) sings twice in a row and  (O) only once.

Explanation

The other answers are not correct:

- In sequence S-F-O-F-F-S-F  (S) does not sing twice in a row.
- In sequence S-S-S-O-S-S-S  (F) does not sing at all.
- In sequence S-F-O-S-S-O-F  (O) sings twice.

BACKGROUND INFORMATION

This question is related to the concepts of *pattern recognition* and *classification*, which are common tasks in many computer science applications.

Pattern recognition refers in general to the task of automatic discovery of regularities (= “pattern”) in data in order to for example *classify* the data into some category. In this task the three conditions for good weather specified a pattern, and the data consisted of a sequence of bird sounds. The goal was then to classify the data to one of two categories — “good weather” or “bad weather” — based on whether we could recognise the good weather pattern from the data. This type of classification problem that has only two categories is known as *binary classification*.

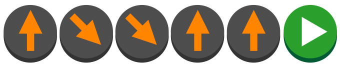
This question comes from
Iceland

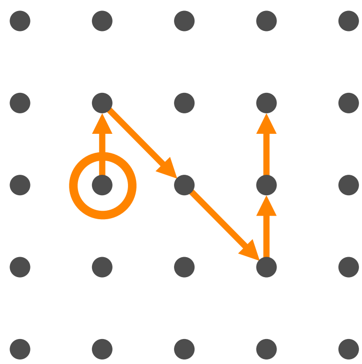
Between Dots

Emma plays with a “DoodleBot” that draws lines between dots.

She pushes arrow buttons to send the robot to the next dot.

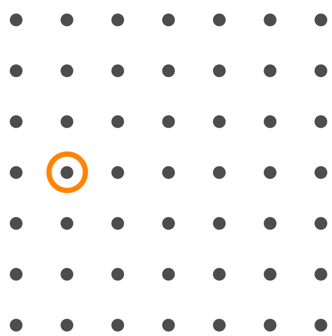
The robot starts on the dot with the circle around it.

This arrow sequence  tells the robot to draw:

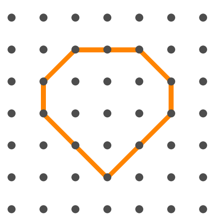


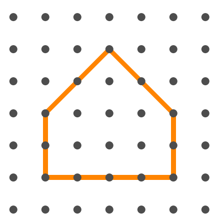
Question

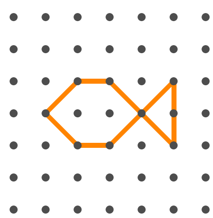
Emma pushed the buttons in this sequence, and started on the dot with an orange circle around it:

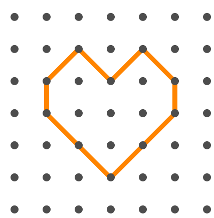


What did the robot draw in this grid?







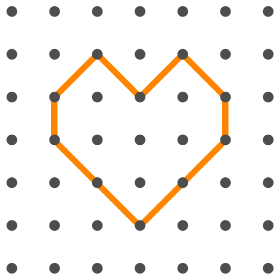


Continued on next page

Between Dots – continued

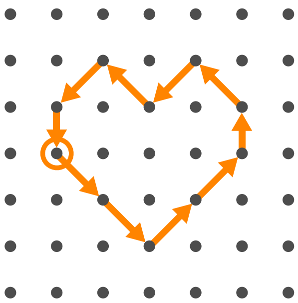
EXPLANATION

Answer



Explanation

The robot starts on the dot with the surrounding circle and the arrows shows how it follows the commands:



BACKGROUND INFORMATION

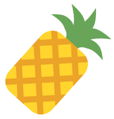
Computer programs are a series of commands that tell a device what to do. All computer programs are run (activated) by pressing a button or typing a command.

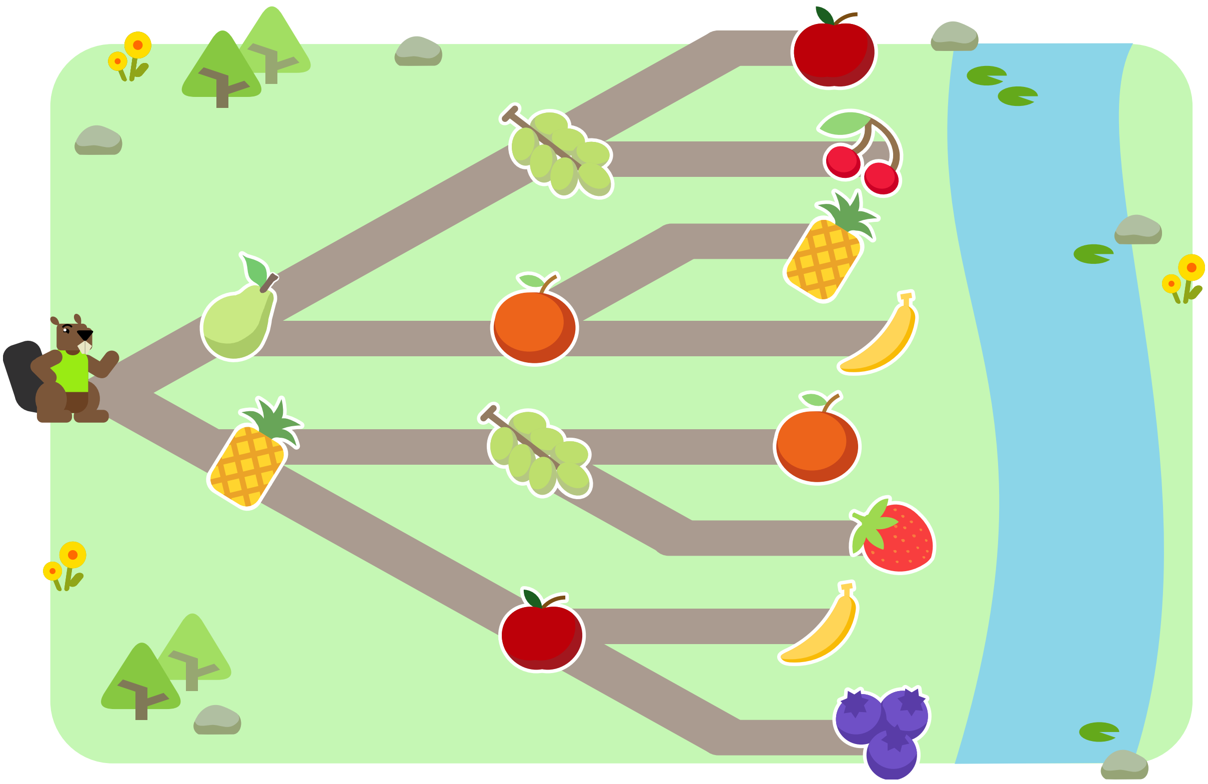
One of the basic things in informatics and robotics for students to learn is that a robot or a computer follows commands in a sequence / specific order and executes actions in that order. Students will learn that a computer or robot will only respond to commands it has been given. The commands must be precise. Students use algorithmic thinking to follow the sequence of button commands in the question. They will also look at the data representations given in the multiple choice answers to select the correct image representation. Using recipes as a metaphor might help students to understand better. The steps of cooking are like instructions, and only correct instructions can make delicious meals.

It is important to understand the instructions and execute them step by step to see what is happening. These steps will be useful for debugging. Debugging is the process of working through a program sequence to see where errors might have occurred. It is useful to have students 'think-aloud' (talk through actions as they are making them), especially when debugging. Discussing where they might have gone wrong is a useful process for students to learn that we all make mistakes and need to follow steps to fix them. Even the best computer programmers make mistakes. Debugging is a natural part of the process of learning.

Fruit Road

Beaver Nerida walks from her home to the river and collects all the fruits on her path. At the end of her walk, she sits down on the river bank and eats each of the three fruits she has picked.

She wants to eat a pineapple  but NOT an orange  .



Question

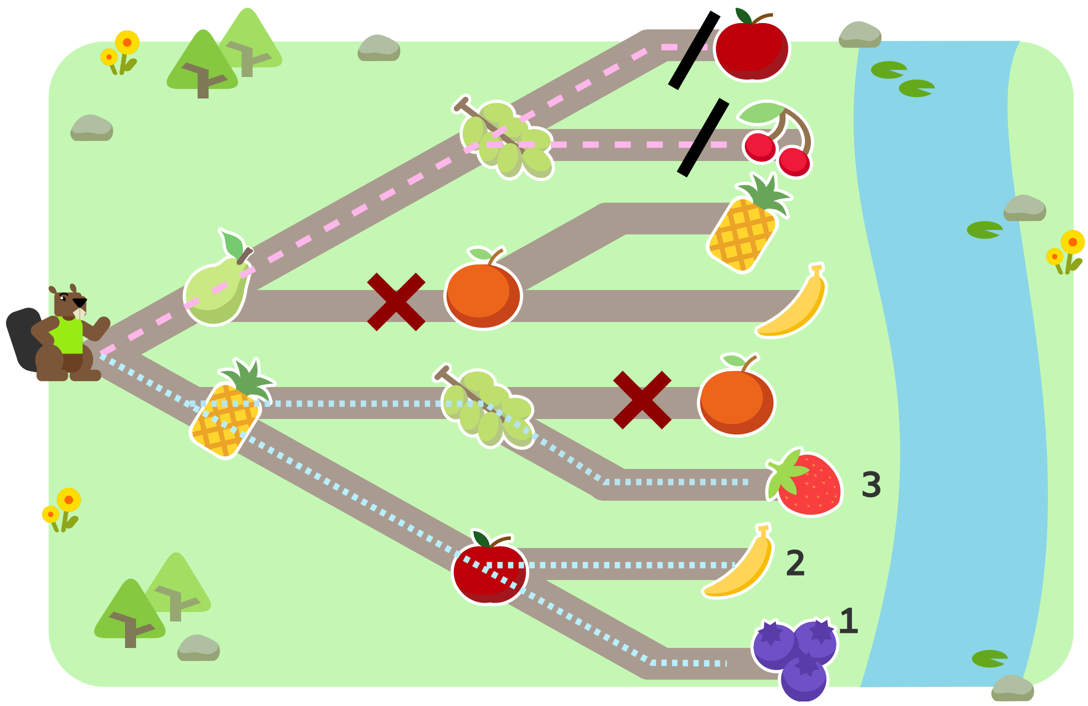
How many different paths could she use?

Fruit Road – continued

EXPLANATION

Answer

The correct answer is 3.



Explanation

There are exactly three paths that contain a pineapple but not an orange, as shown on the picture.

BACKGROUND INFORMATION

On the paths, when Nerida reaches a fruit, she picks it then decides where to go moving forward. She can choose only one direction at a time from a fruit towards the river.

In the figure, the paths are drawn as *edges* and the fruits as *vertices*. All paths to the river are one-way.

In informatics, such figures are called *graphs*. These types of graphs (without a cycling walk) are called *trees*.

This graph has a *root* - the residence of the beaver - from where the paths begin. The graph consists of edges - the paths, that connect vertices - the fruits. The ends of the paths are at the river, these vertices are called *leaves* of the graph.

At each fruit - at each vertex - Nerida has to decide where to go forward, which path she chooses.

In informatics, we call such figures *decision trees*.

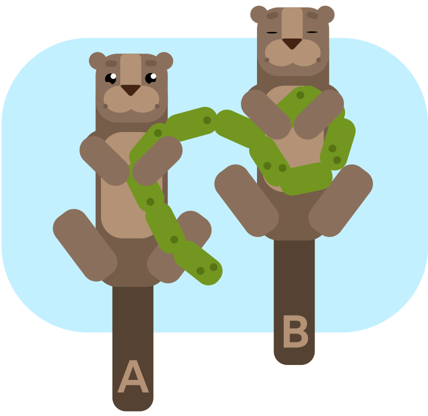
We can count the paths from Nerida's residence to the river - the edges from the root to the leaves on a graph. There are special paths that meet certain conditions, for example the path where a pineapple is present and a path where an orange isn't on them.

Napping Together

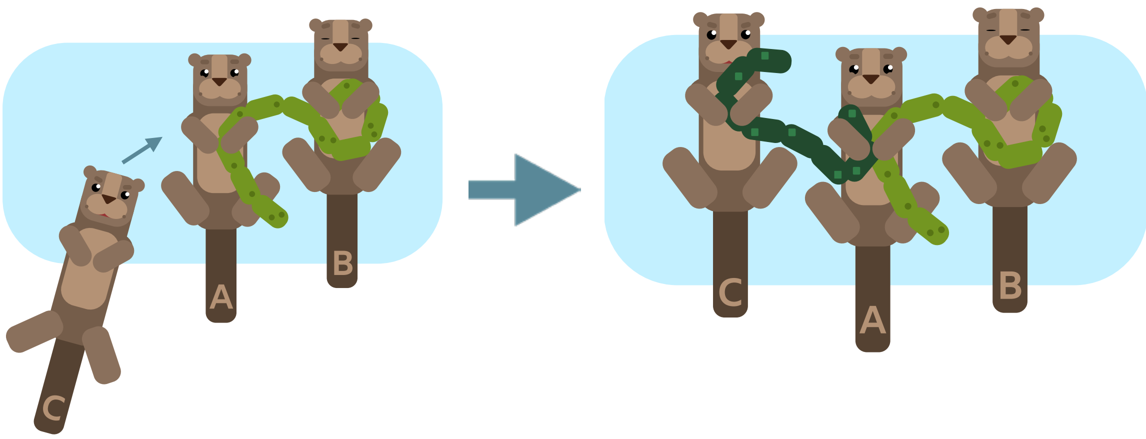
When two otters in Otter Kingdom meet each other, they will wrap seaweed around themselves so that they can stay together during nap time. However, to avoid knots, if two otters are already connected through a seaweed chain, they won't wrap another seaweed.

For instance, if otters meet each other in the following order: A - B, A - C, B - C:

1. Otter B meets Otter A. They wrap seaweed around themselves.



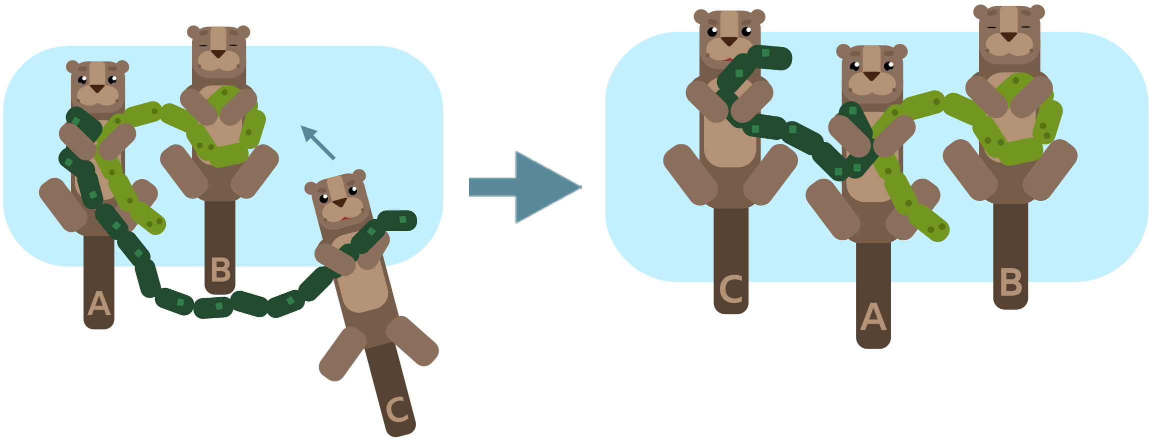
2. Otter C meets Otter A. They wrap seaweed around themselves.



Continued on next page

Napping Together – continued

3. Otter C meets Otter B. Since they are already connected through Otter A, they *won't* wrap seaweed around themselves.



Question

Otters meet each other in the following order:

A - B, A - C, B - C, D - E, A - E, D - F, A - F

How many seaweeds are wrapped around otter A?

EXPLANATION

Answer

3.

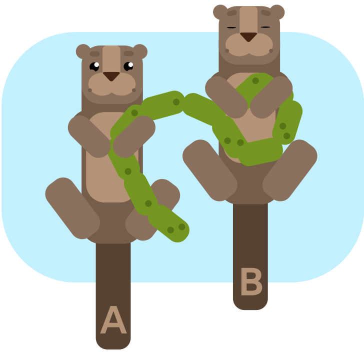
Continued on next page

Napping Together – continued

Explanation

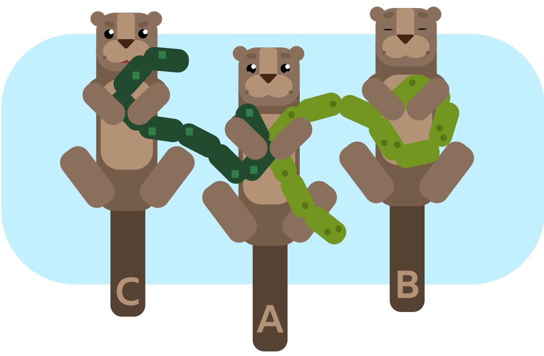
A meets B. They wrap seaweed around themselves.

One seaweed is wrapped around otter A at this point.

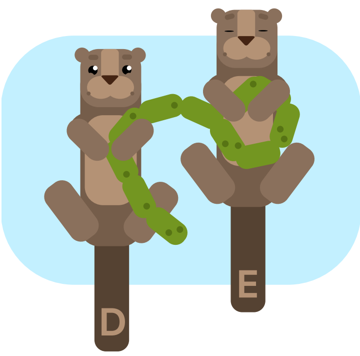


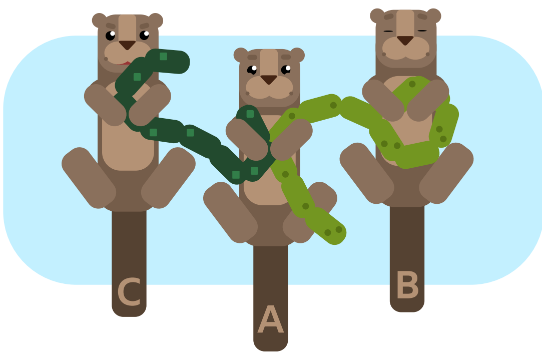
A meets C. They wrap seaweed around themselves.

Two seaweeds is wrapped around otter A at this point.



B meets C. Since they are already conneted, they won't wrap any seaweed around themselves.
D meets E. They wrap seaweed around themselves.





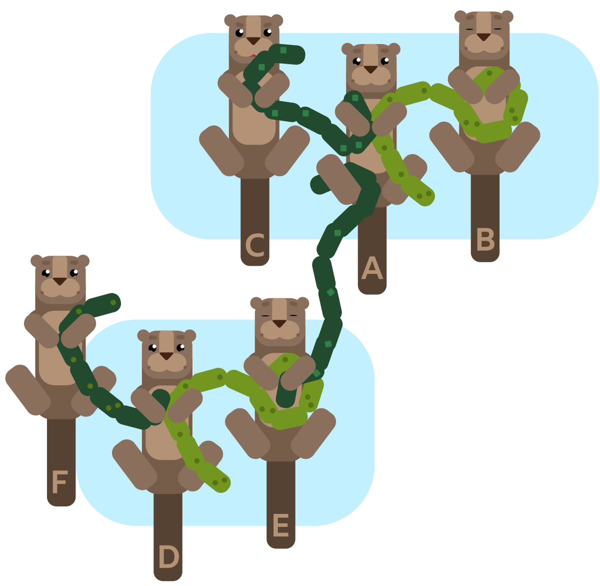
Continued on next page

Napping Together – continued

A meets E. They wrap seaweed around themselves.

Three seaweeds is wrapped around otter A at this point.

D meets F. They wrap seaweed around themselves. A meets F. Since they are already connected, they won't wrap any seaweed around themselves.



Thus, only three seaweeds are wrapped around otter A.

BACKGROUND INFORMATION

"Disjoint-set" or "union-find" is a data structure that stores a collection of *disjoint* (non-overlapping) sets. It provides operations such as adding new sets and merging sets. In this task, we merge two sets by wrapping a seaweed around two otters.

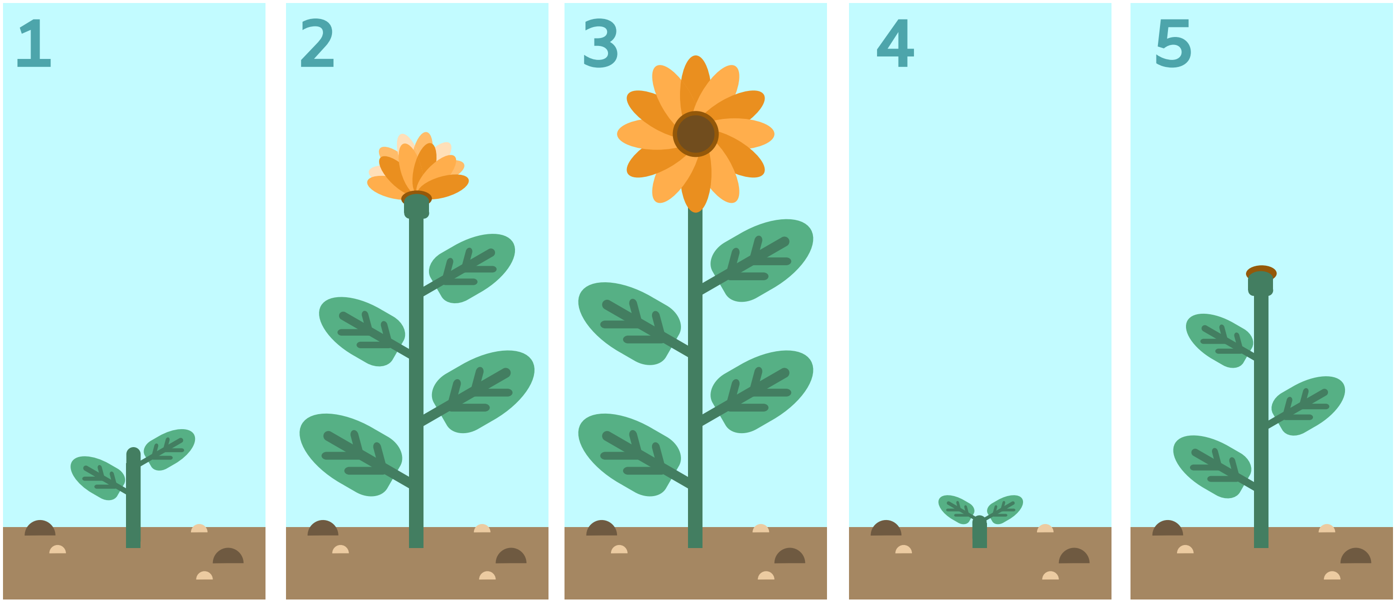
A disjoint-set is an important data structure when implementing *Kruskal's algorithm* to find a minimum spanning tree in a graph. A disjoint-set is used for preventing cycles in the selected edges, which is exactly how we avoid knots in this task.

This question comes from
Saudi Arabia

Years 3+4 Hard
Years 5+6
Years 7+8
Years 9+10
Years 11+12

Flower Growth Phases

Beaver Bo took five pictures of a flower growing for a science project, but they came out in the wrong order. He wants to rearrange his pictures in the correct order from left to right to display them in his class. He can only swap (any) two pictures at a time.



Question

What is the minimum number of swaps needed to put the pictures in the correct order?

3 | 4 | 5 | 6

EXPLANATION

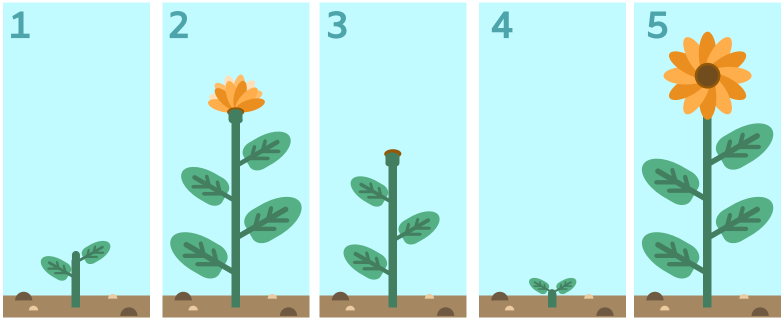
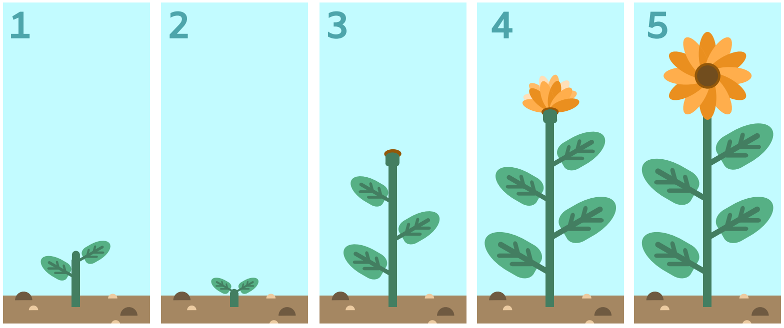
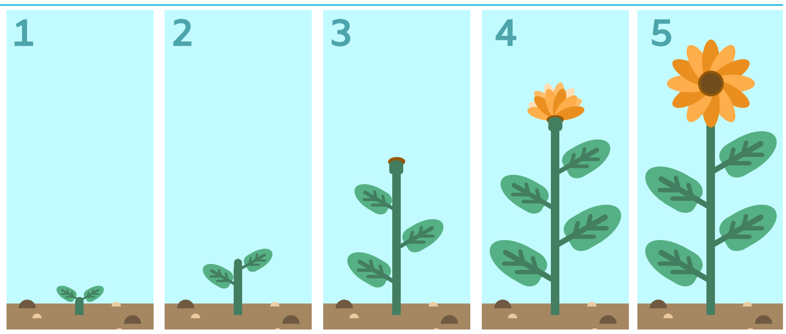
Answer

The correct answer is 3 swaps.

Flower Growth Phases – cont’d

Explanation

To reach the correct order we need to swap the pictures as shown in the following table:

Swaps	Results
1. Swaps pictures at positions 3 and 5	
2. Swaps pictures at positions 2 and 4	
3. Swaps pictures at positions 1 and 2	

BACKGROUND INFORMATION

This question touches on a concept called a *Swap Function*. In computer science, a swapping process refers to mutually exchanging the values of two variables.

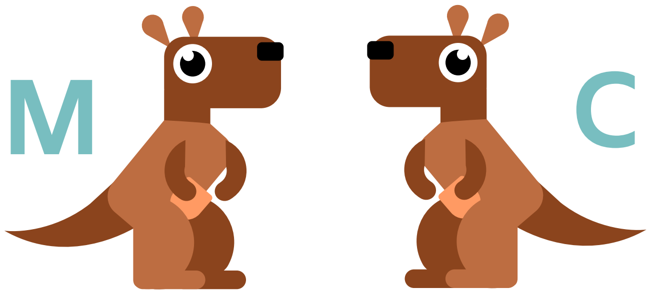
Usually, this process is executed in memory. This operation can be performed for different data types, such as strings and aggregated data types.

In many cases, programmers are required to swap values of two variables while creating a program. So, this is a very common function used in all programming languages.

In computer programming, the *exclusive “OR” swap* (sometimes shortened to XOR swap) is an algorithm that uses the exclusive or bitwise operation to swap the values of two variables without using the temporary variable which is normally required.

Kangaroo

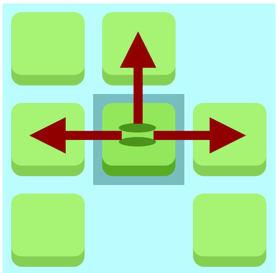
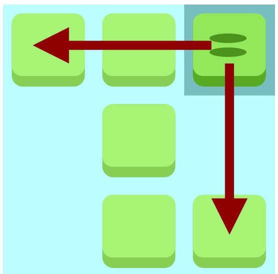
Kangaroo Mae (M) needs to reach kangaroo Claire (C) by jumping on islands in a swamp.



The swamp is represented as a grid as shown below.

Mae can only perform two kinds of jumps: short and long.

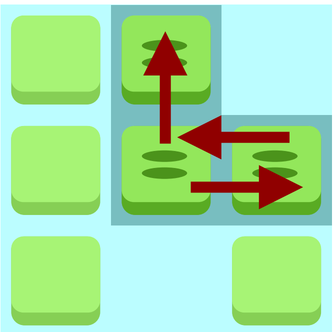
- Short jumps are to move from one island to any of the four neighbouring islands.
- Long jumps are to jump straight *over* any of the four neighbouring spaces to an island two spaces away. Long jumps are more tiring and dangerous, so she cannot make two long jumps in a row.

Short Jump	Long Jump

Mae cannot make any other jumps (diagonal moves, jumps over two or more spaces, etc).

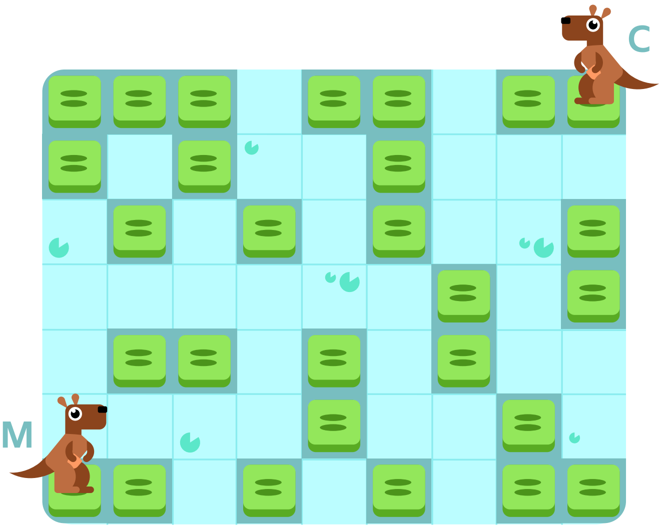
However, Mae is able to revisit an island where she has already been before:



Kangaroo – continued

Question

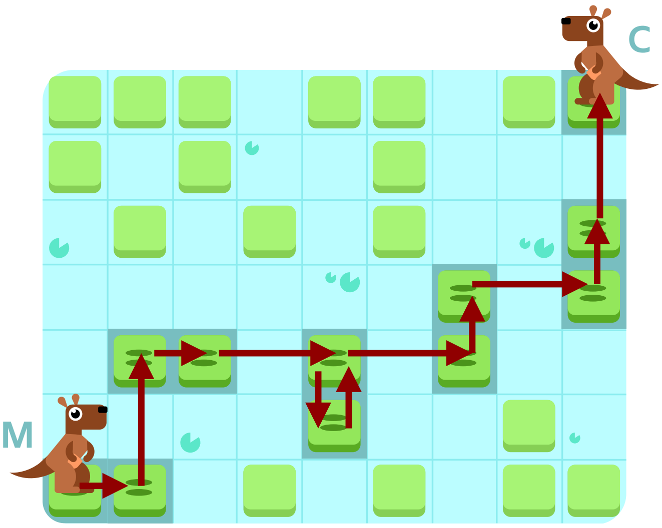
Make a path for Mae to get to Claire by clicking on the islands that Mae jumps on.



EXPLANATION

Answer

The image below shows the solution to the problem:



Explanation

The cells with dark blue background are the cells that could be visited. A correct path is in red.

When you visit a dead end road, one way to find the solution is to leave the boxes already visited marked and to go back to the previous intersection.

BACKGROUND INFORMATION

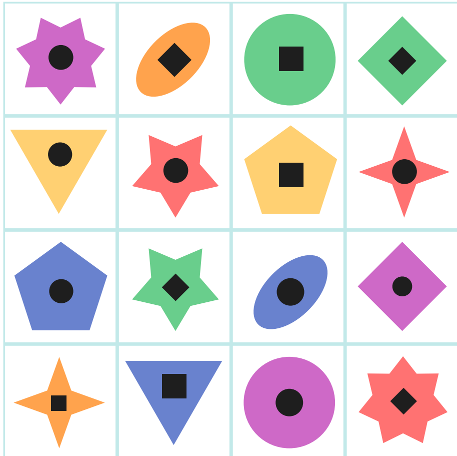
The problem uses a version of *Breadth First Search* - a modified *Lee algorithm*. Lee's algorithm simulates the motion of a wave crest that floods adjacent cells. In this problem, cells will also be adjacent, if they are reached by two jumps of a kangaroo: first a long one, then a short one.

This question comes from
India

Years 3+4 Hard
Years 5+6
Years 7+8
Years 9+10
Years 11+12

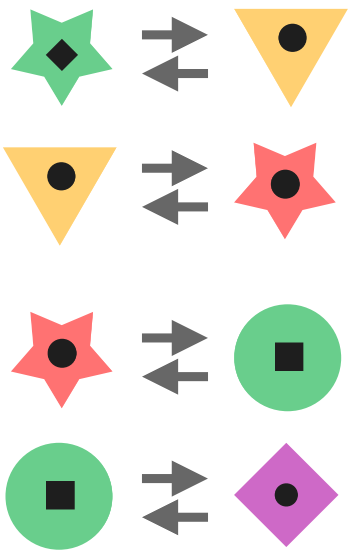
Arranging Shapes

Rinki the beaver is playing with her shapes board game. The board is divided into multiple squares. She starts the game off with the shapes placed in each square like this:



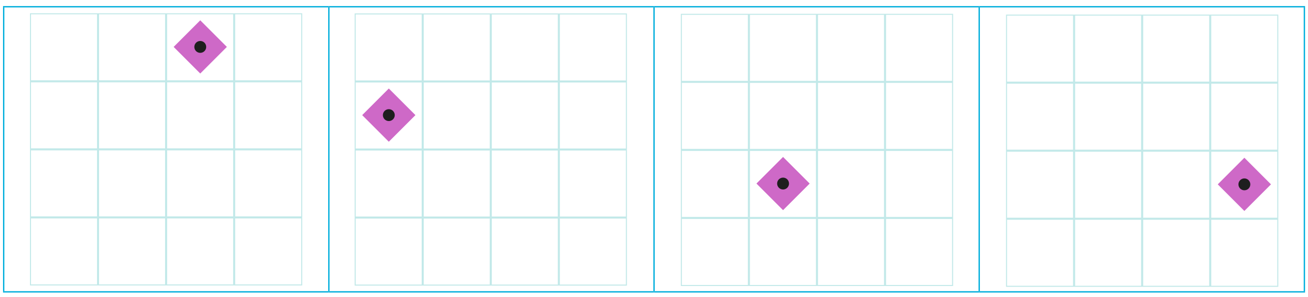
As she plays, she swaps various pairs of shapes, meaning that she makes the two shapes exchange their position.

She makes four swaps, in the following order:



Question

What will the position of  be after the last swap?



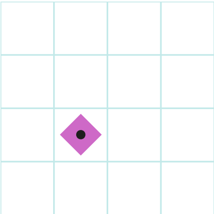
This question comes from India

Years 3+4 Hard
Years 5+6
Years 7+8
Years 9+10
Years 11+12

Arranging Shapes – continued

EXPLANATION

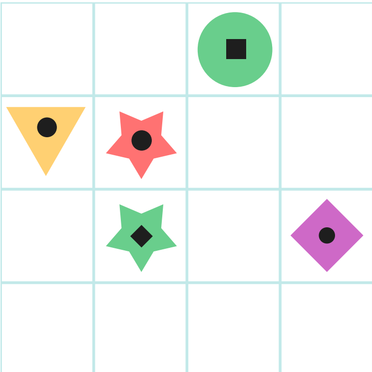
Answer

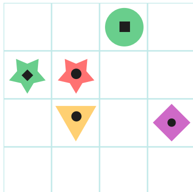
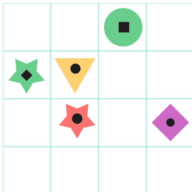
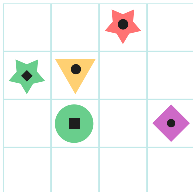
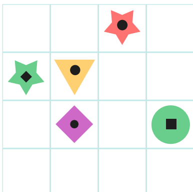


Explanation

In this task, only five of the shapes are swapped around, so we can ignore the shapes not involved.

The initial positions of the five shapes used are shown below:



After the first swap:	
After the second swap:	
After the third swap:	
After the fourth swap:	

BACKGROUND INFORMATION

This task focuses on the *swapping function*. In computer programming, a variable is a memory allocation that can hold information. Swapping involves exchanging the values of any two variables of a compatible datatype.

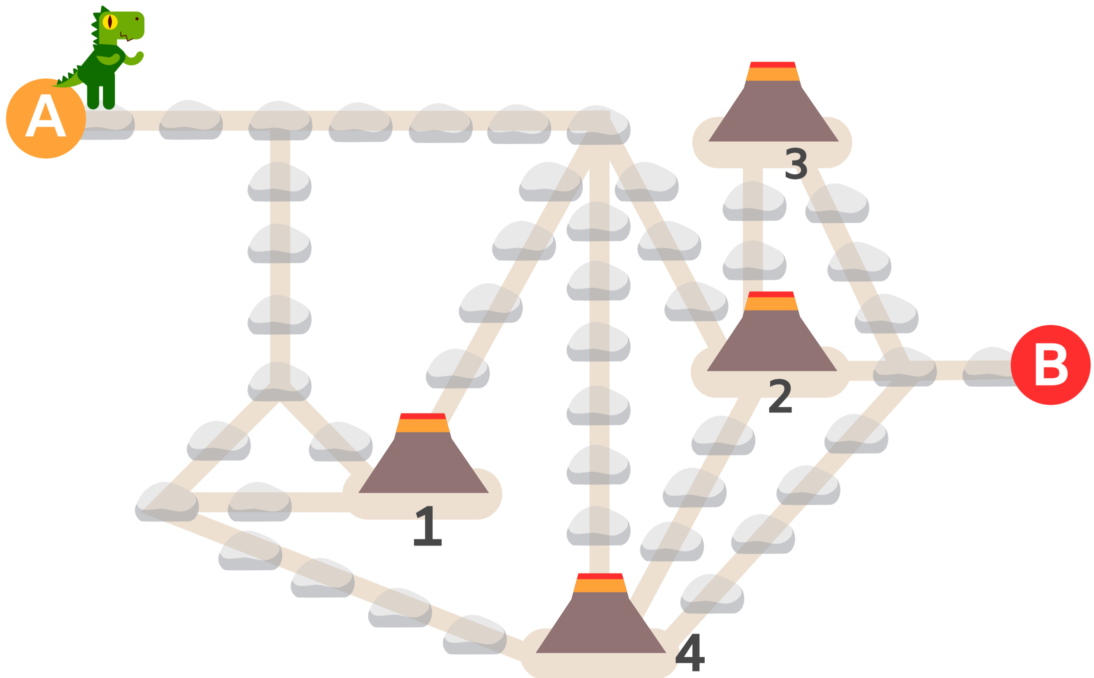
For example, if A is a variable that holds the value “Name” and B is another variable that holds “Date of birth”. After the swap function is performed, variable A will hold “Date of birth” and B will hold “Name”.

These steps of the swap function can be used for sorting a given set of data in any order. For example, ascending or descending order.

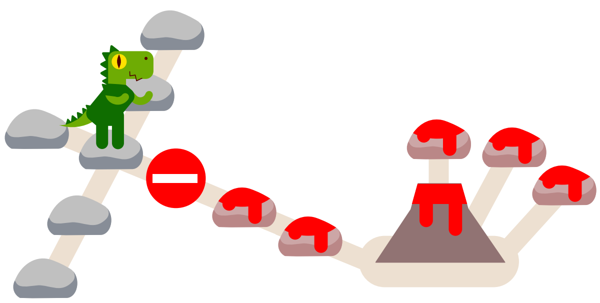
This question comes from
Romania

Volcanoes

Dino wants to get from point A to point B.



For safety reasons, if a volcano erupts, all the roads connected to the volcano are closed up to the next intersection, like in the picture below:



Dino is very worried they won't be able to get to point B if more than one volcano erupts.

Question

Which two volcanoes must erupt at the same time so that Dino can NOT reach their destination?

Volcanoes 1 and 2 | Volcanoes 3 and 2 | Volcanoes 2 and 4 | Volcanoes 1 and 4

Volcanoes – continued

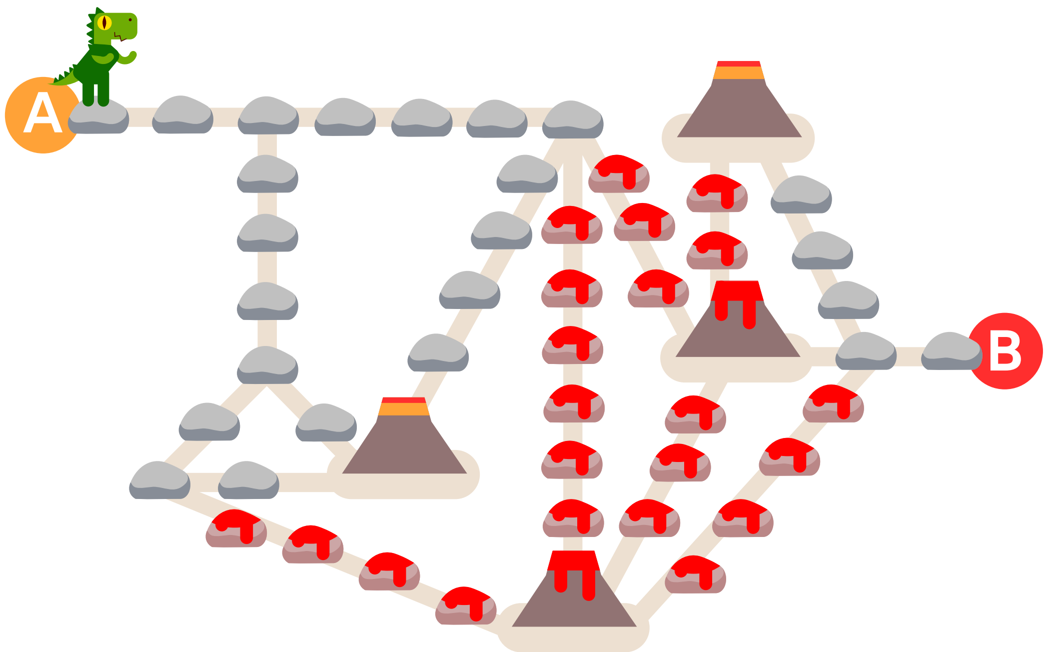
EXPLANATION

Answer

Volcanoes 2 and 4.

Explanation

There are several paths from A to B. However, each path must go through either volcano 2 or 4. If volcanoes 2 and 4 erupt, Dino will no longer be able to reach point B. You can see this in the image.



For the other options, there still exists a path from A to B.

BACKGROUND INFORMATION

A graph is a set of *nodes* (vertices) connected to each other by *edges* (in our case the nodes are volcanoes or intersections), and the edges are represented by the roads between 2 nodes.

In this problem, a subgraph is obtained from the initial graph by removing the selected vertices (erupted volcanoes) and edges that have the nodes at the end. We want to find a subgraph in which the points A and B cannot be connected by a succession of vertices connected by edges after removing two vertices. A vertex is named an *articulation point* (or cut vertex) if removing it and its connected edges disconnects a connected graph. In this problem, all the vertices of volcanoes are not articulation points. Therefore, removing a single vertex of volcano will not disconnect the graph. The above concept can be used to check the fault tolerance of a network system.

Bebras Challenge 2022 Round 2

Years 5+6

This question comes from the Netherlands

Years 3+4
Years 5+6 Easy
Years 7+8
Years 9+10
Years 11+12

Garden Of Eden, Hotel California

In Beaver County there are 10 towns connected by roads. The local council decides to make some of the roads one-way roads, which means beavers can only move along them in one direction.

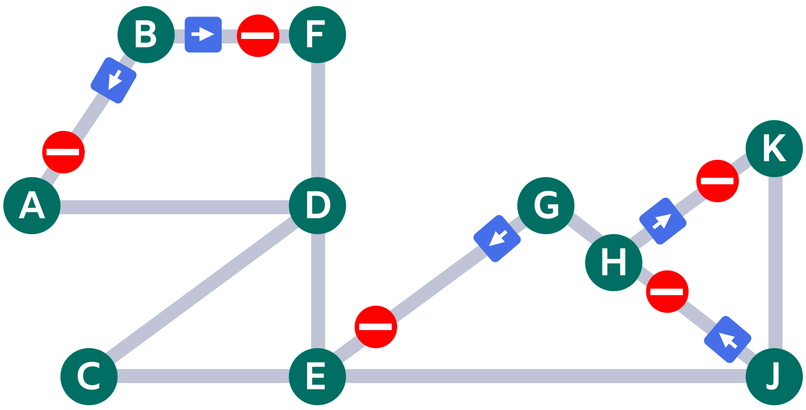
This first design is below, where means a one-way road with the arrow pointing in the direction beavers have to travel.

- If a town can not be entered due to the one-way roads, it is called a “Garden of Eden”.
- If a town can not be left due to the one-way roads, it is called a “Hotel California”.

The council wants to avoid both of these situations.

Question

Which of the towns are a “Garden of Eden” or a “Hotel California”?



EXPLANATION

Answer

B is a “Garden of Eden”.

No town is a Hotel California.

Explanation

In B there are only outward directed one-way roads. All other towns with at least one outward directed one-way road has another road that leads into it (G,H,J).

For a “Hotel California” to exist, the town can only have inward directed one-way roads connected to it. For all towns with at least one inward directed one-way road, there is another road that can lead away from the town - From A, F and E we can go to D, from H we can go to K, and from K we can go to J.

BACKGROUND INFORMATION

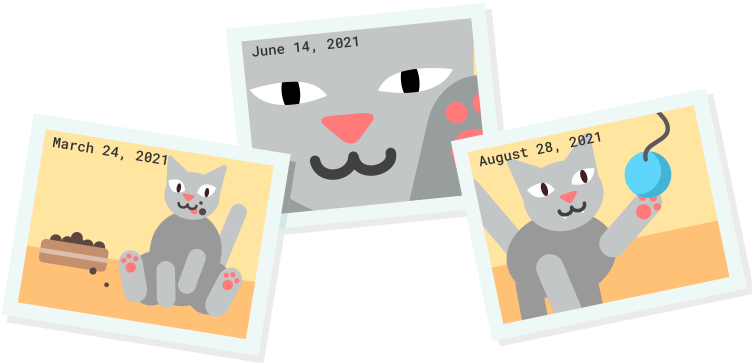
Analysing a network like a map is an important task in many applications. This type of graph using dots and lines and one-way roads is called a *directed graph*. Using graphs can help to analyse the possible routes and to identify possible errors in design, like a “Garden of Eden” or a “Hotel California”.

The flow through a computer program can also be shown using a network or a graph. A software designer can look at the flow of the program using a graph to discover whether some statements can be reached and if they get a proper follow up.

This question comes from Switzerland

Years 3+4
Years 5+6 Easy
Years 7+8
Years 9+10
Years 11+12

Cat Pictures



Aika loves taking pictures of her cat and posting them on Bebragram.

The computer orders pictures by name, from 0 to 9, and from A to Z.

Aika wants her pictures ordered from oldest to newest, so she always adds the date to the name.

Aika tries out some filename patterns for a picture from August 19, 2021.

Question

Which filename pattern should Aika use?

cat_august_19_2021	cat_19_august_2021	cat_19_8_2021	cat_19_08_2021
cat_2021_august_19	cat_2021_19_8	cat_2021_08_19	cat_2021_8_19

EXPLANATION

Answer

Aika should use the pattern G: cat_2021_08_19.

Explanation

Pattern A through D, cat_august_19_2021, cat_19_august_2021, cat_19_8_2021, and cat_19_08_2021 are incorrect.

If we have pictures from different years, then we want all pictures from one year to be next to each other, even if they have different months and days. Therefore, the year should come before the month and day in the name. The reason for this is that then the names of pictures from the same year all have the same beginning part in their name, and so will be ordered close together by the computer.

If either the month or day appear before the year, which happens in the patterns listed above, then two photos taken on the same day and month, but one year apart, will be ordered together. For example, a photo from August 19, 2022, will be ordered next to one from August 19, 2021, which is not what Aika wants.

Cat Pictures – continued

Pattern F, cat_2021_19_8, is also incorrect.

For the same reason, the month should appear before the day: if both the year and month are identical, then the pictures should be grouped close together. For this reason, this pattern will not work, as it would sort a picture from September 19, 2021, in between two pictures from August 18, 2021, and August 20, 2021.

Pattern E, cat_2021_august_19, is also incorrect.

We can also see that the month needs to be written as a number. If it is written as a word, as in this pattern, then the photos from August and April would be close together in the ordering. However, photos from March and May should be close to April, not August.

Pattern H, cat_2021_8_19, is incorrect.

The main difference between patterns G and H is that the month is written as "08" in pattern G and as "8" in pattern H. For the months from January through September, both patterns work fine. But for the months October, November, and December, which have two digits when written as a number, this will cause a problem with pattern H.

Take for example photos from August 19, 2021, and December 19, 2021. In pattern H, these dates will be written as 2021_8_19 and 2021_12_19. We want 2021_8_19 to be ordered before 2021_12_19, but the computer will order 2021_12_19 first because alphabetically the '1' from '12' comes before '8'.

BACKGROUND INFORMATION

In computer science, ordering data is called *sorting*. Sorting data is a very common task; computer science tells us how sorting can be done efficiently.

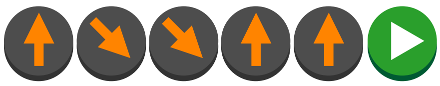
Sorting is so important because it is used as part of many other algorithms. So the efficiency of our sorting algorithms affects the efficiency of other algorithms. The choice of sorting algorithm depends on the data set that we wish to sort. The simplest sorting algorithms that you may know about already are: bubble sort, insertion sort, and selection sort.

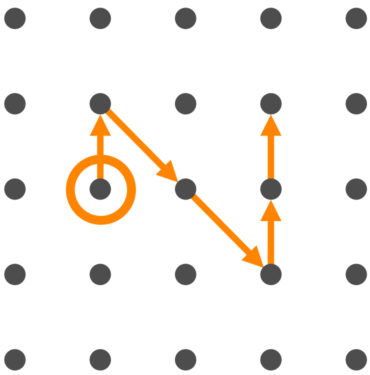
Representation of dates in files and computer memory is an important issue in computer technology. Before the year 2000 many computer systems used only two digits for the year, e.g. 81 to mean 1981. Now, most dates in computer systems follow the pattern YYYYMMDD. Here, YYYY stands for four digits for the year, MM is for the two digits for the month, and DD is for two digits for the day. This pattern will work fine for computer systems until about 8,000 years from now, in the year, 10000.

Between Dots

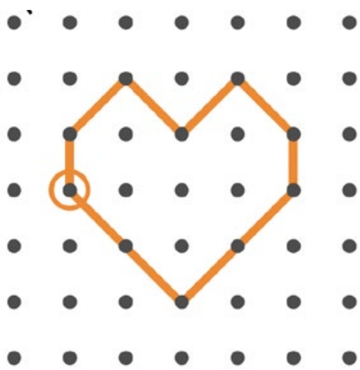
Emma is playing with a robot that draws lines between dots. She pushes arrow buttons to send the robot to the next dot.

The robot starts on the dot with the surrounding circle.

This arrow sequence  tells the robot to draw:



Question



Which of the following commands can draw the shape shown left?

Continued on next page

Between Dots – continued

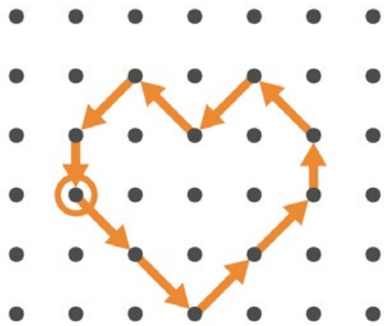
EXPLANATION

Answer

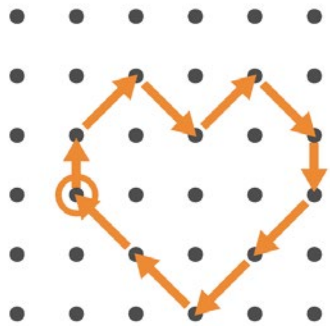
OR

Explanation

The buttons can be pushed in two types of sequences to draw this image:



or



BACKGROUND INFORMATION

Computer programs are a series of commands that tell a device what to do. All computer programs are run (activated) by pressing a button or typing a command.

One of the basic things in informatics and robotics for students to learn is that a robot or a computer follows commands in a sequence / specific order and executes actions in that order. Students will learn that a computer or robot will only respond to commands it has been given. The commands must be precise.

Students use algorithmic thinking to follow the sequence of button commands in the question. They will also look at the data representations through the drawing of the correct image representation by pressing buttons in the correct sequence. Using recipes as a metaphor might help students to understand this concept better. The steps of cooking are like instructions, and only correct instructions can make delicious meals.

It is important to understand the instructions and execute them step by step to see what is happening. These steps will be useful for *debugging*. Debugging is the process of working through a program sequence to see where errors might have occurred. In this case, students can press the reset button to try their solution if the first one doesn't work. It is useful to have students 'think-aloud' (talk through actions as they are making them), especially when debugging. Discussing where they might have gone wrong is a useful process for students to learn that we all make mistakes and need to follow steps to fix them. Even the best computer programmers make mistakes. Debugging is a natural part of the process of learning.

This question comes from
Taiwan

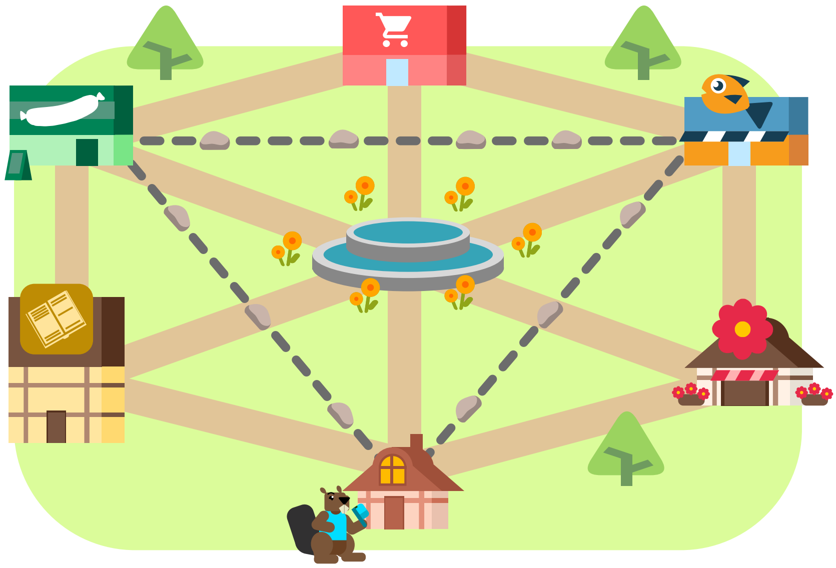
Years 3+4
Years 5+6 Easy
Years 7+8
Years 9+10
Years 11+12

Grocery Shopping

Below is a map of the village where the Beaver family lives. Ordered clockwise/to the left of the Beavers' home is the bookstore, meat shop, supermarket, seafood shop, and finally the flowershop.

To go from one building to another, the Beavers walk along either a muddy path or a rocky path. Walking between two buildings takes Little Beaver 5 minutes by muddy path and 8 minutes by rocky path.

For example, it takes Little Beaver 5 minutes to go home from the flower shop or the bookstore, and 8 minutes to go home from the seafood shop or the meat shop.



Shopping List

- Fish / Seafood Shop
- Chicken / Meat Shop
- Eggs / Supermarket
- Flowers / Flowershop

Mother Beaver asks Little Beaver to help with the grocery shopping. The shopping list is shown above. Little Beaver needs to start the shopping trip from their home, finish all the shopping and then come back home - however, Little Beaver can do the shopping in any order.

Question

What is the minimum walking time Little Beaver needs?

18 minutes | **20 minutes** | **28 minutes** | **30 minutes**

EXPLANATION

Answer

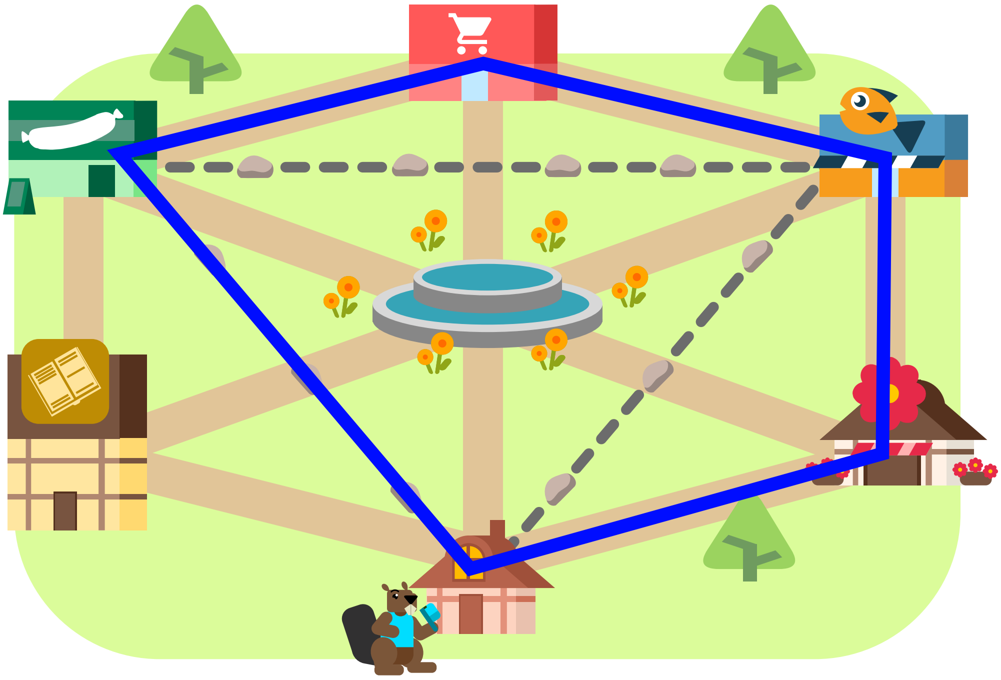
The correct answer is 28 minutes.

Continued on next page

Grocery Shopping – continued

Explanation

Let's assume you are Little Beaver. Since you have to start from your home, finish buying things from 4 shops, and come back home, you will have to walk through at least 5 segments of paths. Each path takes at least 5 minutes, so you need at least 5 * 5= 25 minutes to complete the task. Therefore, options 18 minutes and 20 minutes are wrong.



To get to the meat shop or the seafood shop, it will take you 10 minutes (through the bookstore or the flower shop, respectively), or 8 minutes (directly). Since you also need to go to the supermarket and the flower shop, the most efficient way to go to the seafood shop is not directly through the rocky path.

Similarly, since you don't have to go to the bookstore, it is more efficient to take the rocky path between the meat shop and your home. Therefore, the meat shop should be your first or last stop.

It takes 5 * 4 = 20 minutes to walk from shop to shop, so the total walking time is at least 8 + 20 = 28 minutes. The selected path is shown above.

BACKGROUND INFORMATION

In this task, we can find the shortest route by drawing it out. However, in real life, when there are thousands of paths to choose from instead of 15, we will need help from technology, such as GPS and routing software, to find the shortest path.

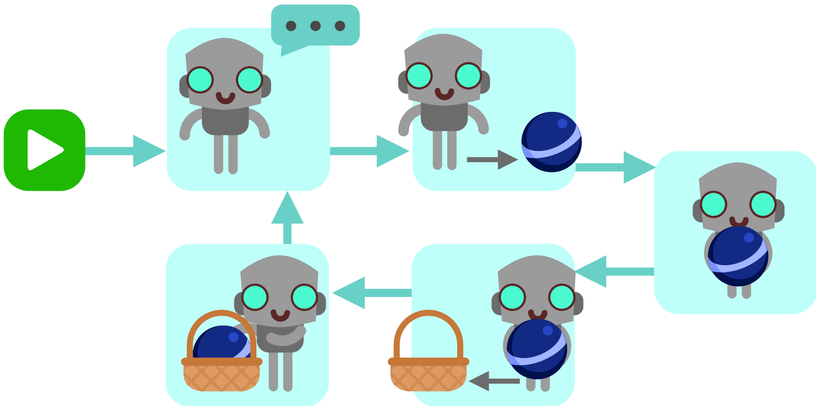
In computer science, graphs are a common way to show relationships between data. Graphs can be used to represent links between objects, it is a method of representing the connections between things with vertices and edges. Graphs also make it easier to describe the relationships (often represented by edges) amongst key points (often represented by vertices) of complex concepts.

In graph theory, the shortest path problem aims to find the shortest path (edge) between two points (vertices). These paths may have different priorities (weights). To account for these priorities (weights), we have to multiply each distance by its weight. This is called a weighted distance. In this case, the solution to the shortest path problem is finding the smallest weighted distance.

Robot

Ram is a tennis player. They programmed a robot to pick up tennis balls that end up scattered on the court, and put them into a basket at the back of the court.

The flowchart of this program is shown below, starting with the green ‘play’ symbol:



Here are the programmed commands in clockwise motion, following the arrows:

1. Wait for a ball on the floor
2. Go to the ball
3. Pick up the ball
4. Go to the basket
5. Put the ball into the basket

Ram throws a ball on the court to test the robot. Unfortunately, after some actions, the robot stays in the middle of the court where the ball was, and is holding the ball but not moving.

Question

What is the first step that the robot failed to perform?

- “Wait for a ball on the floor!”
- “Go to the ball!”
- “Pick up the ball!”
- “Go to the basket!”
- “Put the ball into the basket!”

EXPLANATION

Answer

The robot failed to perform the next step: go to the basket.

Explanation

Lets understand the observations made by Ram. Ram observed that the robot is in the middle of the court, holding the ball.

This means the robot succeeded in waiting for a ball on the floor, then managed to go to the ball, and finally managed to pick up the ball. However, the robot stays in the middle of the court where the ball was, and is still holding the ball, so it failed to perform the next step: go to the basket.

BACKGROUND INFORMATION

This task focuses on *logical reasoning* and *debugging*. Debugging is a process of fixing a bug or removing errors in the program. This act of debugging begins after the program fails to execute properly. As a developer/programmer, one must be able to identify the part of code that needs improvement.

Once the incorrect patch of the program is identified it becomes convenient for the programmer to change the code and apply the patch to the main program. The main program is then validated.

Animal Sorting

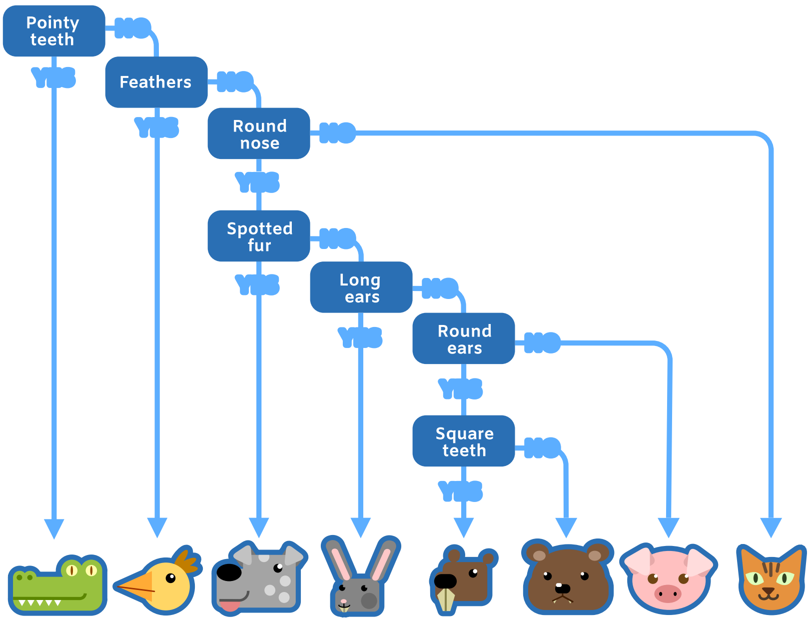
The animal species in Beavertown can be distinguished by their features.

To identify an animal, citizens of Beavertown use a decision tree. When the answer to a question is "YES" they go *down* and when the answer to a question is "NO" they go *right*.

The decision tree below can tell the difference between eight animal species. Some animals can be identified with just one question, others need seven questions before they are identified.

Question

Click on all the animals that need *more* than three questions before the citizens of Beavertown can identify them with this decision tree.



EXPLANATION

Answer

Animal species 3 (dog), 4 (rabbit), 5 (beaver), 6 (bear) and 7 (pig) need more than 3 questions before we can identify them.

Explanation

Animal species 1 (crocodile), 2 (bird) and 8 (cat) will be identified with the first three questions.

BACKGROUND INFORMATION

In informatics *decision trees* are a useful way to display information. It uses a *tree-like model* (with roots and branches) to display decisions and consequences. Decision trees are also often used in *Artificial Intelligence*. An AI-machine can use a decision tree to filter through facts and past incidences to predict a likely future outcome, event or action or for identification, example diagnostic in medicine.

In this task we used a decision tree to display information about a set of animals. The decisions are the questions about the animals, and the final consequence is the animal that you end up with.

A decision tree can be compared with what computer programmers would call an 'IF-THEN-ELSE'-statement. This instruction is often nested and may contain logical connectors. Sometimes people use flow charts for these statements as well.

This question comes from Ireland

Years 3+4
Years 5+6 Medium
Years 7+8
Years 9+10
Years 11+12

Presents Program

A group of friends give presents to one another according to these rules:

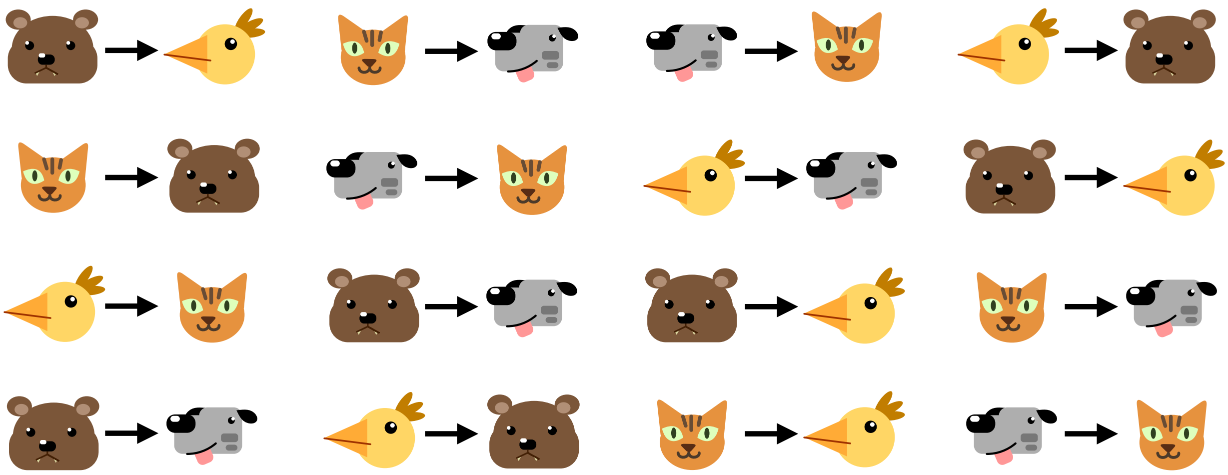
- Rule 1: No friend can give a present to themselves.
- Rule 2: Each friend must give one present.
- Rule 3: Each friend must receive one present.

For any size group of friends, who gives a present to whom is decided according to these rules.

For example, a group of **two friends** would look like:

Question

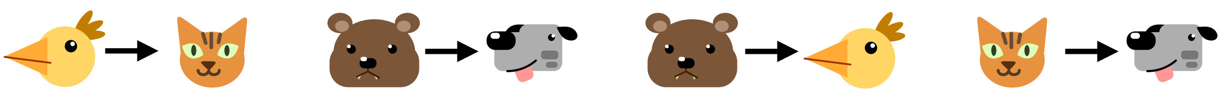
For a group of four friends, you are given the following options.
One of the following figures does not follow the rules! Which one?



EXPLANATION

Answer

The correct answer is the one started from Chiki Chi giving a present to Cathy Cat.



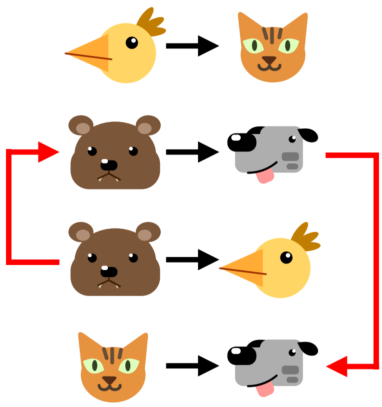
Explanation

Chiki Chi giving a present to Cathy Cat has mistakes because it says that Beary Bear should give two presents and that Don Dog should receive two presents. According to the first rule each friend is only supposed to give one present and according to the second rule each friend is only suppposed to receive one present.

Other answers conform to all three rules: each friend gives and receives one present and no friend gives a present to themselves.

Continued on next page

Presents Program – continued



In the first one, Beary Bear exchanges present with Chiki Chi and Cathy Cat exchanges presents with Don Dog.

In the second one, Cathy Cat should give a present to Beary Bear; Beary Bear should give a present to Chiki Chi; Chiki Chi should give a present to Don Dog; Don Dog should give a present to Cathy Cat.

In the last one, Beary Bear should give a present to Don Dog; Don Dog should give a present to Cathy Cat; Cathy Cat should give a present to Chiki Chi; and Chiki Chi should give a present to Beary Bear.

BACKGROUND INFORMATION

This task can be regarded as a pattern recognition task. Each answer is a correct representation of a pattern with specific properties: each blue arrow joins *two different friends* (Rule #1) and each friend appears exactly once on each side of a blue arrow (Rule #2 and Rule #3).

If each answer is considered as an output of a program for distributing presents in a group of friends that must comply with all three rules, then the process of tracing and examining each output is part of the program (software) testing. The conformity of each answer is checked against the rules. The objective of software testing is to find errors.

Being sure that a computer program will run correctly is a big problem in computer science. Computer scientists have devised techniques to prove that some computer programs are correct, but computer scientists also know that no one set of techniques will work for all programs (this comes from computer theory, where we know that any nontrivial property of a computer program is undecidable). So, we often have to run programs many times to increase our confidence that they are correct. This is called **software testing**.

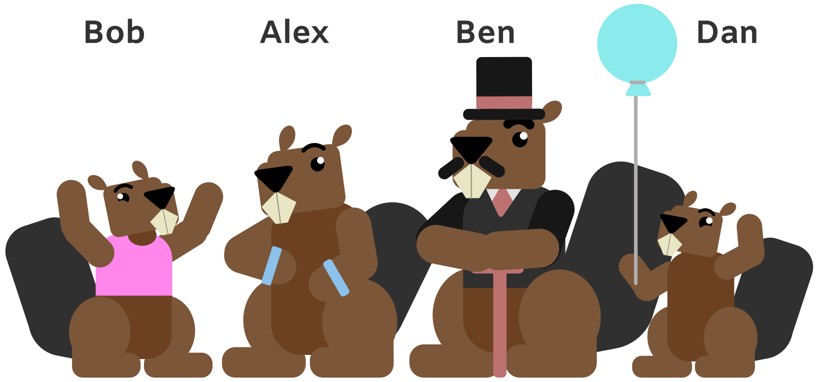
Software testing is a big part of computer science. There are many specialised automated techniques that can test all parts of a computer program, and all professional software is automatically re-tested each time the developers change it. However, it still requires considerable expertise to design tests. Two things you have to do to solve this task are in common with software testing: you have to analyse what the goals of the program are and look for specific features in the program's output to tell if it is working correctly or not. While software testing can be very useful, unfortunately it doesn't find all the errors in programs all the time. Computer programs are just very difficult to get right, especially as they get larger and larger.

Fifo Restaurant

A busy beaver restaurant uses a waiting list - when a customer arrives they write their name on the list. Normally, the chef takes the order of the customer who arrived first on the waiting list. However, the oldest beaver is given priority even if they arrive after a younger beaver.

Four beavers want to order food. Ben is the oldest beaver, followed by Alex, then Bob, and finally Dan is the youngest.

Bob and Alex arrive first and write their names first on the waiting list. Then Ben arrives. The last beaver to arrive is Dan. The chef then comes out to serve the four beavers.



Question

Based on the rules followed at the restaurant, in what order will the host serve the customers?

- Bob, Alex, Ben, Dan
- Dan, Bob, Alex, Ben
- Alex, Ben, Bob, Dan
- Ben, Alex, Bob, Dan

EXPLANATION

Answer

The correct answer is Ben, Alex, Bob, Dan.

Explanation

Beavers in the waiting list are ordered by age when they arrive in the waiting list. In fact, as long as the four beavers are served together, it doesn't matter which order that they arrive.

State-1: When Bob and Alex are in the waiting list, Alex has the highest priority because he is older than Bob.

Alex	Bob
	
1st priority	2nd priority

Continued on next page

Fifo Restaurant – continued

State-2: When Ben arrives, Ben will have the highest priority because he is older than Alex and Bob.

Ben	Alex	Bob
		
1st priority	2nd priority	3rd priority

State-3: When Dan arrives, she will be put as the last beaver to be served in the waiting list, because she is the youngest.

Ben	Alex	Bob	Dan
			
1st priority	2nd priority	3rd priority	4th priority

So the order the customers are served in is (Ben, Alex, Bob, Dan).

Answer A (Bob, Alex, Ben, Dan) is not correct, because the order of service is First Come, First Served, which means it is purely according to the beaver's arrival. However, this is not how the restaurant chooses to serve their beavers.

Answer B (Dan, Bob, Alex, Ben) is not correct because the order of service is Last Come First Served, which also is not how the restaurant has chosen to serve their beavers.

The answer C (Alex, Ben, Bob, Dan) is not correct because it does not consider who comes first, nor the age of the beaver. The waiting list is ordered alphabetically by the beavers' names.

Please note that there are no more customers after Dan, so the order of service is only for Alex, Ben, Bob, Dan.

BACKGROUND INFORMATION

In computer science, there is a data structure called *priority queue*. The data structure representing the waiting list of the beaver restaurant is called priority queue.

In an ordinary queue, all elements are managed as first-come-first-served or FIFO (First In First Out). Whilst in a priority queue, an element with certain properties (priority) will be served first. In this task, the priority is the age of the customer (the oldest one). The host of the restaurant manages all customers on the waiting list as a queue, but she gives the priority to the oldest beaver even if the oldest beaver arrives late.

This kind of queue gives each element priority to execute. The element with the highest priority is executed before a lower priority element. If two elements have the same priority, they will be executed according to the order of which they entered the queue.

Each time a service has finished, the host takes the top priority element in the list (dequeue). During a service, new customers can come and join the queue (enqueue). The host should maintain the time of arrival and the priority (in this task, the age of the beaver) of each element.

This question comes from the Philippines

Years 3+4
Years 5+6 Medium
Years 7+8
Years 9+10
Years 11+12

Beaverly's Food Bags

Beaverly's company makes food bags containing acorns and mushrooms. Each food bag has 8 treats, with the last one added by Beaverly herself following this rule:

- If there is an even number of , then she adds a mushroom .
- If there is an odd number of , then she adds another acorn 

For example, if the first 7 treats are , then Beaverly adds a mushroom to complete the food bag: 

But, one evening, a naughty rodent sneaks into the company's storehouse and leaves this note:

"I will swap an , from one of your food bags with a , from another food bag."

The next morning, Beaverly quickly suspects these unusual-looking food bags:

- Food Bag 1: 
- Food Bag 2: 
- Food Bag 3: 
- Food Bag 4: 

Question

Help Beaverly identify the two food bags affected by the naughty rodent's swapping.

Food Bags 1 and 3

Food Bags 2 and 3

Food Bags 1 and 4

Food Bags 2 and 4

Continued on next page

Beaverly's Food Bags – cont'd

EXPLANATION

Answer

The correct answer is Food Bags 2 and 3.

Explanation

To find the solution, we can think about the effect of the last treat added by Beaverly if the rules are followed:

- If there is an even number of acorns among the first 7 treats, her addition of a mushroom does not change the number of acorns. It also does not change the fact that there is an even number of acorns inside the food bag.
- If there is an odd number of acorns among the first 7 treats, her addition of another acorn increases the number of acorns by 1. Thus, the number of acorns now becomes even.

The key insight here is that *every food bag should have an even number of acorns* after Beaverly's addition. If the number of acorns is odd, then the food bag is certainly affected by the naughty rodent's swapping:

- Food Bag 1 has 4 acorns.
- Food Bag 2 has 3 acorns.
- Food Bag 3 has 7 acorns.
- Food Bag 4 has 2 acorns.

Since the problem states that only two food bags are affected, the correct answer is (B), corresponding to Food Bags 2 and 3.

BACKGROUND INFORMATION

In computer science, particularly in the fields of information and coding theories, this task is known as *error detection*, which is crucial since data unavoidably become corrupted as they are transmitted across communication channels and networks. Notice that replacing all the acorns with 1's and all the mushrooms with 0's results in bits (binary digits) of data.

One of the simplest error detection schemes is *parity checking*. Parity refers to whether a number is odd or even. There are two variants of this technique: *even parity* and *odd parity*. In both modes, an extra bit (either a 1 or a 0) — referred to as the *check bit* — is appended to the data. In the case of even parity, the check bit is set to 0 if there is already an even number of 1's; otherwise, it is set to 1. The opposite happens in the case of odd parity. In this problem, the last treat added by Beaverly acts as an even parity check bit.

Consequently, if even parity is employed and the stream of data (together with the check bit) contains an odd number of 1's, then an error flag is raised. For instance, sending 1000101 requires that an even parity bit of 1 is appended: 10001011. If, for some reason, the second bit becomes corrupted during transmission, transforming the data to 11001011, then the recipient is able to recognize the presence of an error since 11001011 has an odd number of 1's — a violation of the parity rule.

Parity checking only works if the number of corrupted bits are odd. This is the reason why this task specifies that the naughty rodent swaps only a *single* acorn from one bag with a *single* mushroom from another. Although this error detection technique does NOT tell us which bits are corrupted (that is, which treats are swapped) nor does it provide us with any way to repair the data, it is still widely used in some buses found inside the system unit (for example, PCI buses) for its efficiency.

Cupcakes

Bebras Bakery produces cupcakes for the hard-working hungry beavers in the town. Each cupcake is decorated with three sweet layers. Firstly, each cupcake gets an icing layer, then a toppings layer, and finally a fruit layer. Each of the layers is changed from one cupcake to the next.

The *icing layer* changes with the following pattern: green ■ → white ★ → red ● → blue ☽ →[repeats again starting with green ■]

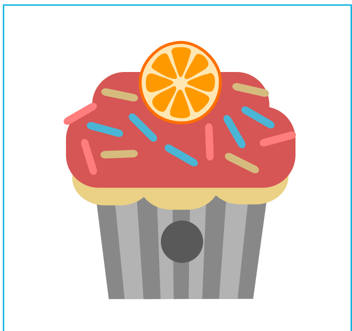
The *toppings layer* changes with the following pattern: sprinkles → chocolate flakes → toasted nuts → [repeats again starting with sprinkles]

The *fruit layer* changes with the following pattern: blueberry → cherry → kiwi → strawberry → orange [repeats again starting with blueberry]

The picture above shows cupcakes on the production line at some point during the day, as they move from left to right.

Question

What will the cupcake marked with “X” look like?




Continued on next page

Cupcakes – continued

EXPLANATION

Answer

The correct answer is *Blue, sprinkles, orange*.

Explanation

1. *Red-sprinkles-orange* - is not correct because cupcake “X” cannot have red icing.
2. *White-chocolate flakes-kiwi* - is not correct because cupcake “X” cannot have white icing.
3. *Blue-toasted nuts-strawberry* - is not correct because toasted nuts and strawberry would come immediately before cupcake “X”.

Blue, sprinkles, orange is correct because:

- After white icing comes red and then blue, so cupcake “X” will have blue icing.
- After sprinkles comes chocolate flakes, then toasted nuts, then sprinkles, so cupcake “X” will have sprinkles.
- After blueberry comes cherry, then kiwi, then strawberry, then orange, so cupcake “X” will have orange.

BACKGROUND INFORMATION

This task illustrates the computational thinking concepts of algorithms and pattern recognition, and the computer programming concept of remembering state.

Pattern recognition is the concept of finding patterns in the problem that will allow reuse in the solution, either in the form of loops in the solution, or reusing parts of solutions from previously solved problems. In this task, the sequence of options for each layer forms a pattern, but also there is a pattern in the way that the application of each layer (icing, toppings, fruit) follows the same fundamental algorithm.

An algorithm is a list of instructions. Following instructions is a very important concept in computer science. This is how a computer works - we tell it what to do, and it follows these steps. For some programming languages, the order of instructions is very important also. By changing the order, we can change the output of the program. The sequence of ingredients in this task is very important for each layer.

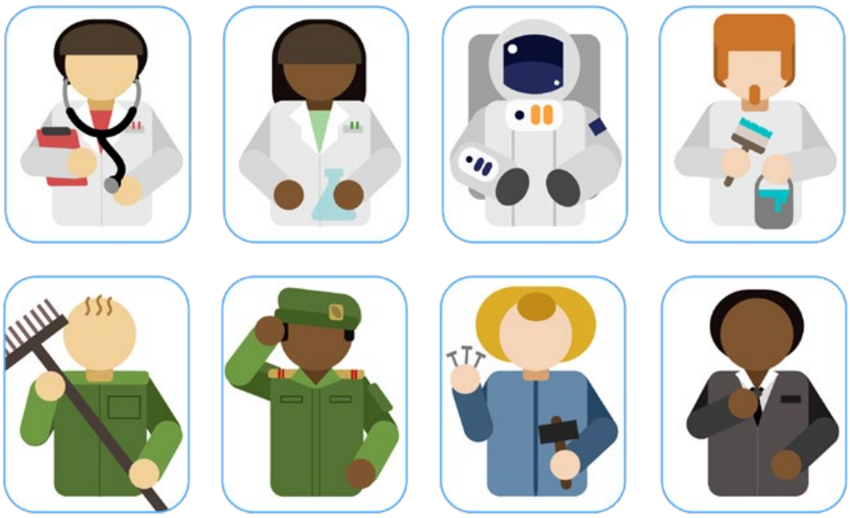
Remembering state is an essential part of many programming languages. A computer program can store information, and affect its future behaviour, by changing its state. The most common way to remember state in a computer program is to store a particular value in a variable.

This question comes from
South Korea

Guess Who?

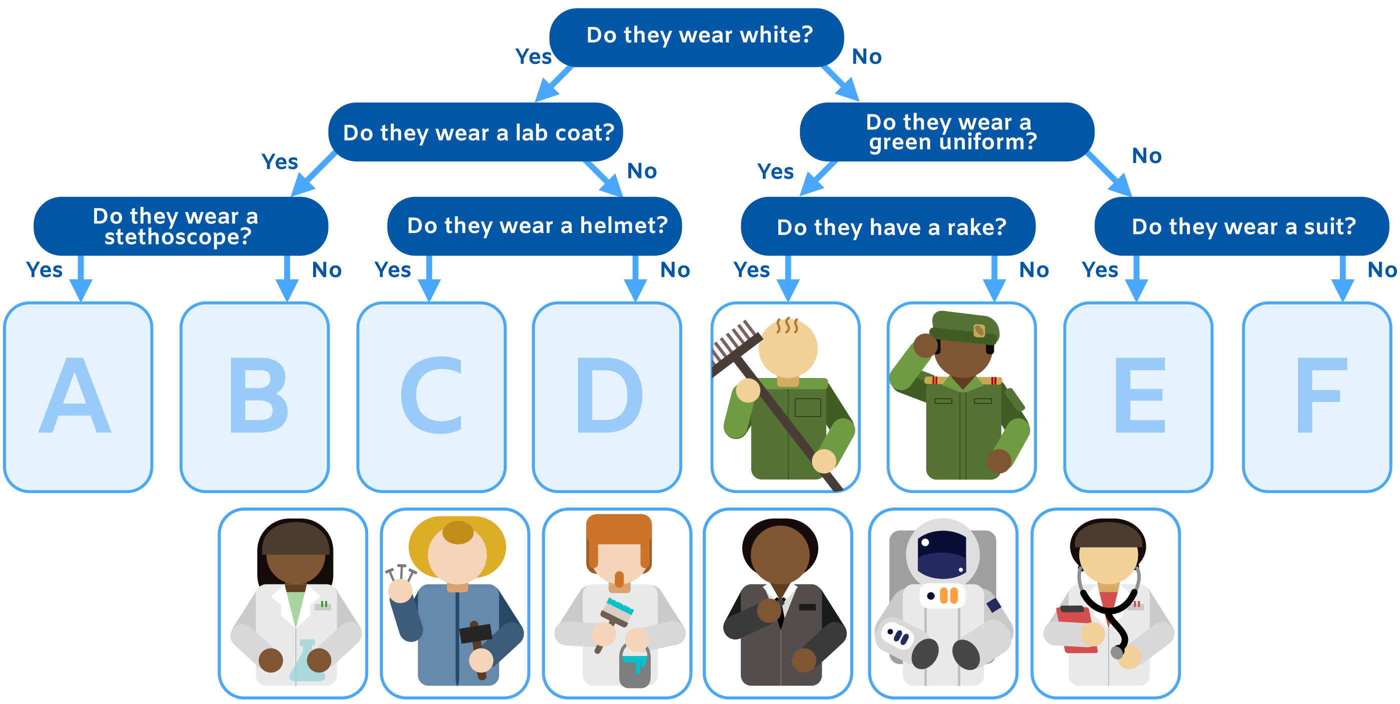
Jack and Jill are playing a quiz game that they made up. Both of them have the same 8 cards with a picture of a person and their profession on them. Jack hides one of his cards and Jill must guess the profession on the hidden card through several questions. Jill only asks one question at a time and Jack can only answer with ‘yes’ or ‘no’.

These are Jack and Jill’s cards (left to right, top to bottom): doctor, scientist, astronaut, painter, gardener, soldier, carpenter, business person.



Question

Jill tries to find the hidden card by creating a diagram like the one below.



Continued on next page

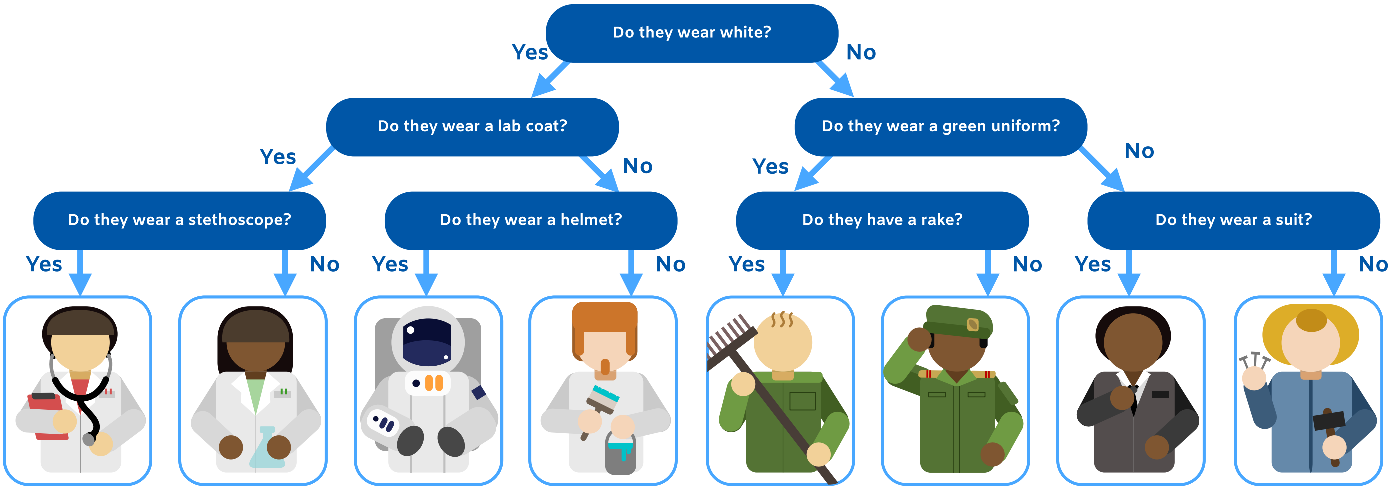
Guess Who? – continued

EXPLANATION

Answer

The correct answer is doctor, scientist, astronaut, painter, business person, carpenter.

Explanation



Jill could find the hidden card after three questions using her own diagram.

- The profession that wears white, a lab coat and a stethoscope is the doctor.
- The profession that wears white, a lab coat but no stethoscope is the scientist.
- The profession that wears white, no lab coat and a helmet is the astronaut.
- The profession that wears white, no lab coat and no helmet is the painter.
- The profession that does not wear white, does not wear a green uniform but wears a suit is the business person.
- The profession that does not wear white, does not wear a green uniform and does not wear a suit is the carpenter.

BACKGROUND INFORMATION

This task demonstrates a tool called *decision trees*. Here it is being used for classification. For each step, there is a classification based on one attribute. In the task, each feature takes one of two values (either it is present or not).

The decision tree in this task is called a *binary tree*. The number of possible options after the next question is reduced by half. So, to determine one option out of a million, 20 questions would be enough, since 2 ^ 20 = 1048576> 1,000,000.

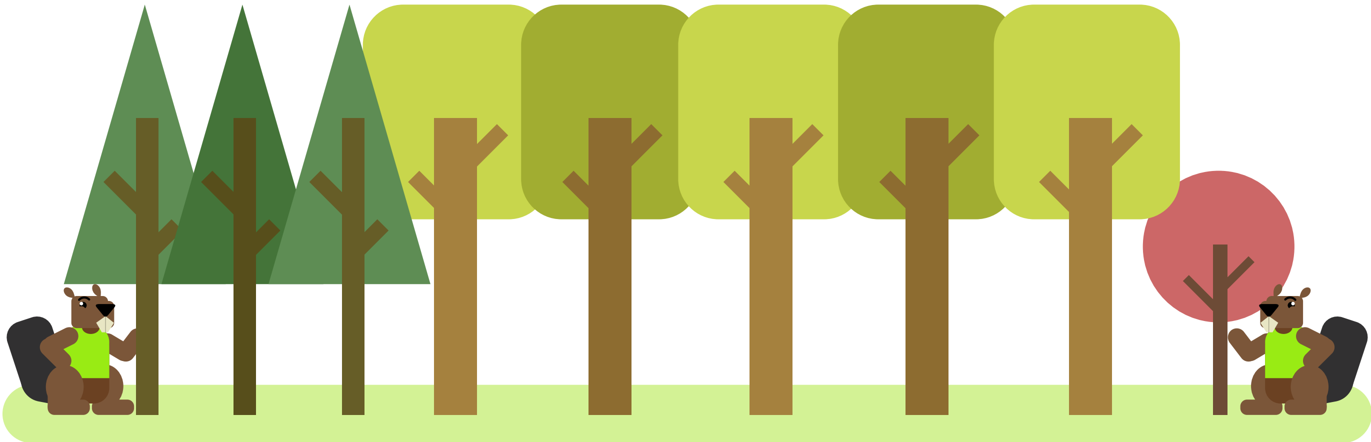
Decision trees are used in some simple automatic telephone robots that ask questions and offer to press a number as an answer (interactive voice response systems). Decision trees are also a popular tool in machine learning.

Three Beavers

Three beavers are cutting down trees. Each beaver works alone. Each tree has a different thickness and a different amount of time to cut them down.

Number of trees of the same thickness	How many hours is needed for cutting one tree
5	4
3	3
1	1

The beavers can cut down the trees in any order they want. But they have to finish the current tree they are working on, before they start cutting down the next one. Also, two beavers can't work on the same tree.



Question

What is the shortest amount of hours needed for the beavers to cut down all the trees?

EXPLANATION

Answer

Correct answer is 11 hours.

Continued on next page

Three Beavers – continued

Explanation

It might appear that the result would be 10 hours.

$$\frac{(5 \times 4 + 3 \times 3 + 1 \times 1)}{3} = 10$$

However, you cannot group the trees into three equal groups that allow each beaver to spend ten hours cutting the logs.

As seen in the diagram, the three beavers cannot spend the same time. Two beavers are cutting trees for 11 hours while the third only uses eight hours cutting trees.

We can see it in the next scheme:

Beaver 1: 4 hours + 4 hours + 3 hours = 11 hours

Beaver 2: 4 hours + 4 hours + 1 hour = 9 hours

Beaver 3: 4 hours + 3 hours + 3 hours = 10 hours

In this scheme, when a beaver finishes cutting a tree, it proceeds to cut the largest tree available.

The three beavers start cutting large trees (4 hours of work) when they finish, simultaneously, the first two start with the last two large trees and the third beaver goes on to cut a medium tree. Since the third beaver finishes earlier, he proceeds to cut a second medium tree. The first two beavers finish later, simultaneously and choose, one the last medium tree and the other the only small tree. The first beaver spends 11 hours at work, the second nine and the third ten.

In this case, the strategy followed has been to choose the largest tree available to continue the work.

BACKGROUND INFORMATION

A way of finding the schedule that yields the shortest time is to try all possible allowed schedules. However, this is impractical in real-world situations; it might require a lot of computer resources. It has been proven that that the scheduling problem is “NP-complete”. That means that it is probably not possible to develop a computer program that can in each case find the optimal solution in acceptable time.

However, for many problem instances, the beavers’ “greedy” strategy can be good enough, and has the advantage that it is very quick to make a schedule and get the beavers working straight away. “Greedy” means that the beavers start with the thickest trees (which require the longest cutting time) than cut the second thickest trees and so on.

Carefully finding a problem instance to cause a strategy to perform poorly is a difficult, yet powerful, technique to find the worst-case running time of a computer program as shown in this question. This skill is called algorithm analysis and is used the field of computational complexity theory.

This question comes from
South Africa

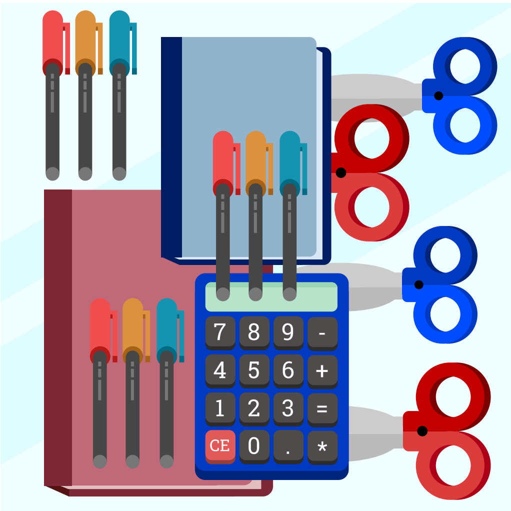
Years 3+4
Years 5+6 Hard
Years 7+8
Years 9+10
Years 11+12

Glass Cabinet

Grandpa Beaver keeps his things neatly stored in a glass cabinet with one glass shelf for each kind of object: books, scissors, pens and calculators.

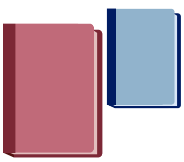
Unfortunately, the young beavers have bumped into the glass cabinet and all of grandpa's things have fallen out.

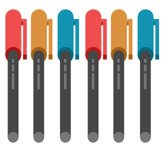
The young beavers want to put grandpa's things back as they were. Unfortunately they don't remember the order of the shelves, but they realise they can figure out the order by looking at the way the objects are arranged after they fell, as shown below:

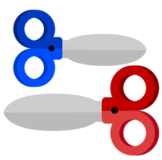


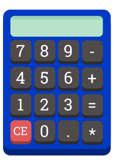
Question

Put grandpa's things in the correct order, from top shelf to bottom shelf, by dragging the items into place.









Continued on next page

Glass Cabinet – continued

Answer

(From top to bottom) Pens, Calculators, Books, Scissors.

Explanation

The approach to solving this question is deducing that the objects on the bottom shelf would've landed first, and the objects on the top shelf would've landed last. One can imagine that the layers are removed one by one from the top.

Top shelf: Pens – nothing obscures the view of the pens.

Second shelf: Calculators – once the pens are removed, nothing obscures the calculators.

Third shelf: Books – once the calculators are removed, the books are in plain view.

Bottom shelf: Scissors – These can clearly be seen to be under the books, with nothing under them.

BACKGROUND INFORMATION

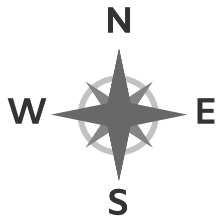
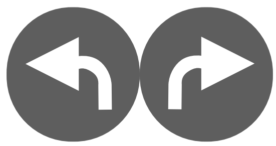
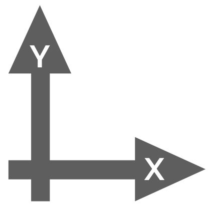
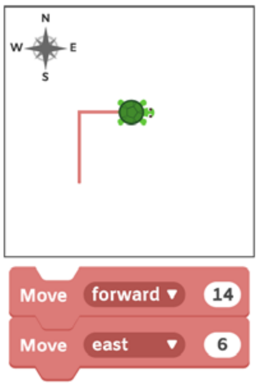
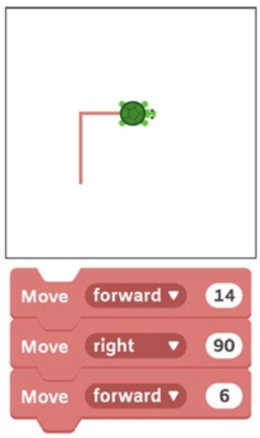
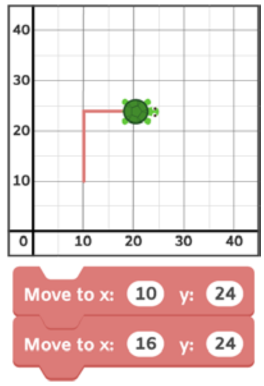
In informatics, the order of things is very important.

There is an underlying data structure of a stack here: the top-most elements hide other elements underneath them. For example, the pens are at the top of the stack, with the calculators under that top element, and so on. Only the top-most elements can be viewed/removed from a stack. Stacks are used in algorithms to manage subroutines: when a program calls a subroutine, we would like to return to the "current state", so that is pushed onto the stack.

Robot Drawing

A school has a turtle robot that has a pen that draws lines when the robot moves.

The robot's movement can be programmed in different ways using three different programming systems:

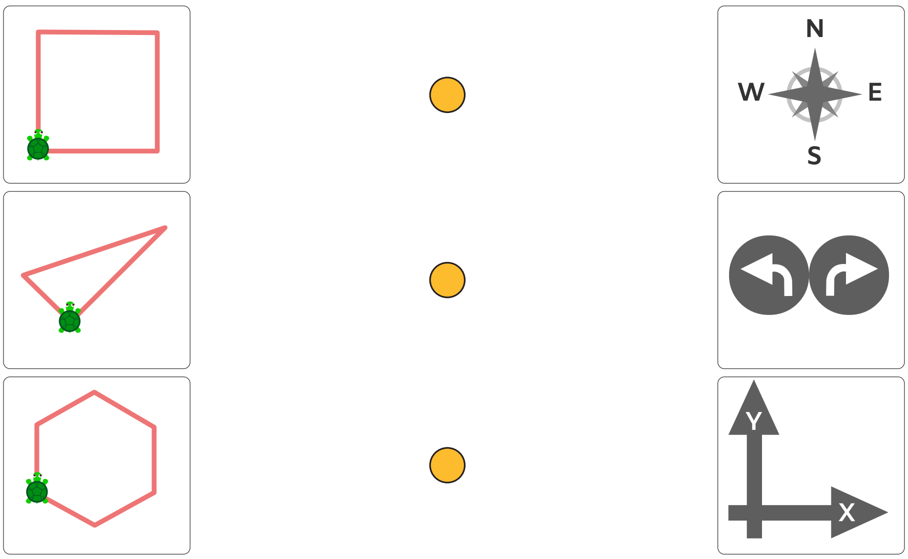
<table>
<tr><th>North, South, East, or West</th><th>Turn Right or Left</th><th>Move to (x, y) coordinates</th></tr>
<tr><td>
</td><td></td><td>
</td></tr>
<tr><td>
</td><td>
</td><td>
</td></tr>
</table>

The three sets of programming commands above make the robot draw the same picture.

The robot can only be programmed using whole numbers without decimals.

Question

Connect the pictures below to the correct programming system so that the robot can be programmed to draw the three pictures using as few "**move to (x, y) coordinates**" programming commands as possible. Each programming system can only be used to draw one picture. Connect points using 3 lines.

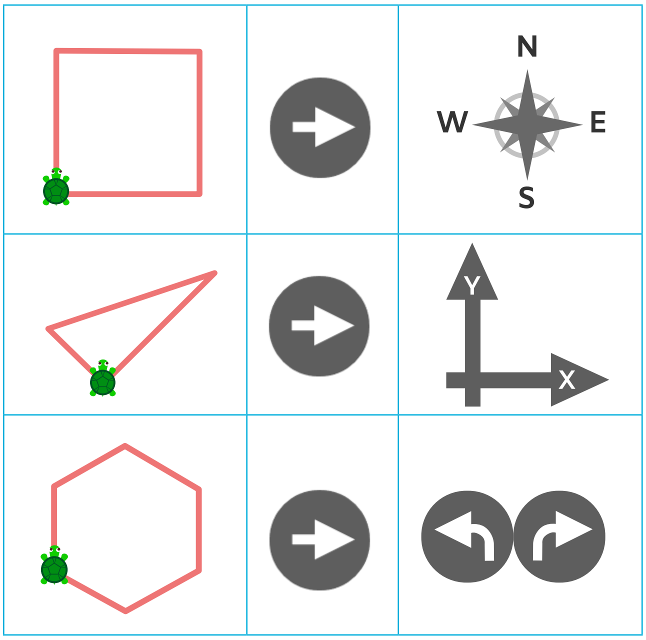


Continued on next page

Robot Drawing – continued

EXPLANATION

Answer



Explanation

There are two constraints:

Must use each of the three programming systems

Avoid using as many **move to (x, y) coordinates** programming commands as possible.

The North South West East programming system can only move the robot horizontally or vertically. That means that the hexagon and triangle can't be drawn with this system, only the square can be drawn with this system.

Students can use the Coordinates programming system for both of the remaining drawings. The triangle has three points and thus needs three coordinate commands, the hexagon has six. This means the triangle should be drawn with the coordinates programming system to satisfy the second constraint. But this is only possible if the robot can draw a hexagon using the Turn Left and Right system.

The hexagon can be programmed using the Turn Left and Right programming system because the turn block allows the programmer to choose how far to turn. Because a hexagon has six of the same angles, the robot repeats going straight and turning 60° a total of 6 times.

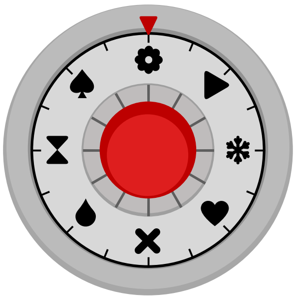
BACKGROUND INFORMATION

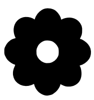
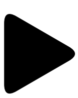
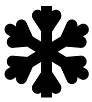
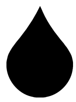
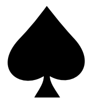
This task is about matching the best programming system for a problem given a number of constraints. In computer science there are often multiple ways to solve a problem - for example, the coding languages of Java and Python can both be used to create programs in many different circumstances. However, Java is often the better choice when building a mobile application, while Python is often the better choice when being used for data science projects. The unique features of each programming language can inform which might be the better choice for a particular project, despite the fact it is theoretically possible to use either.

We can see this in this task- all the shapes could have been programmed with either the Coordinate system or the Left Right system. The North South West East system is not as universal.

Bank Lock

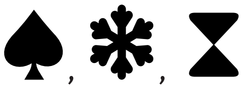
In the Bebras Bank, the passcode to the safe is a combination of three out of eight symbols:



1	2	3	4	5	6	7	8
							

The passcode automatically changes every day. To change the passcode, each symbol is shifted to the right. That is:

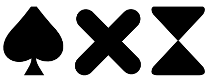
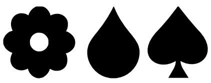
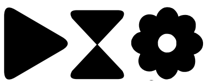
- The symbol is replaced by the symbol to the right.
- The rightmost symbol is replaced by the leftmost symbol.

For example, if on Sunday the passcode is , on Monday the passcode will be .

Last Sunday, a bank manager set the passcode to . Then, the manager wrote a list of passcodes for some days of the following week. However they made one mistake.

Question

Which of the passcodes is wrong?

Wednesday	
Thursday	
Friday	
Saturday	

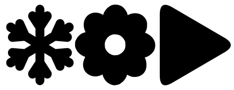
Continued on next page

Bank Lock - continued

EXPLANATION

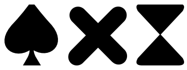
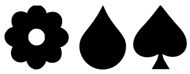
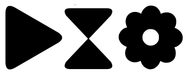
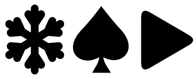
Answer

The correct answer is:

Saturday 

Explanation

By following the rules for changing the daily passcode, the next week should look like this:

The original passcode on Sunday	 (524)
Wednesday	**Wednesday**  (857)
Thursday	**Thursday**  (168)
Friday	**Friday**  (271)
Saturday	**Saturday**  (382)

Therefore, the passcode on Wednesday, Thursday, and Friday are correct. On Saturday, the passcode should be  (382), instead of  (312).

This question can also be solved without working out the next weeks' passcodes and instead by using pattern recogntion:

We know the original passcode is 524. Therefore, the second symbol is always 5 'ahead' of the first symbol, and the third symbol is always 2 'ahead' of the second symbol. This rule is followed by 857, 168, 271, but NOT 312 - the second symbols is 6 'ahead' of the first symbol, and the third symbol is 1 'ahead' of the second symbol.

BACKGROUND INFORMATION

While banks need to keep money safe, others want to keep information safe - in other words, hidden from others. In history, people have utilised various encryption methods to keep their messages secret to others.

The encryption method used in this example is the Caesar cipher: this method substitutes each letter of a message by another letter, some fixed number of positions down the alphabetical order. This method turned out to be unsafe and easy to crack. Relatively few encrypted messages need to be observed in order to discover the system - when encrypting letters using this method, there are only a maximum of 25 ways to encrypt any message.

Modern computing uses modern encryption methods to safeguard important information. Their design is very different from that of historic methods. Their system is well-known, but is based on mathematical properties such that it can be proven that they are at least very hard to crack. However, as technology improves, so to does its ability to break encryption methods. This constant dance between creating better encryption techniques and finding methods to break them has been observed all throughout history.

We would like to thank the International Bebras Committee and community for their ongoing assistance, resources and collaborative efforts. Special thanks to Eljakim Schrijvers, Alieke Stijf and Dave Oostendorp for their support and technical expertise.

If you would like to contribute a question to the International Bebras community, please contact us via the details below.

Contact us

CSIRO Digital Careers
digitalcareers@csiro.au
csiro.au/Digital-Careers

Australia's National Science Agency

CSIRO digital careers