# Bebras Australia Computational Thinking Challenge

## 2021 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

**523**
Australian schools participated in Round 1 2021

**32,311**
Australian students participated in Round 1 2021

**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.

**digitalcareers.csiro.au**

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 1 of the Bebras Australia Computational Thinking Challenge 2021. 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

| 2021 Round 2 Questions | Grade level | Decomposition | Pattern Recognition | Abstraction | Modelling & Simulation | Algorithms | Evaluation |
|---|---|---|---|---|---|---|---|
| **Years 3+4** | | | | | | | |
| Drawing a Robot | **Easy** | ✓ | ✓ | ✓ | | ✓ | |
| Bear Selection | **Easy** | ✓ | ✓ | | | ✓ | ✓ |
| Crypto Keys | **Easy** | ✓ | ✓ | | | ✓ | ✓ |
| Street | **Easy** | ✓ | ✓ | ✓ | | ✓ | |
| Calendar | **Easy** | ✓ | | ✓ | | ✓ | |
| Stamp Collecting | **Medium** | ✓ | ✓ | ✓ | | ✓ | |
| Train Ticket Reservation | **Medium** | ✓ | | ✓ | | ✓ | ✓ |
| Favourite Animals | **Medium** | ✓ | ✓ | ✓ | | | ✓ |
| Book Encoding | **Medium** | ✓ | ✓ | ✓ | | ✓ | ✓ |
| Picking Mushrooms | **Medium** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Library Books | **Hard** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Lisa's Lovely Lyrics | **Hard** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Family Ties | **Hard** | ✓ | | ✓ | ✓ | ✓ | |
| Swapping Cats | **Hard** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Scarf Making | **Hard** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| **Years 5+6** | | | | | | | |
| Water Filter | **Easy** | ✓ | | ✓ | ✓ | ✓ | ✓ |
| Sock Selection | **Easy** | ✓ | ✓ | ✓ | | ✓ | |
| Image Representation | **Easy** | ✓ | ✓ | | | ✓ | ✓ |
| Beaver Samba | **Easy** | ✓ | ✓ | | ✓ | ✓ | ✓ |
| Classifier | **Easy** | ✓ | ✓ | ✓ | | ✓ | |
| Magic Potions | **Medium** | | ✓ | ✓ | ✓ | ✓ | ✓ |
| Chairs | **Medium** | ✓ | ✓ | ✓ | | ✓ | ✓ |
| Tile Laying | **Medium** | ✓ | ✓ | ✓ | | ✓ | ✓ |
| Tree Sudoku | **Medium** | ✓ | ✓ | ✓ | | ✓ | |
| Zoo Animals | **Medium** | ✓ | ✓ | | | ✓ | ✓ |
| Planting Flowers | **Hard** | ✓ | | ✓ | ✓ | ✓ | ✓ |
| Stars and Moons | **Hard** | ✓ | ✓ | | | ✓ | ✓ |
| Secret Message | **Hard** | ✓ | | ✓ | ✓ | ✓ | ✓ |
| Jigsaw Puzzle | **Hard** | ✓ | ✓ | ✓ | | ✓ | ✓ |
| Dangerous Virus | **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

| 2021 Round 2 Questions | Abstraction | Data Collection | Data Representation | Data Interpretation | Specification | Algorithms | Implementation | Digital Systems | Interactions | Impacts |
|---|---|---|---|---|---|---|---|---|---|---|
| **Years 3+4** | | | | | | | | | | |
| Drawing a Robot | ■ | | ■ | ■ | ■ | ■ | | | | |
| Bear Selection | | | ■ | ■ | ■ | ■ | | | | |
| Crypto Keys | ■ | | ■ | ■ | ■ | ■ | | ■ | | |
| Street | ■ | | ■ | ■ | ■ | ■ | | | | |
| Calendar | ■ | | | | ■ | ■ | | | | |
| Stamp Collecting | ■ | | ■ | ■ | ■ | ■ | | | | |
| Train Ticket Reservation | ■ | | ■ | ■ | ■ | ■ | | ■ | | ■ |
| Favourite Animals | ■ | | ■ | ■ | ■ | ■ | | | | |
| Book Encoding | ■ | | ■ | ■ | ■ | ■ | | ■ | | |
| Picking Mushrooms | ■ | | ■ | ■ | ■ | ■ | | | | |
| Library Books | ■ | | ■ | ■ | ■ | ■ | ■ | ■ | | |
| Lisa's Lovely Lyrics | ■ | | ■ | ■ | ■ | ■ | ■ | ■ | | |
| Family Ties | ■ | | ■ | ■ | ■ | ■ | | | | |
| Swapping Cats | ■ | | ■ | ■ | ■ | ■ | | | | |
| Scarf Making | ■ | | ■ | ■ | ■ | ■ | | | | |
| **Years 5+6** | | | | | | | | | | |
| Water Filter | ■ | | ■ | ■ | ■ | ■ | | | ■ | |
| Sock Selection | ■ | | ■ | ■ | ■ | ■ | | | | |
| Image Representation | | | ■ | ■ | ■ | ■ | ■ | ■ | | |
| Beaver Samba | | | ■ | ■ | ■ | ■ | | | | |
| Classifier | ■ | | ■ | ■ | ■ | ■ | | ■ | | |
| Magic Potions | ■ | | ■ | ■ | ■ | ■ | | | | |
| Chairs | ■ | | ■ | ■ | ■ | ■ | | | | |
| Tile Laying | ■ | | ■ | ■ | ■ | ■ | | | | |
| Tree Sudoku | ■ | | ■ | ■ | ■ | ■ | ■ | | | |
| Zoo Animals | | | ■ | ■ | ■ | ■ | | | | |
| Planting Flowers | ■ | | ■ | ■ | ■ | ■ | | | | |
| Stars and Moons | | | ■ | ■ | ■ | ■ | ■ | ■ | | |
| Secret Message | ■ | | ■ | ■ | ■ | ■ | | ■ | ■ | |
| Jigsaw Puzzle | ■ | | ■ | ■ | ■ | ■ | | | | |
| Dangerous Virus | ■ | | ■ | ■ | ■ | ■ | | ■ | ■ | ■ |

# Bebras Challenge 2021 Round 2

## Years 3+4

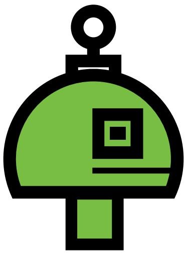
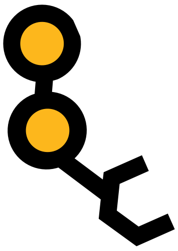
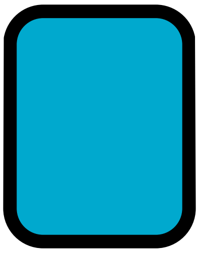
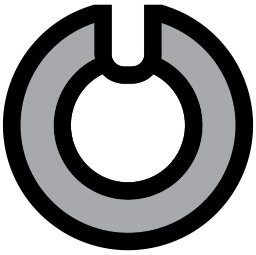
This question comes from
Serbia

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

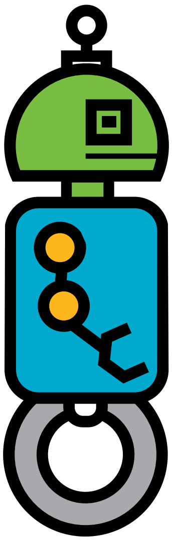
# Drawing A Robot

Jelena loves to draw using a computer. Today she is drawing a robot.

First, she drew the following parts of the robot:

| | | | |
|---|---|---|---|
| Head: |  | Arm: |  |
| Body: |  | Wheel: |  |

After that, she merged them into the image shown below:



Each new robot part in the picture goes above the previous one, if they overlap.

## Question

Which order of parts did Jelena use?

*Select one of the following options.*

**head, wheel, body, arm**

**wheel, body, head, arm**

**body, wheel, arm, head**

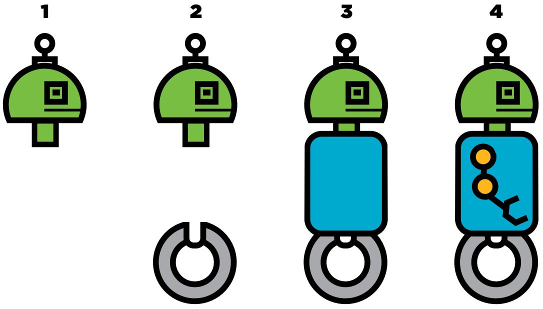
**wheel, head, arm, body**

*Continued on next page*

# Drawing A Robot – continued

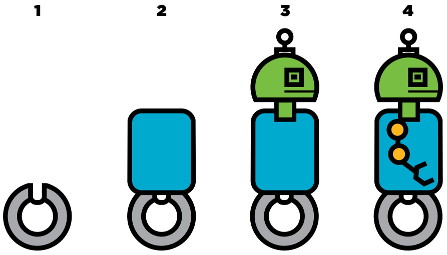
## Answer

**The answer is: head, wheel, body, arm**

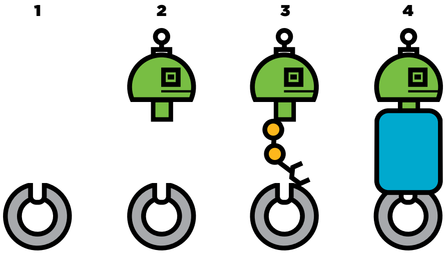


Below, the wrong combinations are explained:

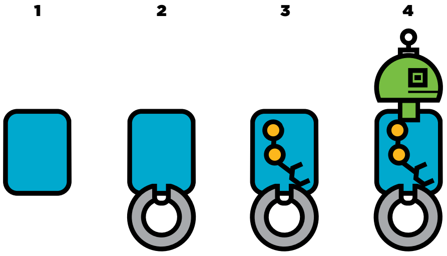
wheel, body, head, arm (look at the neck, it overlaps the body)



wheel, head, arm, body (the arm disappears behind the body)



body, wheel, arm, head (both the wheel and the neck overlap the body)



## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

Layers offer the ability to manage and organize complex designs easily, by separating a large design into sections of related objects.

Each layer is like a sheet of transparent paper drawn on top of the layers below it. Layers are drawn as a stack, with the layer at the bottom of the list drawn first and each following layer drawn on top of the previous one.

This question comes from
Canada

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

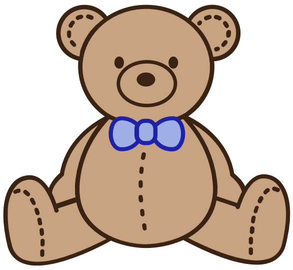
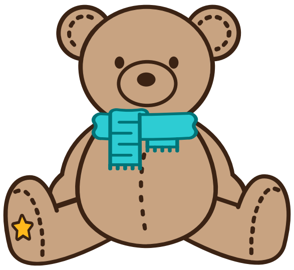
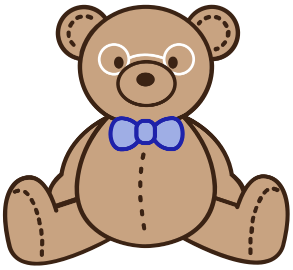
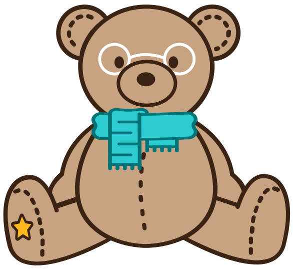
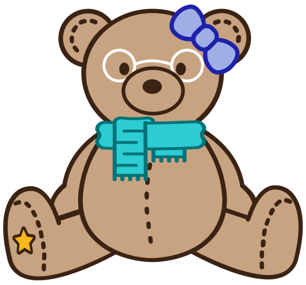
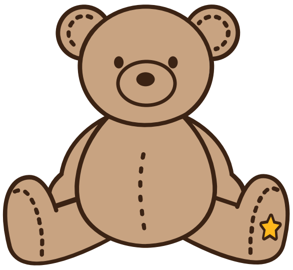
# Bear Selection

Ren is allowed to bring one of his teddy bears to school for show and tell. Ren chooses a bear that has a star on its foot, and is wearing a scarf or a bow, but not glasses.

## Question

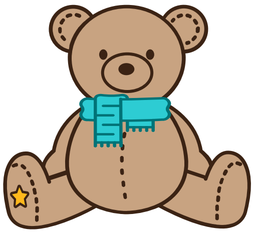
Which bear does Ren choose?

*Select one of the following options.*

## Answer

**The answer is:**



Ren did not choose , or because those bears have glasses.

Ren did not choose since that bear is not wearing a scarf or a bow.

Ren did not choose or as those bears do not have a star on their feet.

Ren chose because this bear has a star on its foot, is wearing a scarf, and does not have glasses.

*Continued on next page*

# Bear Selection – continued

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Algorithms, Evaluation

**Concepts:** Data Representation, Data Interpretation, Specification, Algorithms

Ren decides that the bear he takes to school has to meet a specific set of requirements. This set of requirements comprises of 3 different statements about the bear:

- It has a star on its foot,
- It is wearing a scarf or a bow,
- But not glasses

This task requires the use of *boolean logic*. Each of Ren’s requirements is expressed as a *boolean statement*. A boolean statement has the property of being either *true or false*.

The individual statements are then combined together using boolean operators **AND**, **OR**, and **NOT**.

The meaning of these operators is similar, but not identical, to how these words are understood in English.

- **AND** is true if all statements it connects are true.
- **OR** is true if at least one of the statements it connects is true.
- **NOT** is true if a statement is false.

In the case of Ren’s set of requirements, all 3 individual statements are combined by the boolean operator **AND**. Note that the second and third **AND** is present implicitly in the colloquial version as it is expressed by a comma.

1. It has a star on its foot **AND**
2. It is wearing a scarf **OR** a bow **AND**
3. It does **NOT** have glasses

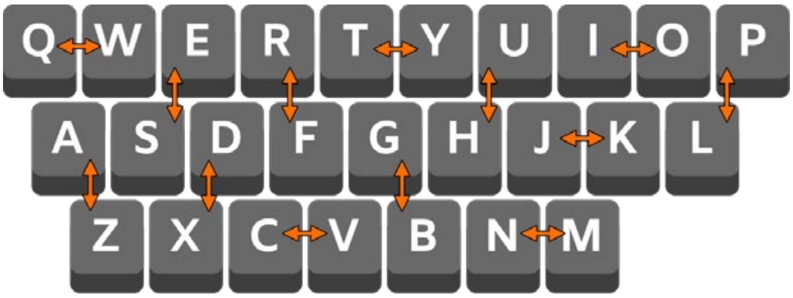
This means that in order to meet Ren’s requirements, every single one of the 3 statements about the bear needs to be *true*.

This question comes from
Canada

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

# Crypto Keys

Jan has a special keyboard for writing secret messages. When a key is pressed, a different letter is displayed on the screen, according to the following keyboard map:



The arrows indicate which letter is displayed when a key is pressed.

**Example:** When Jan presses ‘S’, the letter ‘E’ is displayed on the screen, and when Jan presses ‘E’, the letter ‘S’ appears.

Jan’s screen displays a secret message with the following letters: ‘NIFMOMB’.

## Question

What is the original message written by Jan?

*Select one of the following options.*

**MOLDING** **MERMAID** **MORNING** **MICROBE**

## Answer

**The answer is: c) morning**

Jan’s secret message can be figured out by matching the key pressed with the letter displayed on Jan’s screen, according to the arrows on the keyboard map.

| | N | I | F | M | O | M | B |
|---|---|---|---|---|---|---|---|
| becomes | M | O | R | | | | |

From here we can deduce that the secret message must be C, that is, ‘MORNING’, but let’s continue matching the rest of the letters just to be sure.

| | N | I | F | M | O | M | B |
|---|---|---|---|---|---|---|---|
| becomes | M | O | R | N | I | N | G |

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Digital Systems

Cryptography is a part of Computer Science. The keyboard used in this task is based on the monoalphabetic sustitution cipher example called the *Vatsyayana cipher*.

The idea came from an Indian text from the 4th century AD. The Vatsyayana cipher creates unique pairs for the alphabet letters – one letter always matches another letter and one letter can only be used in one pair.

During encryption in an original message one letter is directly substituted by a paired letter. Interestingly, the exact same process is used to decrypt the message. By directly substituting a letter in the encrypted message with its paired letter, the original message can be retrieved.

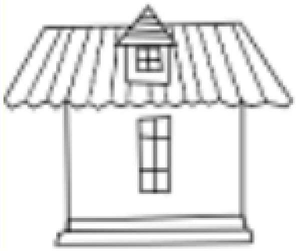
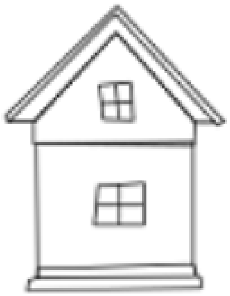
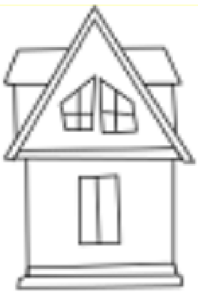
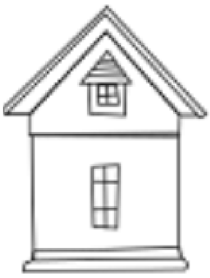
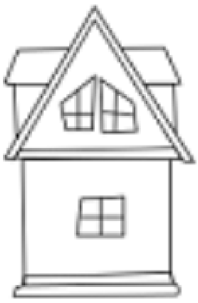
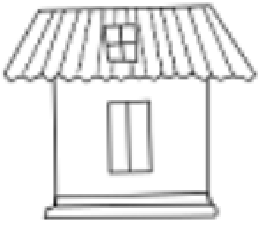
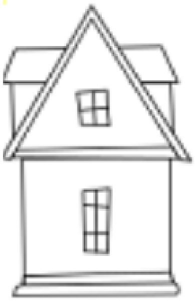
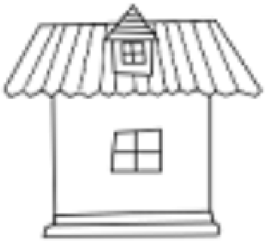
This encryption method is easy to crack as, once someone is sure about a letter pair, they can decrypt all other pairs.

This question comes from
Hungary

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

# Street

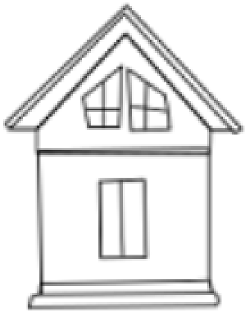
Sue draws streets using a system. Can you recognise it?

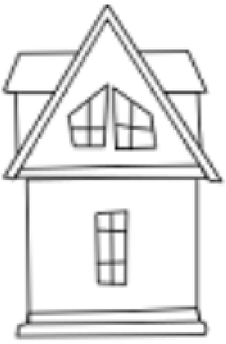
| | | | |
|---|---|---|---|
| **Street 1** |  |  |  |
| **Street 2** |  |  |  |
| **Street 3** |  |  | |

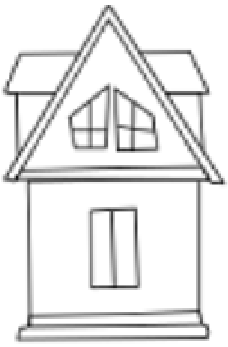
## Question

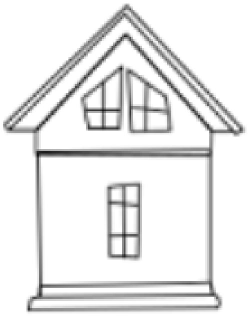
Which house is missing from the 3rd street?

*Select one of the following options.*







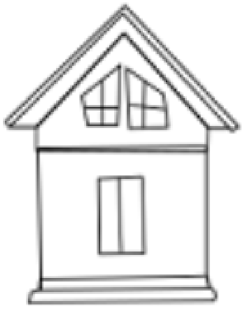


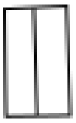
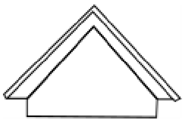
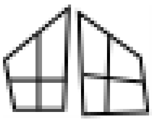
*Continued on next page*

# Street – continued

## Answer

**The answer is: A**



The answer is A) because we don't have a house with this window , with this roof , and with this dormer window .

Each house has its own style of window, roof and dormer window in each street. You have to find the house which has the missing style from these in the third street.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

In Computer Science we often work with objects that have attributes. In this task you had to concentrate on 3 attributes and the values. Each attribute could have 3 different values. Computer scientists like to work with large quantities of objects and find the common attributes or find objects different from the other objects in all attributes. This can help to find anomalies, illnesses or a wrong behavior within a system.

This question comes from Latvia

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

# Calendar

The day before, three days ago, was the day before Sunday.

## Question

Which day will tomorrow be?

*Select one of the following options.*

Monday    Sunday    Thursday    Wednesday

## Answer

**The answer is: Thursday**

The day before three days ago was four days ago. It was Saturday (the day before Sunday).

That means that today is Wednesday, and that tomorrow will be Thursday.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Abstraction, Algorithms

**Concepts:** Abstraction, Specification, Algorithms

This task is based on the ability to use logic.

Logic plays a key role in Computer Science (databases, computational complexity, programming languages, artificial intelligence, verification, hardware and software design etc.), It is undoubtedly one of the foundations that will provide the agility to assimilate future Computer Science concepts, languages, techniques, etc.

This question comes from
Australia

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

# Stamp Collecting

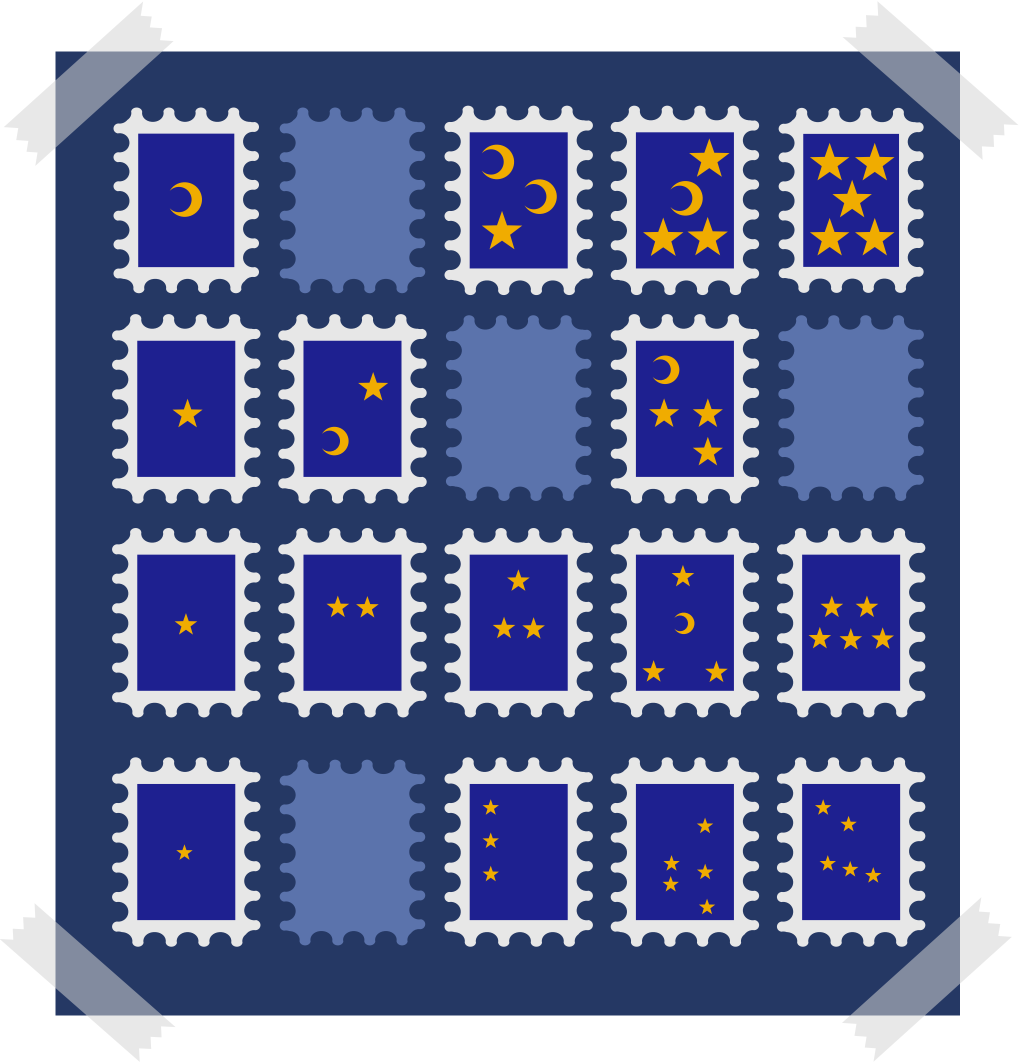
Beaver Jo loves collecting and organising her stamp collection.

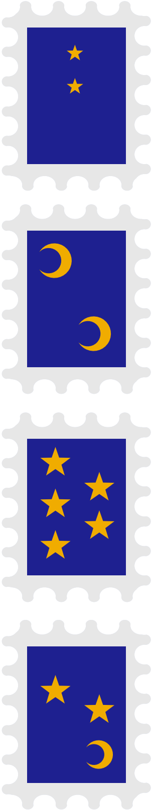
Sadly, some of her stamps fell out or order!

## Question

Where do the missing stamps belong?

*Help Jo organise the stamps by dragging them into the correct positions.*



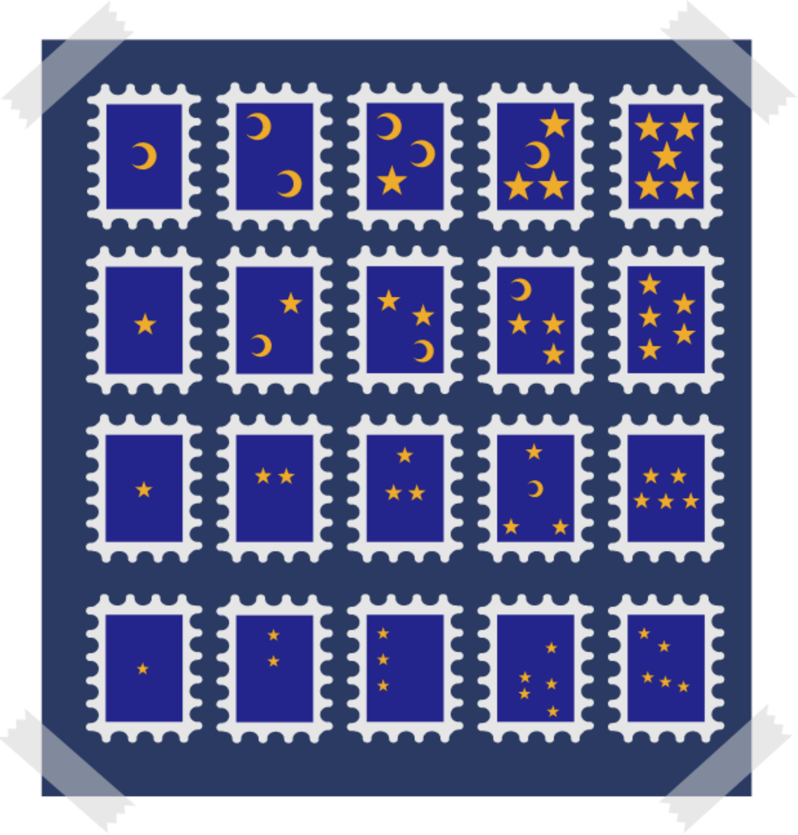


*Continued on next page*

# Stamp Collecting – continued

## Answer

**The answer is:**



The correct answer follows the application of two rules:

- The number of objects increases left to right
- The size of the shapes on the stamp decreases from top to bottom down the collection.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

**Sorting:** Data should be stored in a way that helps to handle it as quickly as possible. Most common data structures try to organise in some order. Here we see a sorting problem where we have two parameters:

1. Size of the objects
2. Number of objects

The one way to meaningfully organise this dataset is to sort stamps and present them in some order. So, each row has stamps with objects of the same size, they are sorted from left to right according to the number of objects. This order is called ascending. The columns have stamps with the same number of objects and are sorted according to the size of the objects in stamps from the largest to the smallest. This order is called descending.

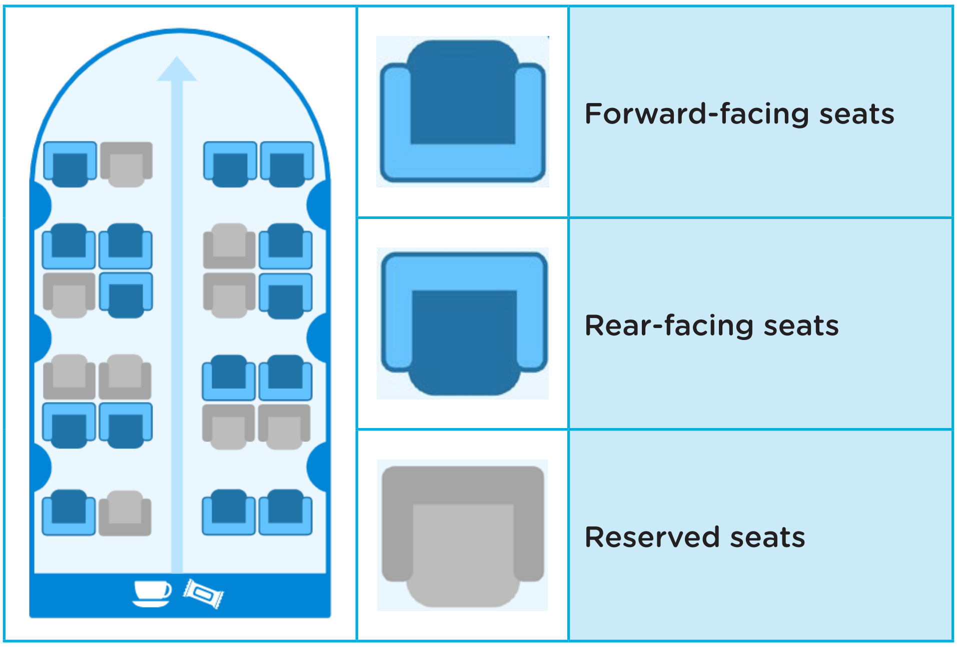
**Data, Data Structures and Representations:** Data can take many forms, for example, pictures, text or numbers. When we look at data in this question, we are looking for a sequence of images that will assist in solving the problem. By identifying these images we can make predictions, create rules and solve more general problems.

This question comes from
South Korea

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

# Train Ticket Reservation

Alex and Bob decided to go on a train trip this weekend. When they accessed the train reservation software to book the train tickets, the following screen appeared.



Reserved seats cannot be selected. Here is what Alex and Bob should consider when booking the tickets:

- Alex and Bob want to sit next to each other.
- Alex wants to sit in the forward position (because of his motion sickness).
- Bob wants to sit as close as possible to the snack bar at the back of the train.

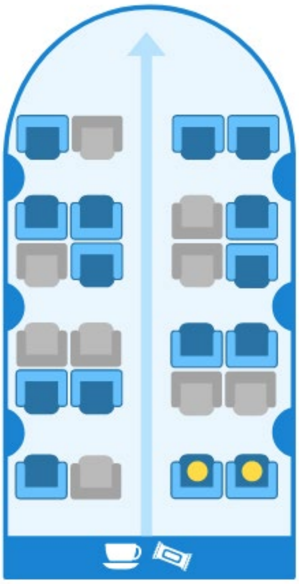
## Question

Which seats should Alex and Bob book?

*Select the two seats that satisfy both Alex and Bob's needs.*

## Answer

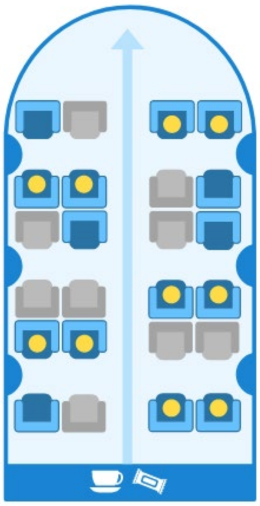
**The answer is:**



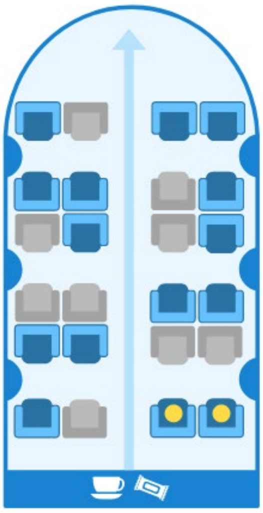
*Continued on next page*

# Train Ticket Reservation – cont.

The two available seats must be beside each other. The seats that satisfy this requirement are indicated by the yellow circles in the image below.



On the other hand, Alex wants to sit facing forward, and Bob wants to be close to the snackbar. The place that satisfies both of these requirements are the two forward-facing seats in the last row. Therefore, the correct answer is shown in the image below.



## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Algorithms, Evaluation

**Concepts:** Abstraction, Data Collection, Data Representation, Data Interpretation, Specification, Algorithms, Implementation, Digital Systems, Interactions, Impacts

This task deals with the structure of a conditional statement that selects and executes a different task depending on the result when conditions are given. This conditional statement structure is used in all programming languages.

In this task, Alex and Bob need to check the conditions (available for reservation, sitting together, forward-facing seats, distance from snackbar) and select seats.

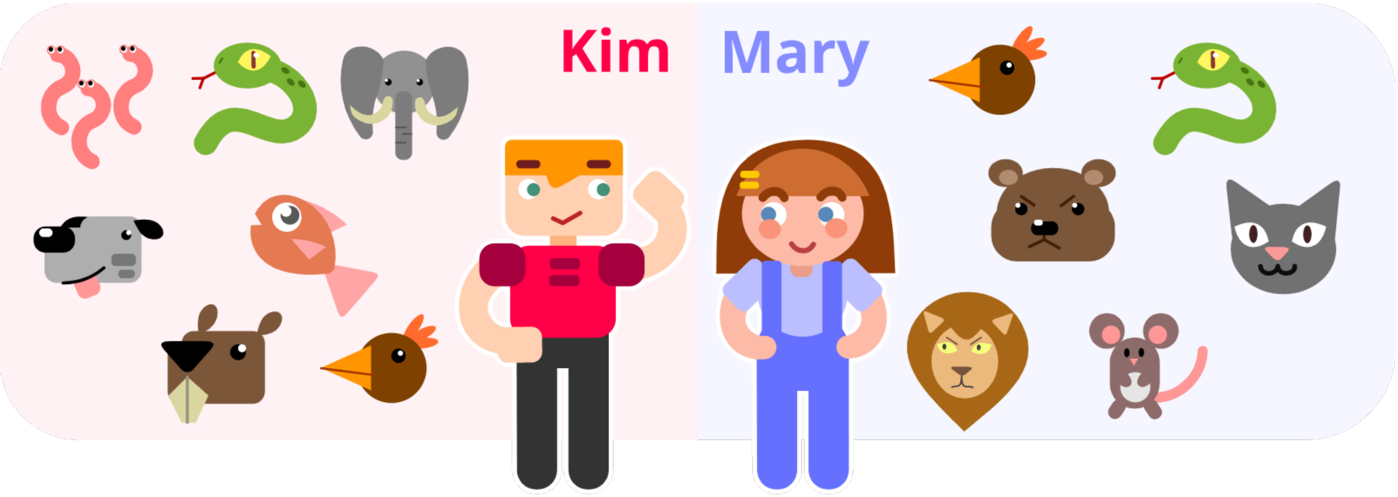
One of the useful processes to solve this kind of task is to exclude items that do not meet the conditions. When creating software, we collect and analyze the features that users want, which we call user requirements. These requirements serve as rules, criteria or conditions for creating software.

This question comes from
the United States

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

# Favourite Animals

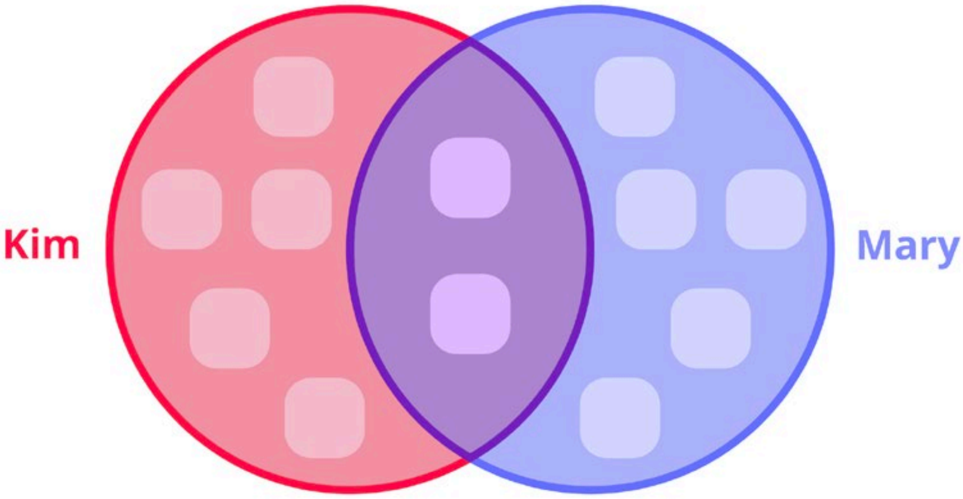
Kim and Mary are talking about which animals they like. After chatting, they find out that there are certain animals they both like.

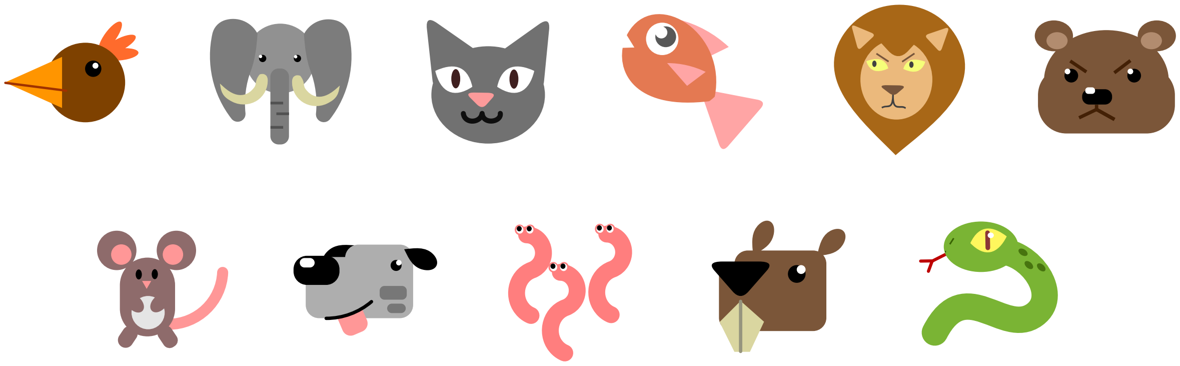


They decide to put the animals in the diagram below.

## Question

Which animals do Mary and Kim both like?



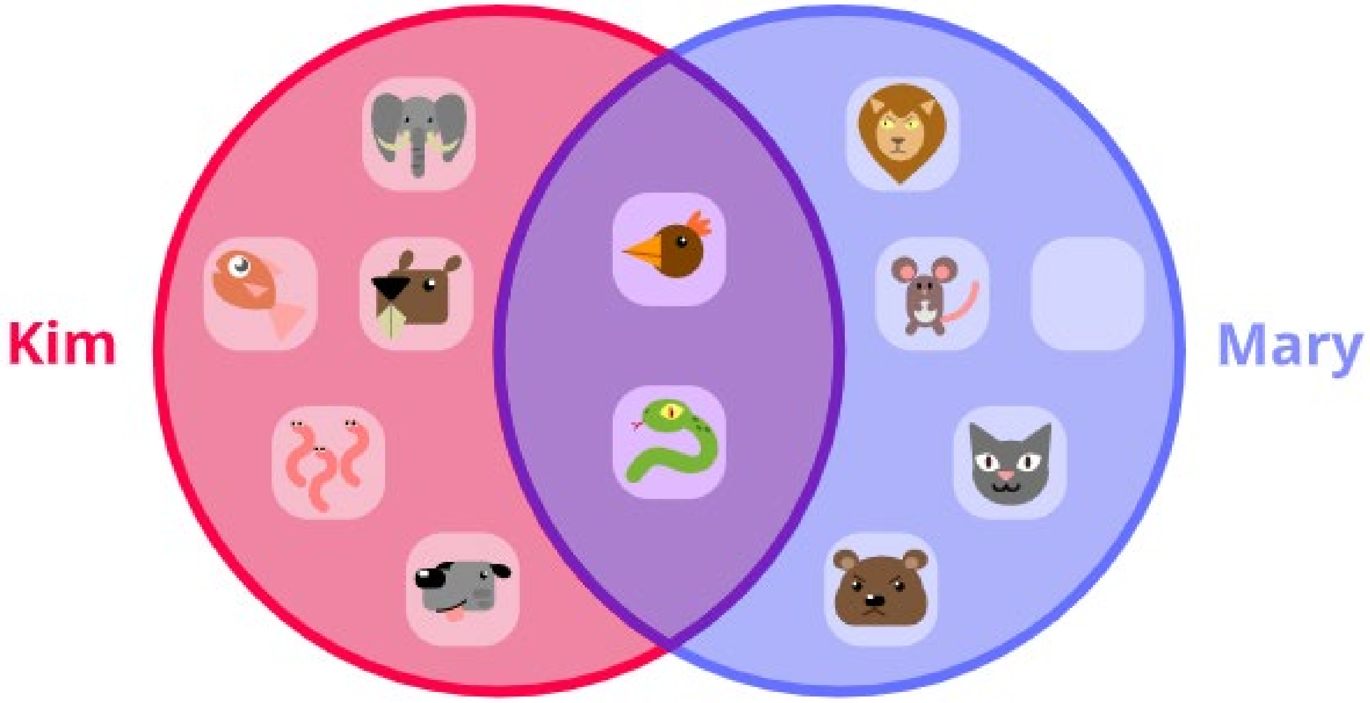


*Continued on next page*

# Favourite Animals – continued

## Answer

**The answer is: both like birds and snakes**



It does not matter in which order the animals are placed into the circles.

**Only Kim likes:** beavers, fish, earthworms, dogs, elephants

**Only Mary likes:** lions, mice, cats, bears

**Both like:** snakes, birds

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

When storing information in a computer you try to store things in an efficient way. That way your information does not take up too much space.

Kim and Mary are using a Venn-diagram to store information about which animals they like. This way every animal appears only once in the picture, even if they both like it.

Their Venn diagram is not only useful because it saves space, it also makes it immediately clear which animals they both like. You don’t have to search for this, you just look in the overlapping circle.

If you look at each animal, some are loved by Kim **OR** Mary, and some are loved by Kim **AND** Mary. The **AND** animals are shown in the overlap.

This question comes from
Taiwan

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

# Book Encoding

The librarian beaver, collects many precious books written in Bebras language. To store books, there are 10 bookcases in the Bebras Library, and in each bookcase, there are 10 shelves.

The librarian beaver, assigns a special number to each book to represent its location so that the books can be found quickly.

The special number is assigned to each book based on the Bebras letters in the book title according to the table below:

| Bebras letter | □ | △ | ☆ | ○ | ◎ | ◇ | ♡ | ☁ |
|---|---|---|---|---|---|---|---|---|
| Location | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |

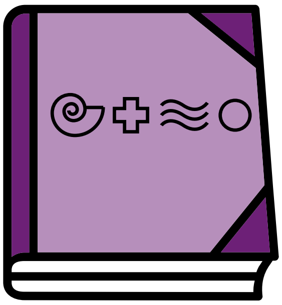
| Bebras letter | ✚ | ✿ | ☾ | 🐟 | ⛰ | ≋ | 💧 | 🐚 |
|---|---|---|---|---|---|---|---|---|
| Location | **9** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |

Example: If the title of the book is ◎ ◇ ○ ☆ △,

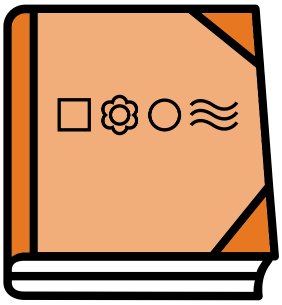
◎ ◇ ○ ☆ △
then its special number will be 56432: **5 6 4 3 2**

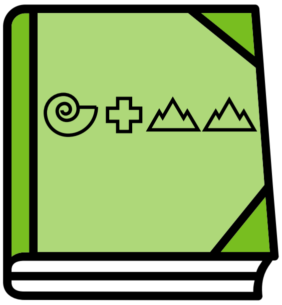
## Question

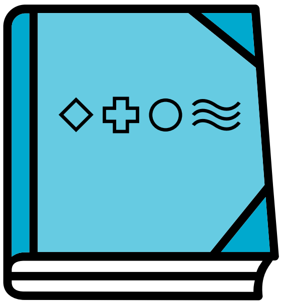
Which of the following books will have the same special number as the purple book below?

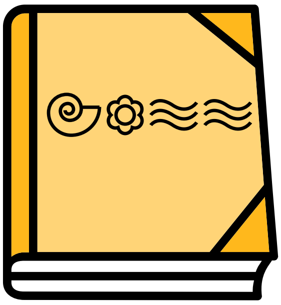


*Select one of the following options.*









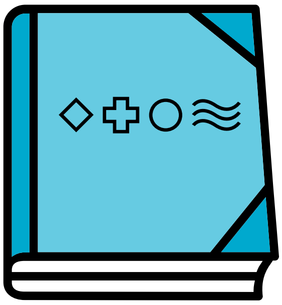
*Continued on next page*

This question comes from Taiwan

# Book Encoding – continued

## Answer

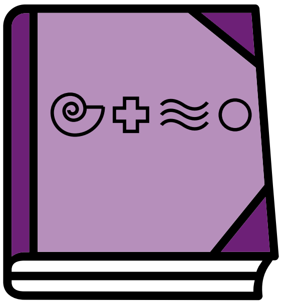
**The answer is: C**



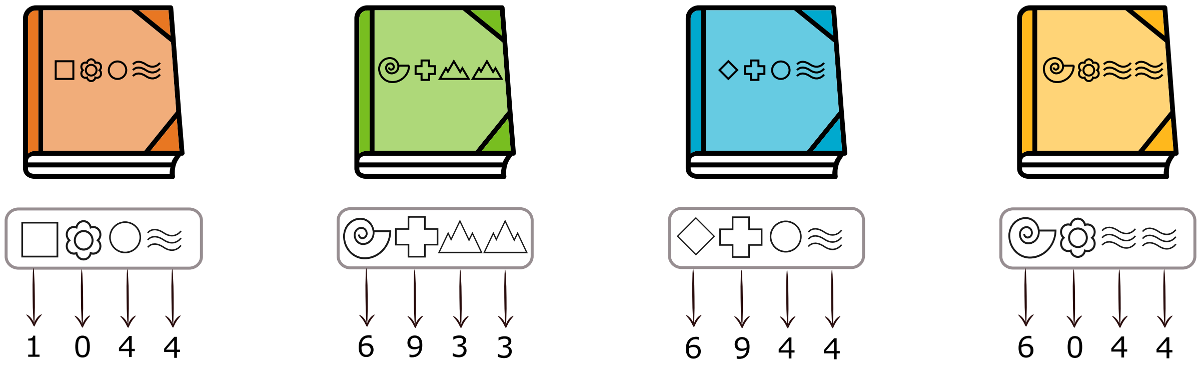
| Bebras letter | □ | △ | ☆ | ○ | ◎ | ◇ | ♡ | ☁ |
|---|---|---|---|---|---|---|---|---|
| Location | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |

| Bebras letter | ✚ | ❀ | ☾ | 🐟 | ⛰ | ≋ | 💧 | 🐚 |
|---|---|---|---|---|---|---|---|---|
| Location | **9** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |

First find the special number of the purple book: 6944



Now find the special numbers for each of the books in the answer options and determine which book has the special number 6944.



We can see that the book in option 3 has the special number 6944.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms, Evaluation

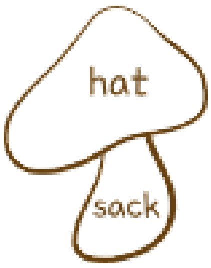
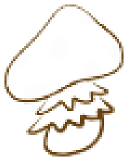
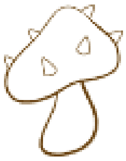
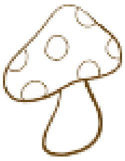
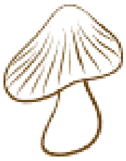
**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Digital Systems

In Computer Science, the 'hash table' is a type of data structure often used to store data that are frequently queried. The hash table uses an index code (usually a numerical code), which is called a 'key' to indicate the location of data storage. In this task, the mapping table method is used to generate keys for each book. However, as it happens, the same key can correspond to multiple books. This situation is called 'collision'. When designing a method of key generation for a hash table, computer scientists need to consider the possibility of collisions and design a way to deal with them.

# Picking Mushrooms

Beavers do not eat mushrooms, they just collect them for the Mushroom Museum.

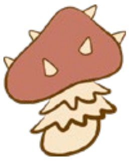
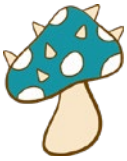
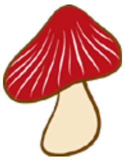
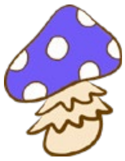
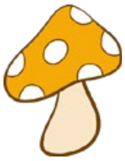
The museum is only interested in rare mushrooms. The rarity of the mushroom is defined by the scores in the following table:

| Shape and score for each mushroom | | | | | Mushroom rarity score |
|---|---|---|---|---|---|
| | sack | hat | | | |
|  | <br>multiple layers<br>1 point | <br>horned<br>2 points | <br>dotted<br>3 points | <br>striped<br>5 points | Mushroom total score is<br>**0–3 points = common**<br>**4 points or more = rare** |

Beaver Somi goes to the forest to collect rare mushrooms to show in the museum.

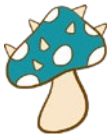
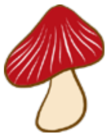
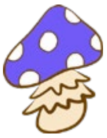
## Question

Which rare mushrooms should Somi pick?

## Answer

**The answer is:**

Based on the results from analysing each mushroom shape, this task can easily be solved by creating a score table as follows:

| | Multiple layered sack | Horned hat | White dotted hat | White striped hat | Total mushroom score | Rarity |
|---|---|---|---|---|---|---|
| | 1 point | 2 points | - | - | 3 points | Common |
| | - | 2 points | 3 points | - | 5 points | Rare |
| | - | - | - | 5 points | 5 points | Rare |
| | 1 point | - | 3 points | - | 4 points | Rare |
| | - | - | 3 points | - | 3 points | Common |

*Continued on next page*

# Picking Mushrooms – continued

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

This is an example of a classification task, we try to classify each object with a certain label based on its characteristics (usually called features). For example, given a set of mushrooms we classify each of them as collectible or not based on its features (colour and shape). Common real life examples of classification tasks include classifying medical images, documents by topic, facial recognition, and spam email detection.

The table is a function that assigns a score to an object based on the characteristics of that object. Such a score is typically totalled from different individual values which each are based on a single aspect of that object. This kind of a function is used in computers quite often because in many cases it is simpler to measure certain aspects and then add up the individual values. In many cases the score is then compared to a threshold to handle the object differently.

A typical example for this kind of a function would be a spam filter for emails. Based on keywords in the message, transmission details, or sender reputation values like +2.5 or -1.0 are assigned to a message and if the sum of these values is larger than, for example, 5.0 the message is considered spam.

This procedure could be automated by writing a computer program that would allow the classification of millions of mushrooms in a short amount of time. Similar programs are used everyday to solve classification tasks automatically. Given the incredibly large amount of data available, classifying instances manually by a human is not possible because of time, money and privacy restrictions. Imagine if every email had to be read by someone before classifying it as spam or not!

In the example above, the algorithm for classification is already given to us, we only have to apply the rules. What if we wanted to build a new classification algorithm? Let's say for example that we want to develop a program that classifies an image as a 'cat' or 'dog' depending on whether a cat or a dog is in the image. There are two main approaches to solve this problem:

1. One based on expert knowledge of the application domain
2. Another based on having a large quantity of labeled data

In the first case, we would talk with cat and dog experts that would describe which features are most important for each species (ear shape, size, or type of fur) and attempt to write a program to detect those specific features. This is the traditional approach and, in general, very hard and expensive to do.

The second approach is based on machine learning. In this approach we use a program that can automatically learn to classify, if it is given sufficient examples. So, if we have one million cat pictures and one million dog pictures, we can train the program so that new images with no label can be identified by the program as more similar to the cat images or the dog ones. This approach, based on big data and machine learning, has recently allowed many new applications and is one of the reasons why data and privacy are so important today.

This question comes from
Canada

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

# Library Books

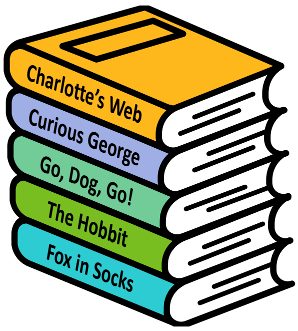
Beavertown Library only has a small pile of books.

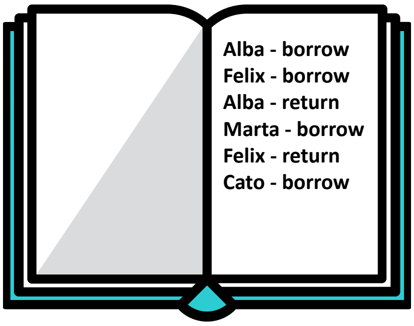
When a beaver borrows a book, they record their name and take the book from the top of the pile.

When a beaver returns a book, they record their name and place their book back on the top of the pile.

At the beginning of the week, the pile of books is arranged as shown:

The library log book for the week shows the following information:





Alba - borrow
Felix - borrow
Alba - return
Marta - borrow
Felix - return
Cato - borrow

## Question

Which book did Cato borrow?

*Choose one of the following options.*

Charlotte's Web

Curious George

Go, Dog, Go!

The Hobbit

## Answer

**The answer is:** ***Curious George***

Alba and Felix started, Alba borrowed *Charlotte's Web* from the top of the pile.
Felix then picked up *Curious Georg*e from the top.
Alba returned *Charlotte's Web* to the top of the pile.
Marta then borrowed *Charlotte's Web* from the top of the pile.
Felix returned *Curious George* to the pile.
When Cato borrowed her book, she picked up the top book, the one Felix had returned.
So Cato borrowed *Curious Georg*e.

It is interesting to note that in order to solve the task we only need to consider which was the last book returned before Cato borrowed a book.

Some of the borrowing and returning during the week – which book Alba returned or which book Marta borrowed – does not need to be considered.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Implementation, Digital Systems

The pile of books forms a *stack*. A stack is a structure that can only be manipulated from one end, which in this case is the top of the pile. Two useful stack operations are:

- *push* (which adds an item to the stack)
- *pop* (which removes an item from the stack)

Stacks have a *last in, first out* (LIFO) properties. This means that the last item *pushed* is the first item *popped*. In this task, the *last* book returned is the *first* book borrowed.

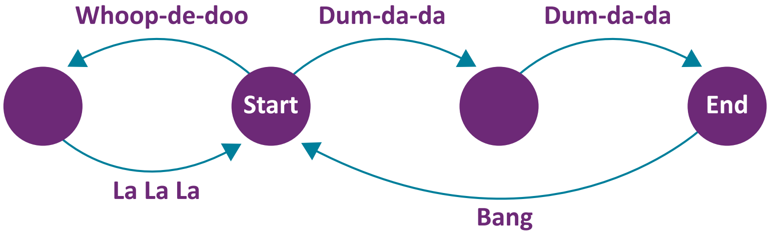
In Computer Science, a stack is a useful way to store and organise data that has these LIFO properties. For example, a web browser's back button makes use of a stack: the last page visited is the first page revisited.

This question comes from Switzerland

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

# Lisa’s Lovely Lyrics

Beaver Lisa likes lyrics. Not just any kind of lyrics though, only *Lisa’s Lovely Lyrics*, all of which can be sung using the following diagram:



To sing one of her songs, Lisa simply begins at *Start* and then follows the arcs to put the lyrics in the correct order.

She may only finish her song once she reaches *End* **but** she does not have to stop when she reaches *End*.

## Example

Possible songs are therefore:

*Whoop-de-doo lalala Whoop-de-doo La La La*

*Dum-da-da Dum-da-da Bang Dum-da-da Dum-da-da*

or

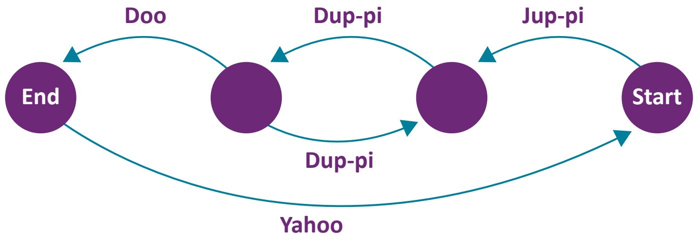
*Dum-da-da Dum-da-da Bang Whoop-de-doo La La La*

*Dum-da-da Dum-da-da Bang Whoop-de-doo La La La*

*Dum-da-da Dum-da-da Bang Dum-da-da Dum-da-da*

## Question

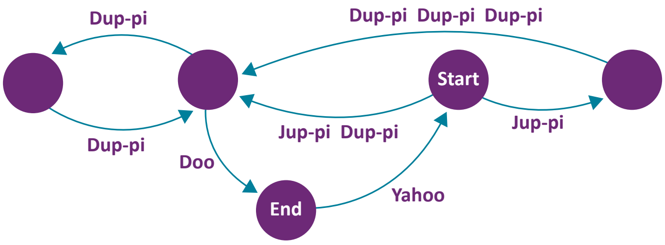
Consider the following diagram, which Lisa uses for alternative *Lisa’s Lovely Lyrics*:



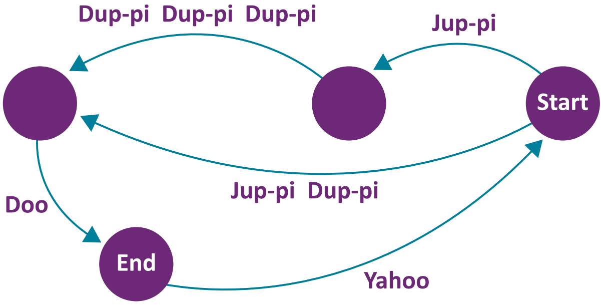
Which of the following three diagrams is for the same songs?

*Select one of the following.*

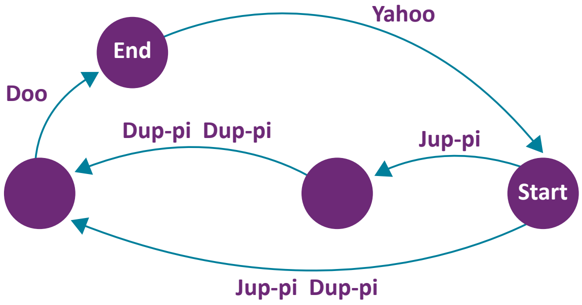
**A**



**B**



**C**

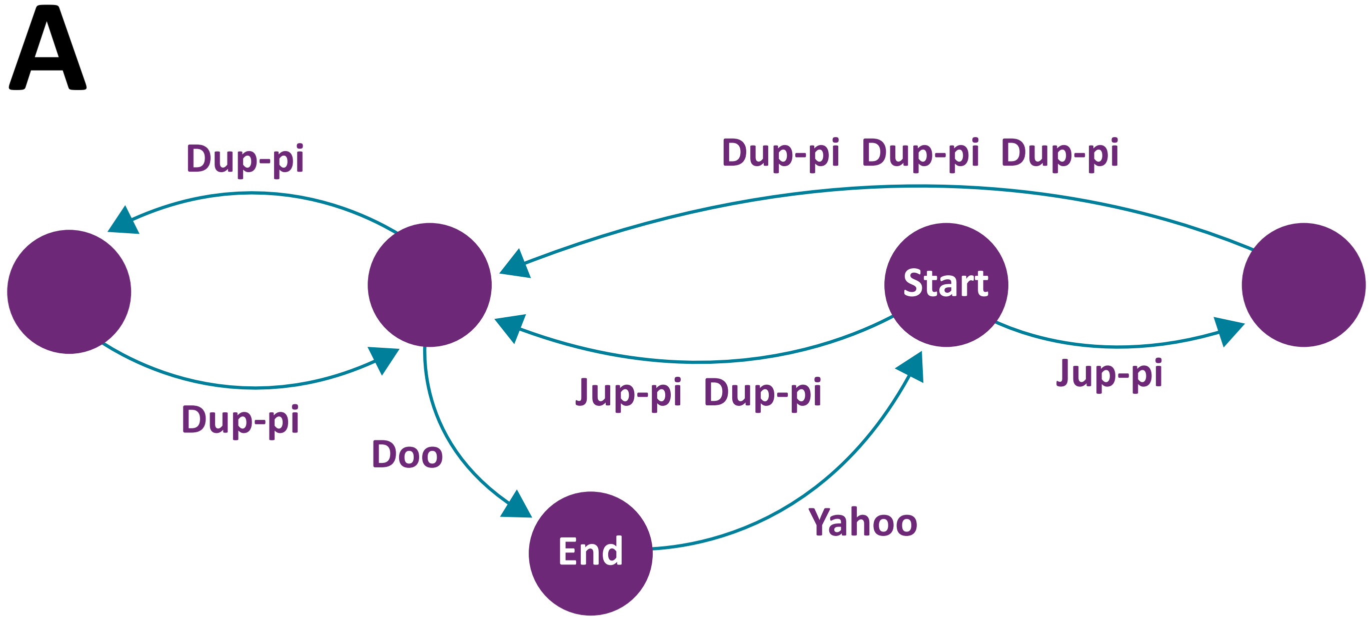


*Continued on next page*

# Lisa’s Lovely Lyrics – continued

## Answer

**The answer is: A**



The first diagram is for the same songs as the original diagram. The crucial observation here is that ‘*Dup-pi*’ can appear an odd number of times consecutively, but never an even number of times.

For option B, it can only appear once or three times consecutively, but not five times.

For option C, it can only appear once or twice consecutively.

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Implementation, Digital Systems

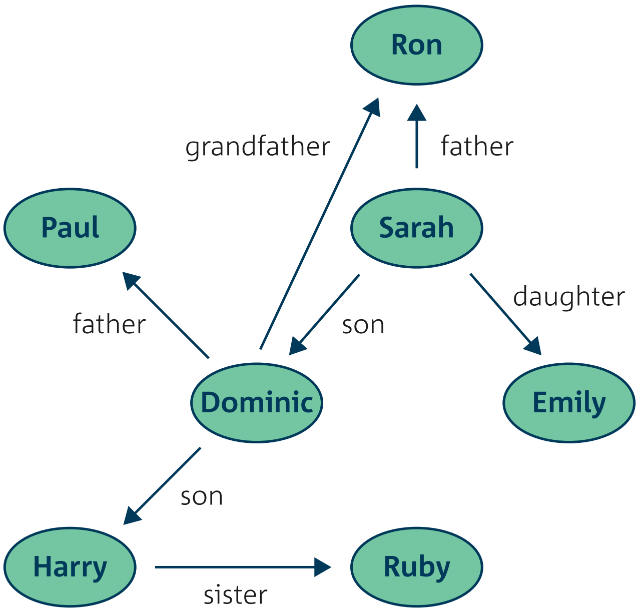
To a large extent, Computer Science deals with identifying and using structures in data. In this task, we consider structured texts (song lyrics) that are designed following a strict system of rules. The generating mechanisms (diagrams) are referred to as *finite automata*. These play a key role in the design of programming languages: languages that can be ‘understood’ by computers.

Pattern recognition in problem solving is key to determining appropriate solutions to problems and knowing how to solve certain types of problems. Recognising a pattern, or similar characteristics helps break down the problem and also build a construct as a path to the solution.

# Family Ties

Sarah is mother to Dominic and Emily. Dominic is father to Harry. Harry is brother to Ruby.

Ron is father to Sarah. Paul is father to Dominic. Ron is grandfather to Dominic.



## Question

How is Ron related to Emily?

*Select one of the following options.*

| | |
|---|---|
| Ron is Emily's father | Ron is Emily's brother |
| Ron is Emily's uncle | Ron is Emily's grandfather |

## Answer

**The answer is: d) Ron is Emily's grandfather**

He cannot be her father, brother or uncle based on the information provided.

The only right answer can be that he is her grandfather, as it shows that Ron is Sarah's father, thus Emily's grandfather.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Abstraction, Modelling & Simulation, Algorithms

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

Data can take many forms, for example pictures, text or numbers. When we look at the data in this question, we are looking for a sequence of images that will assist in solving the problem. By identifying these images we can make predictions, create rules and solve more general problems.

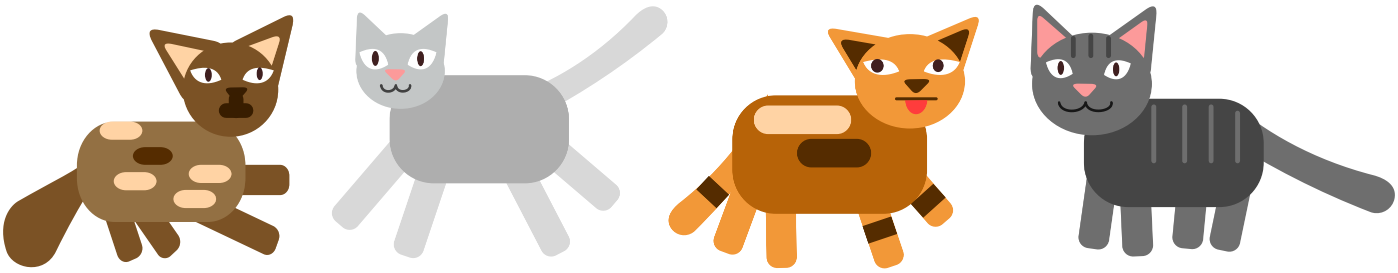
A graph is used in this question to represent the relationships in the family. This can be a useful planning device when working with data and is often used when planning databases to store large amounts of data. You use databases all the time without even knowing it. For example, when you buy items in a store, the item and its cost are stored in a database.

This question comes from
Canada

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

# Swapping Cats

Four cats stand in a line as shown below.



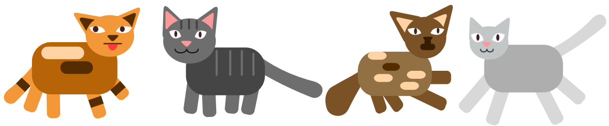
A swap occurs when any two cats exchange positions.

## Question

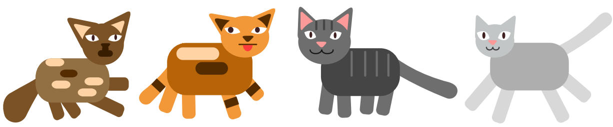
If exactly two swaps occur (one after the other) which of the following cannot be the result?

*Select one of the following options.*

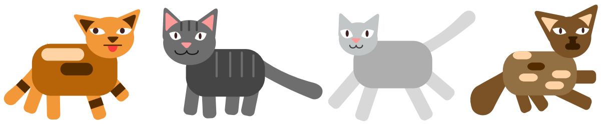
A



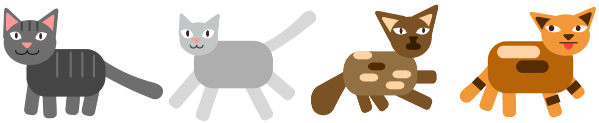
B



C

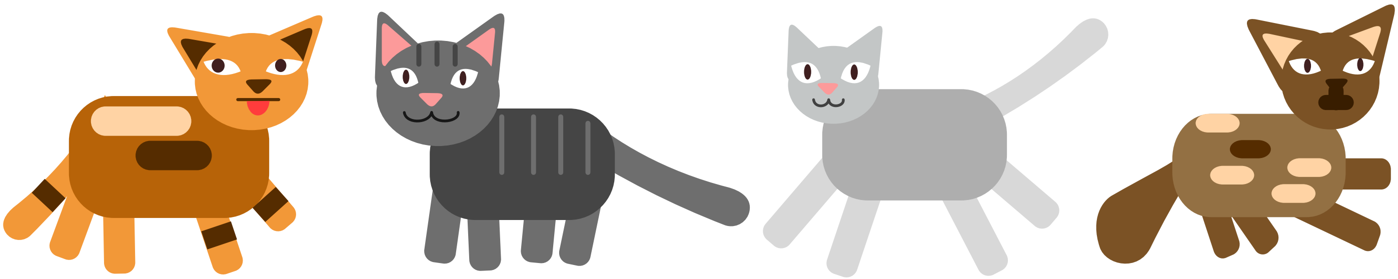


D



## Answer

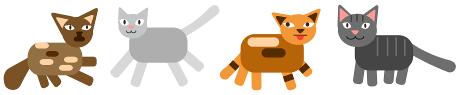
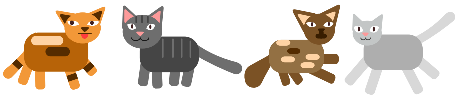
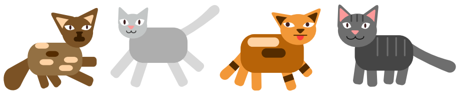
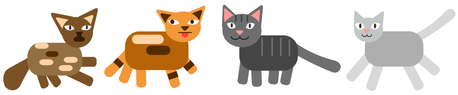
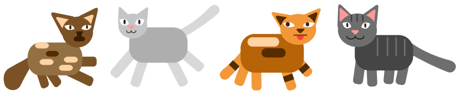
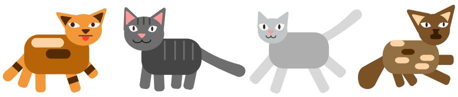
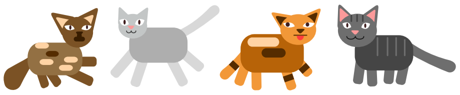
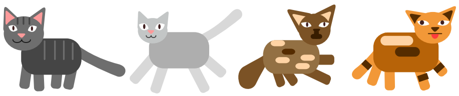
**The answer is: C**



Answer C cannot be the result of two swaps, as all four cats have changed positions. This would mean two cats had to change positions with one swap and the other two cats with the other swap. Since the two swaps deal with different cats, it doesn't matter in what order we make the swaps. So, let's consider the swap involving the first cat in the question. This cat has to swap with the fourth cat in order to end up in its position in answer C. But after the swap, the remaining three cats are still not in position. It is impossible to move the three cats with the one remaining swap, and so answer C could not have been the result of two swaps.

*Continued on next page*

# Swapping Cats – continued

| Original state | Swaps | Answer options |
|---|---|---|
|  | If the first and third cats swap and then the second and fourth cats swap, the result is answer A. |  |
|  | If the second and third cats swap and then the third and fourth cats swap, the result is answer B. |  |
|  | Answer C cannot be the result of two swaps, all four cats have changed positions. |  |
|  | If the first and fourth cats swap and then the third and fourth cats swap, the result is answer D. |  |

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

This problem is about designing algorithms using only a single operation: swap. In order to be able to properly use the operations available to them, an algorithm designer needs to fully understand the limits of those operations. This task helps to understand the limits of the swap operation when constrained to two operations. Furthermore, knowing that something cannot be accomplished by an algorithm, with or without constraints, is important in computational thinking.

The swap operation is also very common in Computer Science. If we think of the cats as data stored in the memory of a computer, then a swap involves changing the location of two pieces of data, either within the same memory or across different memories.

When you swap data in a computer, this is not as easy to do as you may think. Most computers don't just swap values, they copy values. But if you copy a value from A to B you lose the original value in B. So you have to copy that value elsewhere first.

In order to swap the values in A and B, you can execute the following steps:

- Copy the value in A to C, now C has the original value of A.
- Copy the value in B to A, overwriting the old value in A. Now A has the original value of B.
- Copy the value in C to B, now B has the old value of A.

This method of swapping two values uses a temporary variable. This method always works.

When dealing with numbers you can also swap two values without using a temporary variable. This is surprisingly complex way of doing things. Look at the following example and see if you can figure out how it works:

- Add the value in B to A, and store this in A (A = A + B).
- Subtract the value in B from A, and store this in B (B = A – B).
- Subtract the value in B from A, and store this in A (A = A – B).

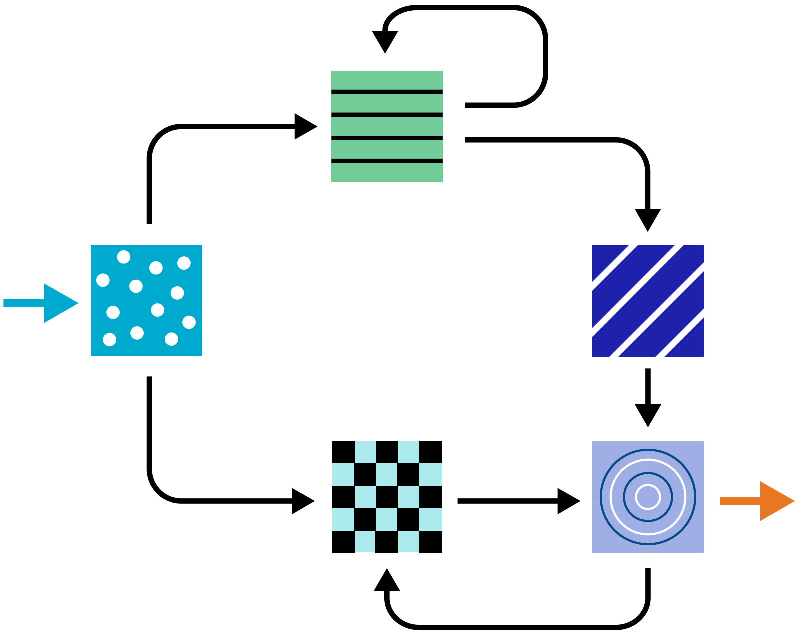
This question comes from
Lithuania

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

# Scarf Making

A machine has been created to make a scarf automatically.

The rules that the machine follows to create the pattern of the scarf is shown in the following diagram:



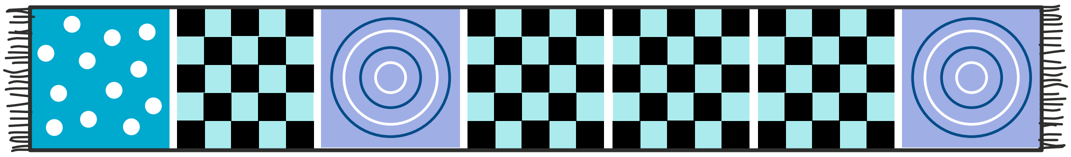
1. The machine starts a new scarf pattern by the beginning at the blue arrow on the left.
2. When the machine follows an arrow, it adds the pattern at the end of the arrow to the scarf.
3. The machine then follows one of the arrows pointing away from the pattern to the next pattern.
4. When the machine reaches the orange arrow to right, the scarf is complete.

## Question

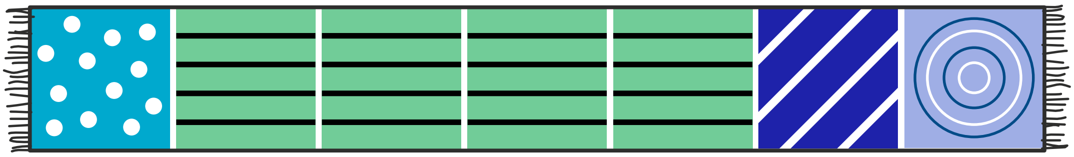
Which scarf can be made by following the above rules?

*Select one of the following.*

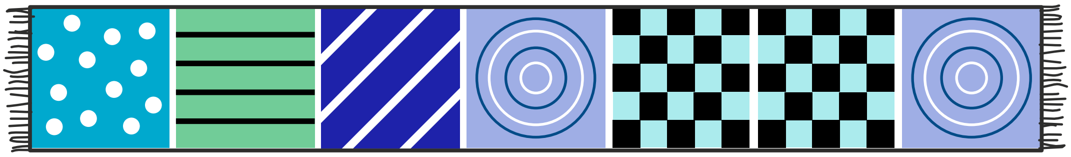
**A**



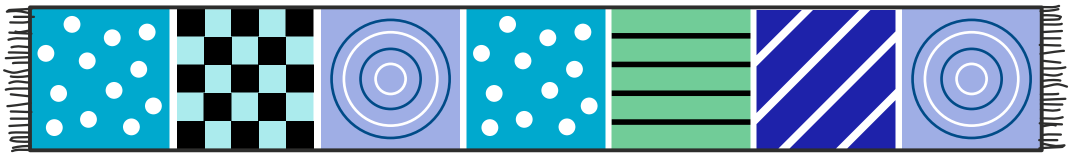
**B**



**C**



**D**



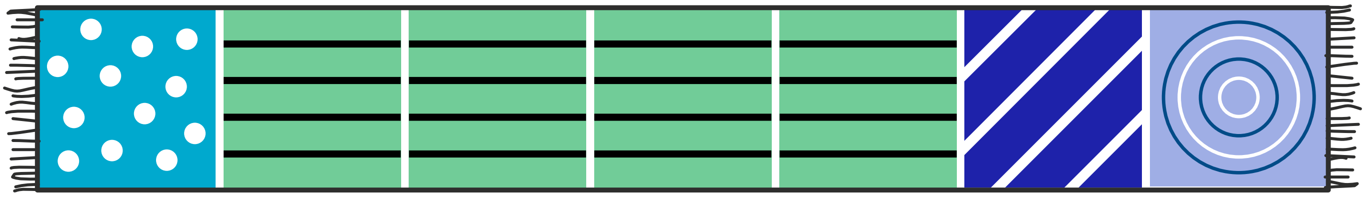
*Continued on next page*

This question comes from
Lithuania

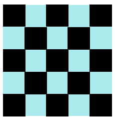
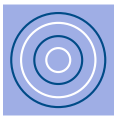
# Scarf Making – continued

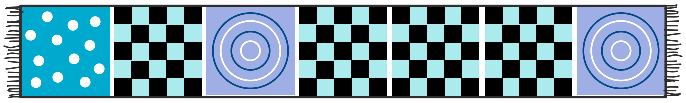
## Answer

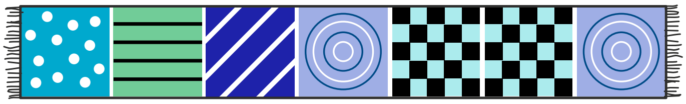
**The answer is: B**

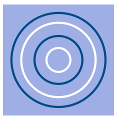
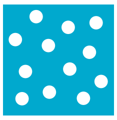
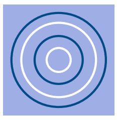
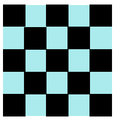


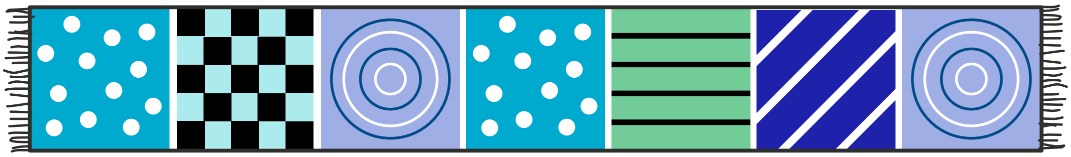
Scarves A and C have the same type of mistake:

every  in the pattern should be followed by a  .





In Scarf D, after the first  there is a  , which is impossible. After a  , we can only finish the scarf, or add a  .



## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

The schema presented in the task is the *state diagram* of a *finite-state automaton (finite-state machine or simple state machine)*.

In computing the state machine is used to describe systems that can be in a finite set of possible states and move from one state to another by means of a finite set of actions. The next state depends only on the current state and the executed action. Usually, the *initial* state and the *final* states of the system are also specified. The behaviour of the system is described by a *language accepted* by the finite-state automaton (FSA) i.e. the set of sequencesor patterns that can be obtained by following the arrows in the diagram.

This task is also a nice example of the *syntax validation process*. Before searching for logical errors in programs (debugging), the syntax of the program is checked. In this case, there are 25 different possible pattern pairs. Only seven of these pairs can be created using this machine. Each of the options (A, B, C and D) has 6 consecutive pairs of pictures. The syntax validation process checks that each pair belongs to the set of seven valid pairs. Only option B passes this check without errors.

# Bebras Challenge 2021 Round 2

## Years 5+6

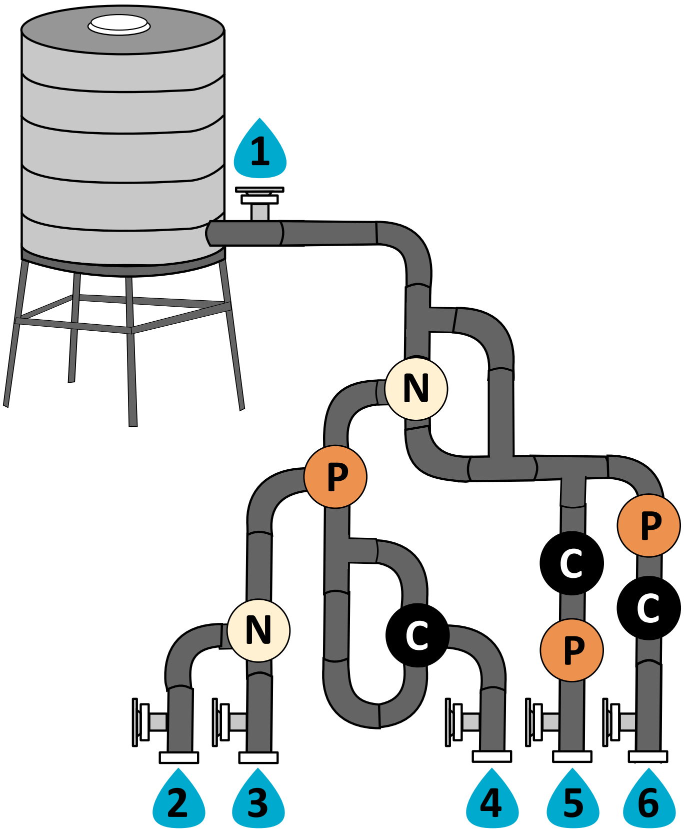
This question comes from Romania

# Water Filter

A beaver, sick of the polluted water in their city, decides to build a system of water filters to remove impurities and pollutants.

In order to be clean enough to drink, the water must pass through sand (N), gravel (P), and coal (C) filters.

It is essential that the water passes through the filters in that order.



## Question

If tap 1 is already open, which tap or taps must be opened to allow for clean water to come out?

*Select your answer from the options below.*

**Tap 4** | **Taps 4 and 6** | **Taps 3 and 5** | **Tap 6**

## Answer

**The answer is: Taps 4 and 6.**

Water through taps 2 and 3 water doesn't pass through a coal filter. For tap 5, the water can come out after passing through all the filters but not in the correct order as the coal filter is before the gravel filter on this route. For tap 4, the water passes through sand, gravel, and coal filters. Even if there is a tunnel through which the water can return, it will return to a point where it will pass through the coal filter again. Through tap 6, the water passes through sand, gravel, coal.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Collection, Data Representation, Data Interpretation, Specification, Algorithms, Implementation, Digital Systems, Interactions, Impacts

A graph is a set of nodes (intersections) with links between them. Graphs are very important in Informatics and are used to represent different networks (in this case a water filtration system). Here it is necessary to go through the graph starting from node 1 (tap 1) and passing through the different nodes so that special conditions are met (passing through filters in a certain order).

This question comes from
India

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

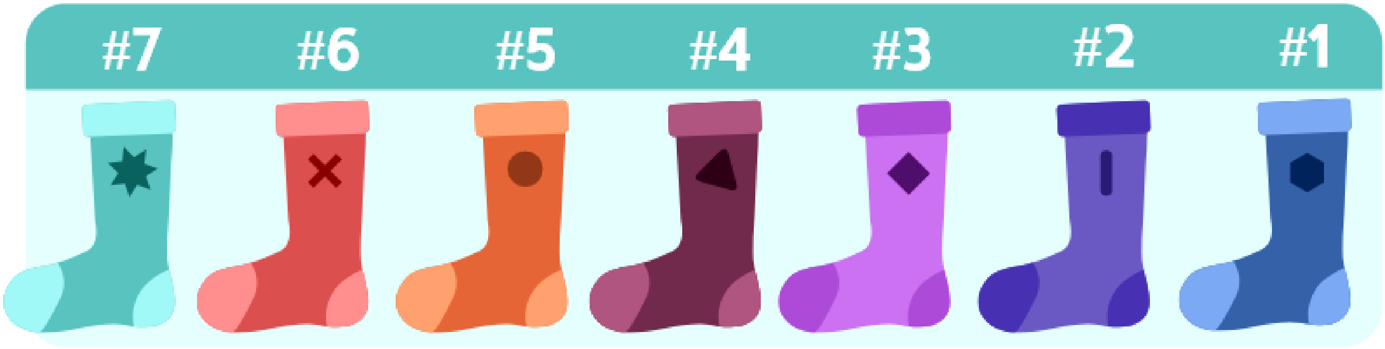
# Sock Selection

Anil likes to wear different coloured socks each day he goes out.

Anil follows the rules given below:

- Washed socks are added to the left side of the sock pile, position #7
- Socks to be worn are always taken from the right of the pile, position #1

On Wednesday, the pile of socks looks like this:

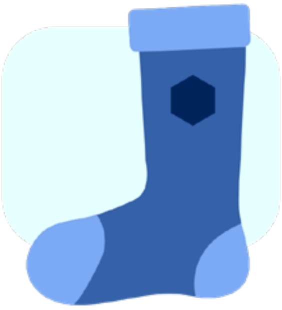


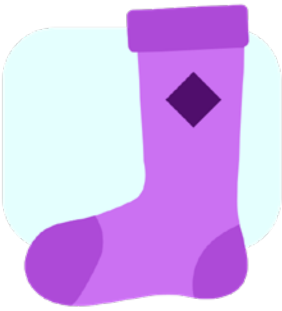
Anil will not be able to go out on Sunday this week, and Tuesday and Thursday next week.

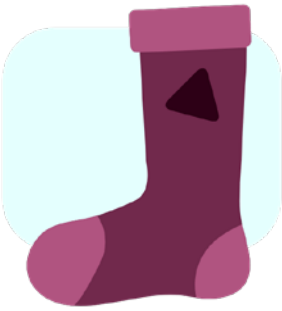
## Question

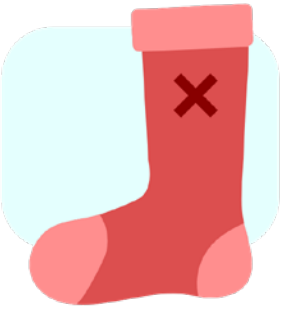
What colour socks will Anil wear next Saturday?

*Select your answer from the options below.*









*Continued on next page*

# Sock Selection – continued

## Answer

**The answer is: the blue socks with the hexagon.**

Since Anil wears socks taken from the right of the pile and he starts with the blue socks on Wednesday, the order of the socks he would wear of the rest of the week and next week are shown below (the number in brackets denotes the position as in the original picture):

| Day of the week | Sock colour [this week] | Sock colour [next week] |
|---|---|---|
| Monday | | Orange with circle (#5) |
| Tuesday | | *- Did not go out -* |
| Wednesday | Blue with hexagon (#1) | Red with cross (#6) |
| Thursday | Purple with rectangle (#2) | *- Did not go out -* |
| Friday | Lilac with diamond (#3) | Cyan with star (#7) |
| Saturday | Maroon with triangle (#4) | **Blue with hexagon (#1)** |
| Sunday | *- Did not go out -* | |

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

In computer science a queue is the data structure where data that is inserted first is taken out first – FIFO (First In First Out). Its principle is similar to standing in line for the bus. The person standing first in line boards the bus first, and a new person would join the queue at the end.

Queue data structure is one of the building blocks of computer science and is widely used. Other example applications are printer queues (a list of documents for printing), keyboard buffer (key presses are stored as queues so they appear in the order they were pressed), scheduling of CPUs etc.

This question comes from
Turkey

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

# Image Representation

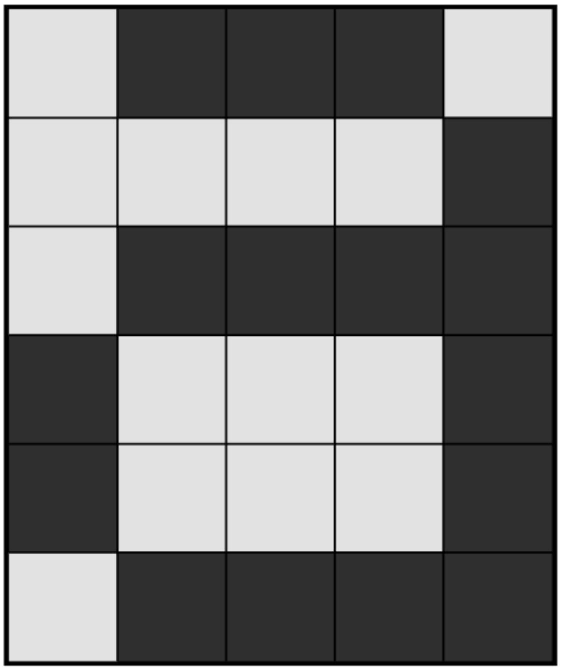
Computer screens are made up of pixels – a grid of small picture elements.

In a black and white picture, each pixel is either black or white. When a computer stores a picture, all that it needs to store is which dots are black and which are white.

It starts by stating the white pixels in the row, then the black, then white again and so forth.

## Example

The letter "a" has been magnified below to show the pixels that form the letter:
1,3,1 – 4,1 – 1,4 – 0,1,3,1 – 0,1,3,1 – 1,4

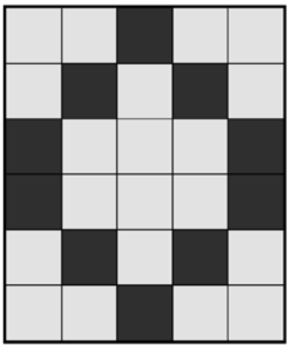


1, 3, 1
4, 1
1, 4
0, 1, 3, 1
0, 1, 3, 1
1, 4

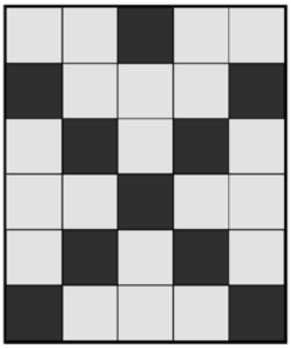
## Question

Which of the following shapes would be stored as: 2,1,2 – 0,1,3,1 – 1,1,1,1,1 – 2,1,2 – 1,1,1,1,1 – 0,1,3,1 ?

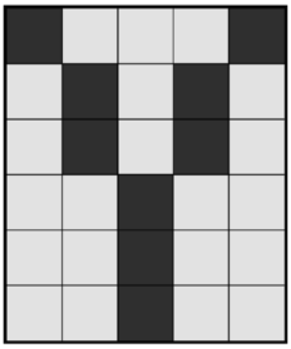
*Select your answer from the options below.*



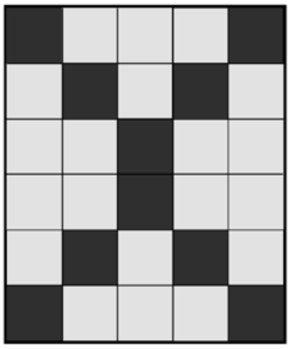
A



B



C



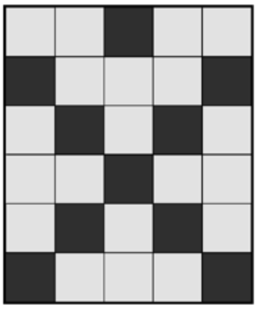
D

*Continued on next page*

# Image Representation – continued

## Answer

The answer is: B



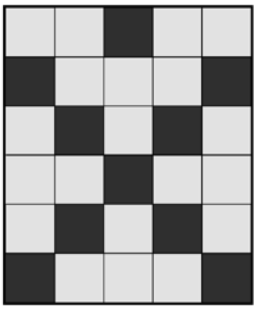
The figure in the example demonstrates that the numbers represent the number of consecutive same colored pixels (starting from white pixels) within a row from left to right.

In the above example, the first set of numbers is 1,3,1. Therefore, the first row contains one white, three black, and then one white pixels, from left to right.

We can determine what pixels each row contains, from left to right:

- **First row (2,1,2)** means 2 white, 1 black, 2 white pixels
- **Second row (0,1,3,1)** means 0 white, 1 black, 3 white, 1 black pixel
- **Third row (1,1,1,1,1)** means 1 white, 1 black, 1 white, 1 black, 1 white pixel
- **Fourth row (2,1,2)** means 2 white, 1 black, 2 white pixels
- **Fifth row (1,1,1,1,1)** means 1 white, 1 black, 1 white, 1 black, 1 white pixel
- **Sixth row (0,1,3,1)** means 0 white, 1 black, 3 white, 1 black pixel

Based on this information, we can then determine the image that is represented by
2,1,2 – 0,1,3,1 – 1,1,1,1,1 – 2,1,2 – 1,1,1,1,1 – 0,1,3,1 to be:



## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Algorithms, Evaluation

**Concepts:** Data Representation, Data Interpretation, Specification, Algorithms, Implementation, Digital Systems

In digital imaging, a pixel is a small picture element of an image. The word pixel comes from the combination of "pix" (from "pictures") and "el" (from "element"). The quantity of pixels in an image is called the image resolution. Resolution is sometimes identified by the width and height of the image as well as the total number of pixels in the image.

Typically images are encoded in some form, that is they are represented in a particular manner in the memory of computers. This particular task shows one such example of encoding, known as running-length encoding, where we omitted the values of the encoded values, since we know that we only have white and black alternating pixels.

This question comes from
Vietnam

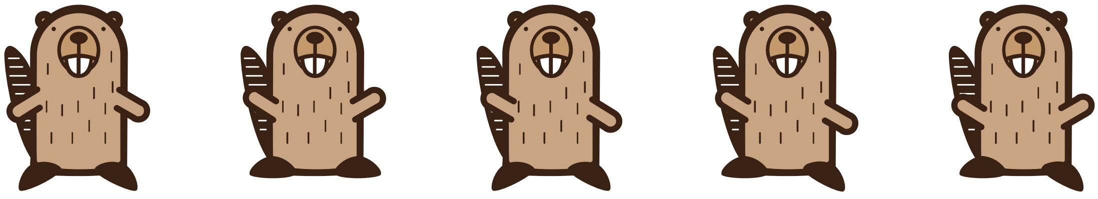
Years 3+4
Years 5+6 Easy
Years 7+8
Years 9+10
Years 11+12

# Beaver Samba

The Beaver Samba is an important cultural dance in Beaverland.

The dance has 5 positions. During each move, beavers change either the position of one leg or one arm.

Anna remembers that the dance has only 5 positions but does not remember the correct order to perform them.

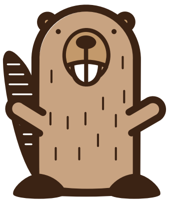


first position

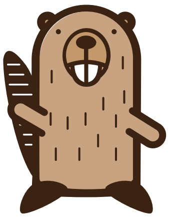
## Question

Which is the third position?

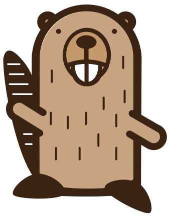
*Select your answer from the options below.*



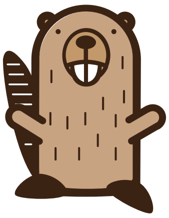
A



B



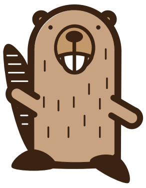
C



D

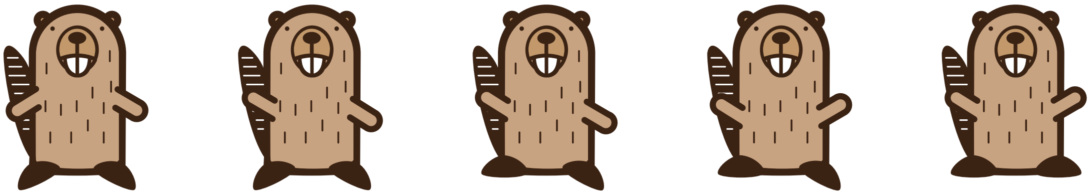
## Answer

The answer is: C



C

The following pictures show the order of the dance positions from start to finish.



## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Data Representation, Data Interpretation, Specification, Algorithms

Algorithms and programming in computers are similar to the arrangement of steps in dancing. The computer always does exactly the sequence or step by step as guided by the human user. It will always start from BEGIN and finish at the END.

# Classifier

The Beaver King wants to collect information on all the animals in his forest.

He has invited them to his castle so they can be counted. To make his task easier, the King has bought a machine that can identify the animals based on some characteristics of their faces.

We know the machine identifies some of the animals as follows:

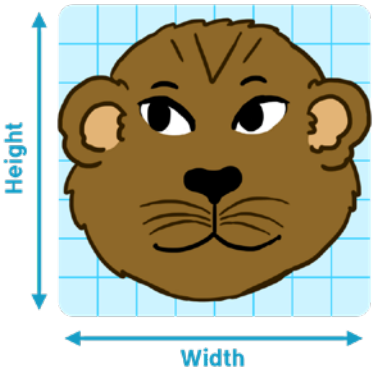
- The width of the whiskers is measured by adding both the left and the right sides.
- All measurements are the maximum width and the maximum height.

| Characteristics | Rabbit | Beaver | Bear | Cat |
|---|---|---|---|---|
| **Ear height** | half of head height | quarter of head height | quarter of head height | half of head height |
| **Whisker width** | head width | half of head width | half of head width | head width |
| **Head width** | half of head height | half of head height | head height | head height |

## Question

Identify the animal with the following face:

*Select your answer from the options below.*



**A) Rabbit** **B) Beaver**

**C) Bear** **D) Cat**

## Answer

**The answer is: c) Bear.**

The image has the characteristics of a Bear, as identified by the machine:

- The head width is equal to that of the head height
- The whisker length is a half of the head width
- The ears are a quarter of the head height

None of the other three animals have all of these characteristics simultaneously.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Digital Systems

This is a typical application of Machine Learning (Artificial Intelligence) for recognising images. Machine Learning is a very important field of Computer Science, given the multitude of applications that are being developed due to the advances in computing power.

An example of a very popular application today is Autonomous Driving, where a Machine Learning algorithm is trained to recognise the lines and signs along a road as well as all possible objects encountered by a car. In this Bebras task, we have a machine that is already trained to classify different kinds of animals and we want the students to apply a basic classification algorithm performed by the Machine Learning algorithm.

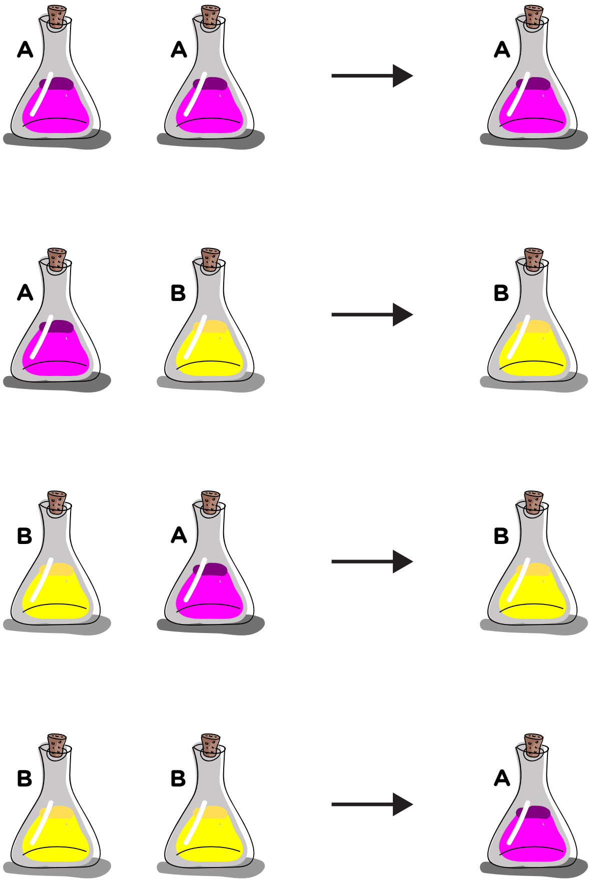
This question comes from
Pakistan

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

# Magic Potions

The wizard has two types of potions, potion A (purple) and potion B (yellow).

If he pours two bottles into a cauldron, one after the other, and he says the magic words, the potion changes colour as shown below.



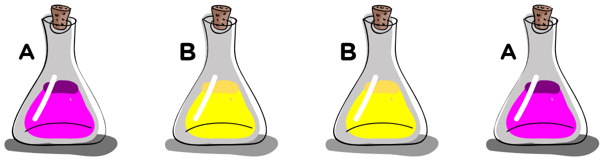
The wizard mixed exactly 4 bottles of his potions in the cauldron, resulting in a mixture that became potion B (yellow).

## Question

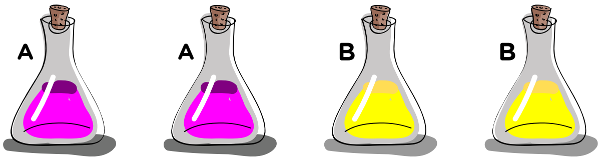
Which is the order of the potion bottles he added to the cauldron?

*Select your answer from the options below.*

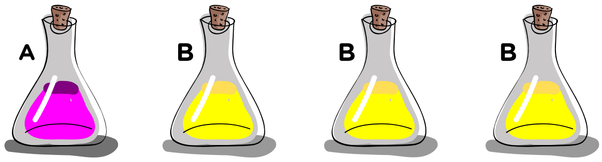
A



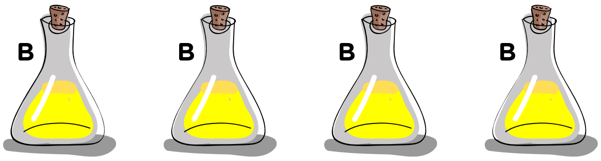
B



C



D



*Continued on next page*

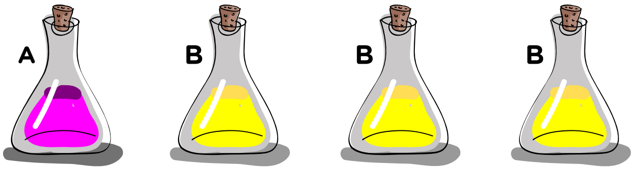
This question comes from
Pakistan

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

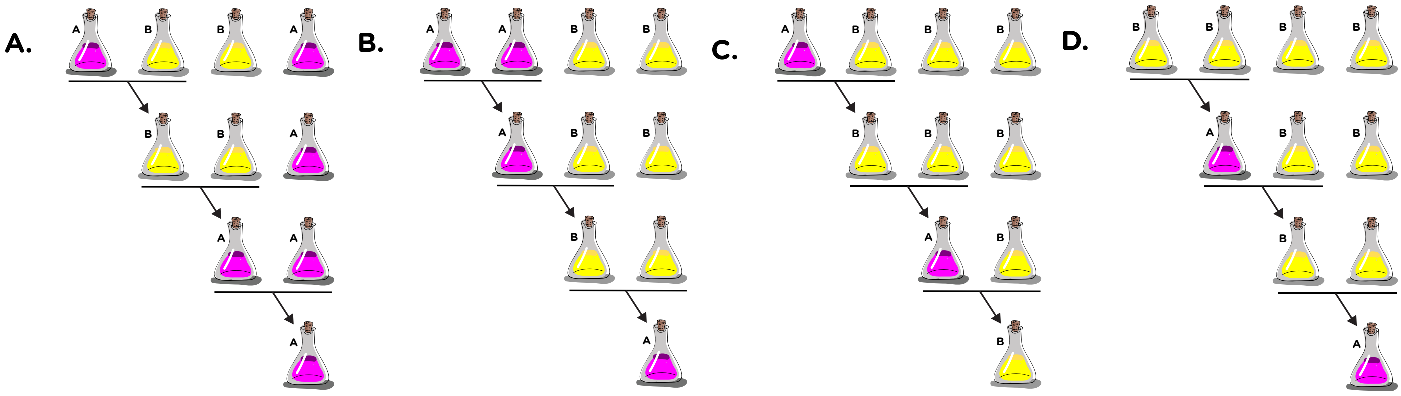
# Magic Potions – continued

## Answer

**The correct answer is: C**



If we mix the potions in given orders, we get the following results:



## It's Computational Thinking

**Computational Thinking Skills:** Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

As computers use electricity, when electricity is powering the computer, we say it is ON. When electricity not powering it, we say it is OFF.

Computer scientists usually represent those two states with the numbers 0 and 1. We call it 'binary representation'. One piece of information is called a 'bit'.

We can make operations with those bits and combine them, just as the wizard combined his potions to get a result. One of these operations is called 'logical XOR' (eXclusive OR) and it is the one presented in this task.

It works as follows:
0 XOR 0 = 0
0 XOR 1 = 1
1 XOR 0 = 1
1 XOR 1 = 0

An example of everyday use: at both ends of a staircase, there are two switches which operate the same light. Each of the switches can turn the light on or off. When both switches are up, the light is on and when both switches are down, the light is on as well. If one is up and one is down, the light is off.

Another example: My mum said to me to take a single fruit for a snack in my lunchbox. The box can contain only one fruit. There is an apple and an orange on the table. If I take nothing, I do not fulfill her wish. If I take both an apple and an orange, it does not fit into the box. So, I need to decide whether to put the apple or the orange into the box.

The XOR gate (eXclusive OR gate) is a logic gate in computers. It outputs 1 (true), when exactly one of its two inputs is 1 (true). If both of its inputs are the same, the output will be 0 (false).

This question comes from
Cyprus

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

# Chairs

The beavers are sitting on 6 chairs in a row and are given numbers from 1 to 6.

They decide to play a game. Before starting the game, a number between 1 and 4 is chosen. This decides how many positions the beavers move in each round.

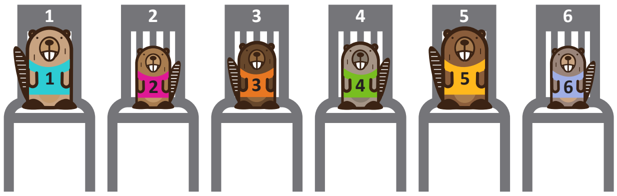
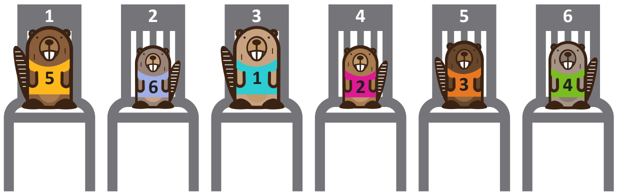
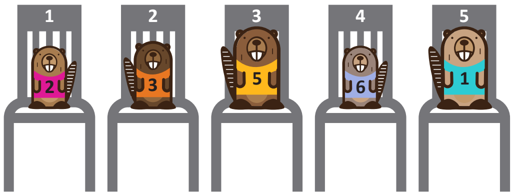
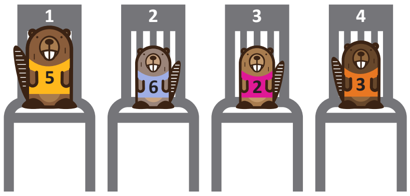
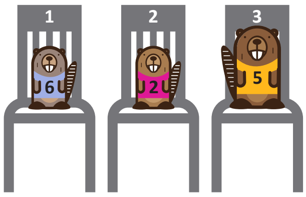
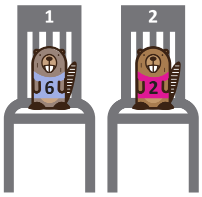
With each round, the beavers move to the right as many positions as the number chosen. The beavers at the end will move to the start of the row.

After each round, the beaver sitting furthest to the right is out of the game and last chair is removed.

The beaver left in the end is the winner.

## Example

If number 2 is chosen, beaver 6 will win as you can see below:

| | |
|---|---|
|  | Start |
|  | After the first round: beaver 4 is out. |
|  | After the second round: beaver 1 is out. |
|  | After the third round: beaver 3 is out. |
|  | After the fourth round: beaver 5 is out. |
|  | After the fifth round: beaver 2 is out. |

## Question

Which beaver would win if number 3 is chosen at the start?

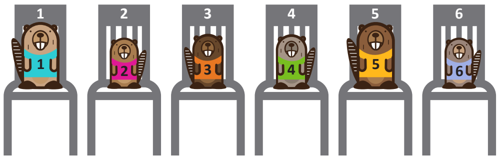
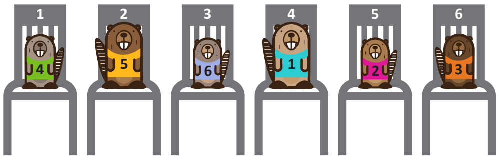
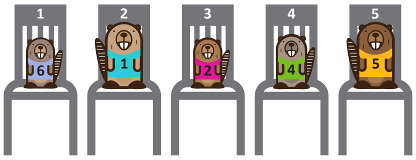
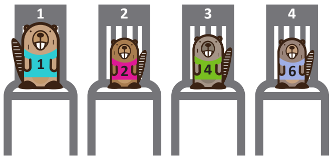
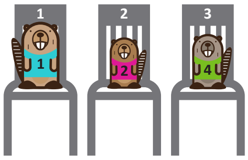
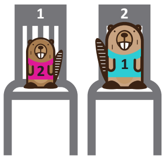
*Continued on next page*

This question comes from Cyprus

# Chairs – continued

## Answer

**The answer is: beaver 2**

| | |
|---|---|
|  | Start |
|  | After the first round: beaver 3 is out. |
|  | After the second round: beaver 5 is out. |
|  | After the third round: beaver 6 is out. |
|  | After the fourth round: beaver 4 is out. |
|  | After the fifth round: beaver 1 is out. |

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

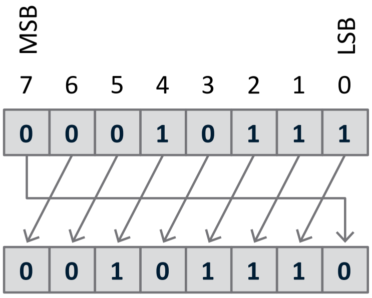
In computer programming, a bitwise operation operates on one or more bit patterns or binary numerals at the level of their individual bits.

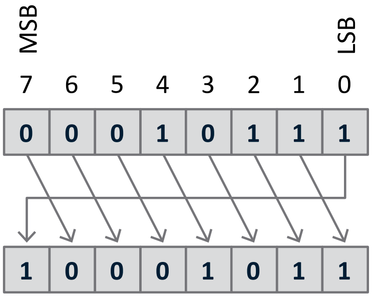
It is a fast and simple action, directly supported by the processor, and is used to manipulate values for comparisons and calculations.

One form of bitwise operation is Circular Shift. In this operation, sometimes called rotate no carry, the bits are "rotated" as if the left and right ends of the register were joined.

The value that is shifted into the right during a left-shift is whatever value was shifted out on the left, and vice versa for a right-shift operation.

This is useful if it is necessary to retain all the existing bits, and is frequently used in digital cryptography.



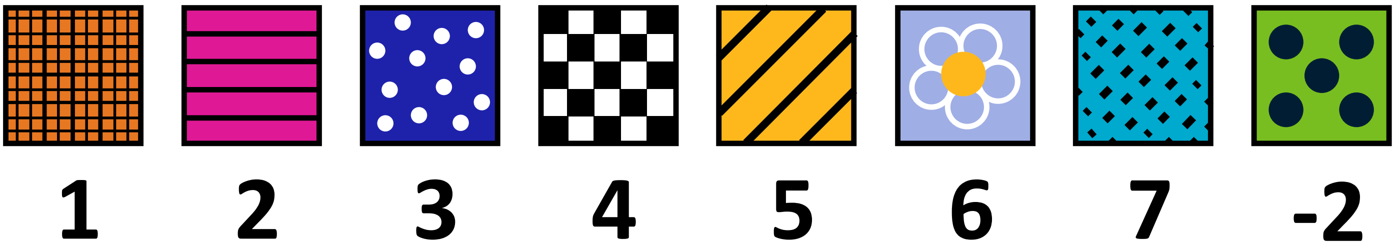


This question comes from
Japan

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

# Tile Laying

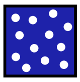
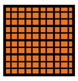
Beaver Bitaro is laying down eight different types of tiles, as shown below.

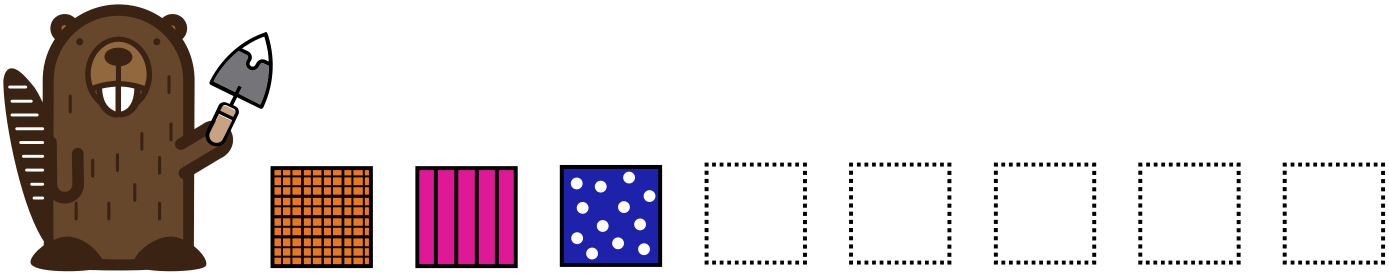


He must put them down in a sequence from left to right according to certain rules:

- The first tile is always type 1, the second tile is always type 2.
- The next tile is determined by adding the number of the two previous tiles. If that number is greater than 7, then a green tile with the number –2 is used instead.

## Example

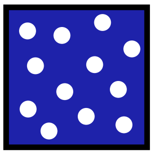
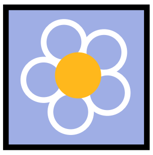
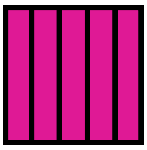
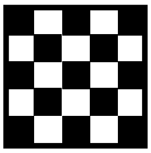
The third position will be a dark blue tile  number 3 because 1 (tile 1 ) plus 2 (tile 2 ) equals 3.



## Question

Which tile must Bitaro lay in the eighth position?

*Select your answer from the options below.*

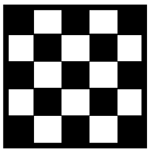





*Continued on next page*

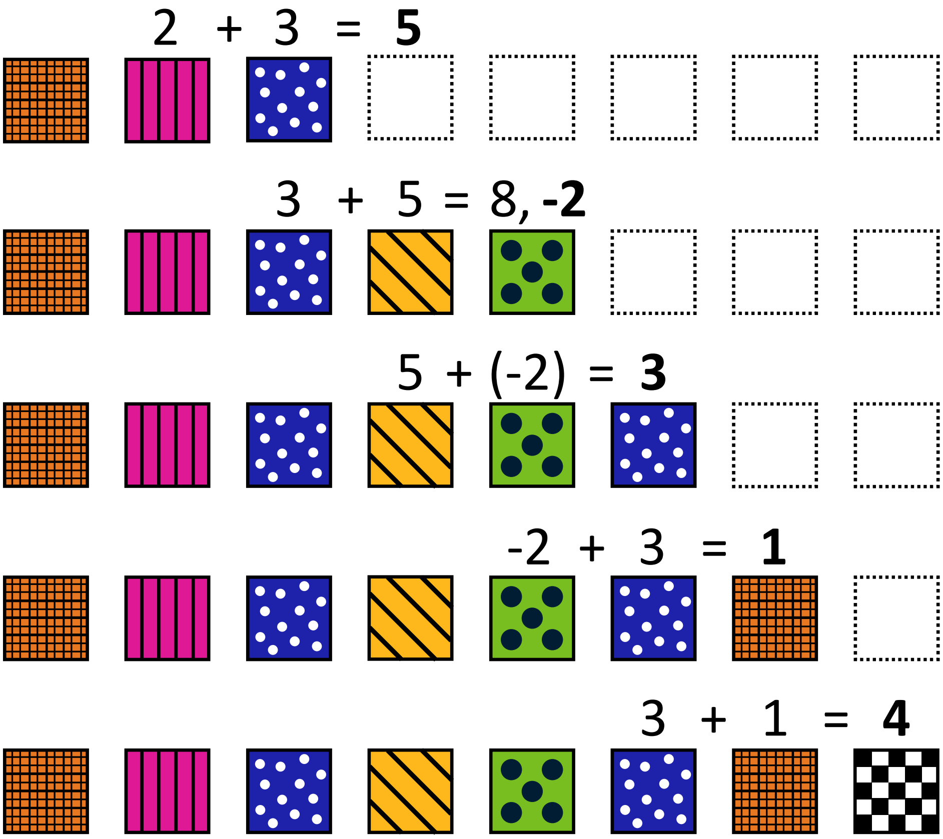
# Tile Laying – continued

## Answer

**The answer is: tile 4**



This figure explains line by line how the next tile type was determined, starting from the fourth tile:



## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

The rule in this task uses the value of the previous two tiles to determine the value of the next tile. Such a definition is called *recursive*: it essentially 'uses itself'.

In mathematics for instance, the *Fibonacci sequence* is defined in a similar way as the next tile is determined here: by adding up the two previous values of the sequence to determine the next one.

Unlike the tiles here, where we have the special –2 green tile, Fibonacci numbers never stop growing. They can be found surprisingly often in real life, as in the way the branches appear in a tree, the sections on the skin of a pineapple, etc.

In computer science, recursion is a widely used programming technique and is useful when a problem can be deconstructed into smaller problems that share the same characteristics with the original problem.

This question comes from
Switzerland

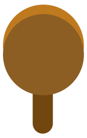
Years 3+4
Years 5+6 Medium
Years 7+8
Years 9+10
Years 11+12

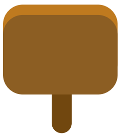
# Tree Sudoku

Beavers planted 9 trees in a 3 x 3 field. The trees had three different heights:

1 , 2 , and 3 .

- In each line (horizontal or vertical), there is exactly one tree of each height.
- No line has two trees of the same height.
- The beavers observed how many trees they could see from every position where there was a sign.
- They then wrote the number on the sign.

 signs for vertical lines.

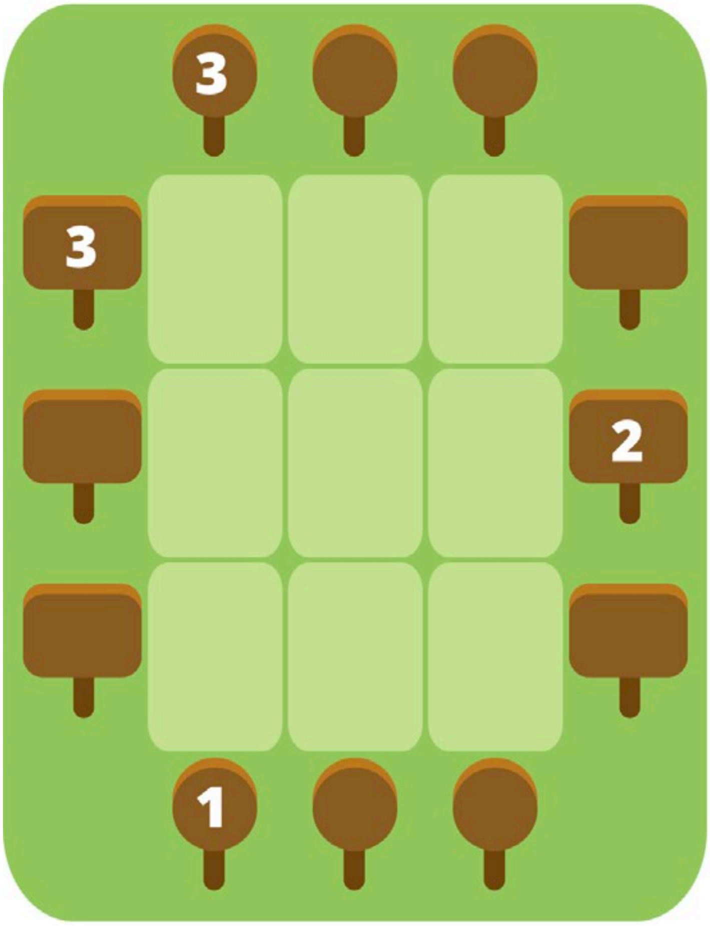
 signs for horizontal lines.

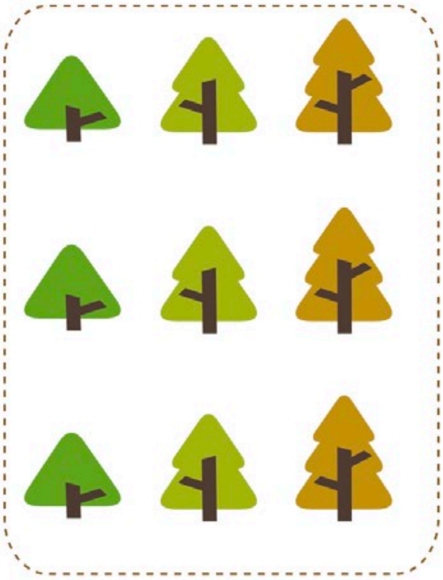
However, when looking along a line of trees, the beavers could not see smaller trees that were planted behind taller trees!

Unfortunately, the rain washed away some of the numbers on the signs.

## Question

Which trees were planted where?





*Continued on next page*

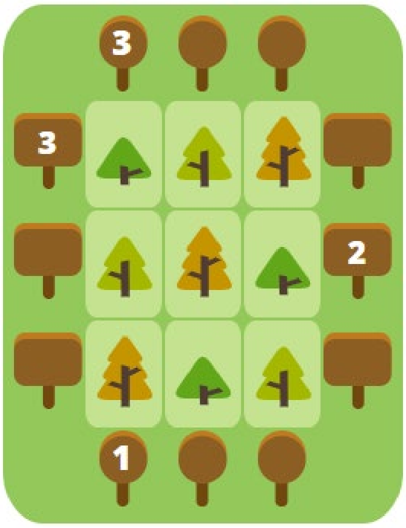
This question comes from
Switzerland

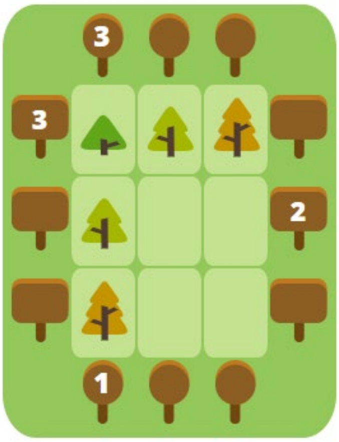
Years 3+4
Years 5+6 Medium
Years 7+8
Years 9+10
Years 11+12

# Tree Sudoku – continued

## Answer

**The answer is:**





There is only one solution. The two signs showing "3" are sufficient to reconstruct the whole field.

If a beaver can see all trees from a position, the trees must be ordered by increasing height 🌲 🌲 🌲 away from their position.

We are able to fix the first row and the first column using this information.

The remaining trees can be placed according to the rules of Sudoku.

As you can only have one tree of each height in each row or column, there is only one way to place the remaining trees.

The position in the middle row at the right hand end must contain a 1 🌲 because there is a 2 🌲 on the third row and the third column has a 3 🌲 already.

We conclude that the bottom right must contain a 2 🌲 because the trees 🌲 and 🌲 are already present in the third coulmn, and so on.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Implementation

This task addresses two fundamental competencies in Computer Science:

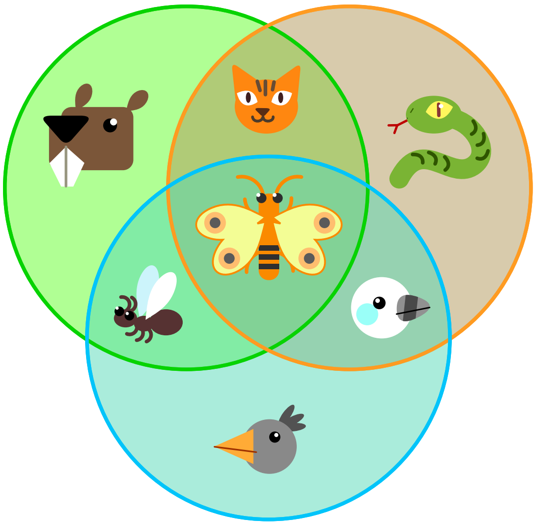
1. To find a solution for a problem that satisfies a given constraint.
2. The ability to reconstruct an object from partial information using knowledge of the properties of said object.

This can be used for a compressed representation of objects. The term lossless compression is used in Computer Science to describe this process as the user does not lose any information that had been compressed when the information is uncompressed.

This question comes from
the United States

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

# Zoo Animals



The following diagram shows three circles, each circle shows animals with different characteristics. One circle shows animals with stripes, one circle shows animals with more than two legs, and one circle shows animals with wings.

If an animals has wings and stripes then it will be placed where both those circles overlap.

Your friend has selected two types of animals that she likes:

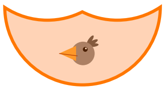
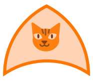
- I like winged animals that don’t have stripes or many legs.
- I also like animals without wings and that have stripes and many legs.

## Question

Which animals does your friend like?

## Answer

**The answer is: the cat and the grey bird.**

- Winged animals that don’t have stripes or many legs: the grey bird, as shown 
- Animals without wings that have stripes and many legs: cat, as shown 

None of the other animals are possible because they don’t meet all the requirements of either rule:

| | Wings | No stripes | Not many legs | No wings | Stripes | Many legs |
|---|---|---|---|---|---|---|
| Beaver | × | ✓ | × | × | × | ✓ |
| Cat | × | × | × | ✓ | ✓ | ✓ |
| Snake | × | × | ✓ | ✓ | ✓ | × |
| Butterfly | ✓ | × | × | × | ✓ | ✓ |
| Fly | ✓ | ✓ | × | × | × | ✓ |
| White bird | ✓ | × | ✓ | × | ✓ | × |
| Grey bird | ✓ | ✓ | ✓ | × | × | × |

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Algorithms, Evaluation

**Concepts:** Data Representation, Data Interpretation, Specification, Algorithms

Logical reasoning is an important skill within Computer Science.

In this task the student is presented with a logical representation of information, and then has to deduce from two statements which animals are being described by these statements.

Each animal has the value “True” or False” for each of the three properties in the diagram (stripes, legs, wings). A butterfly has the value “True” for each property which is why it is at the centre of the Venn diagram.

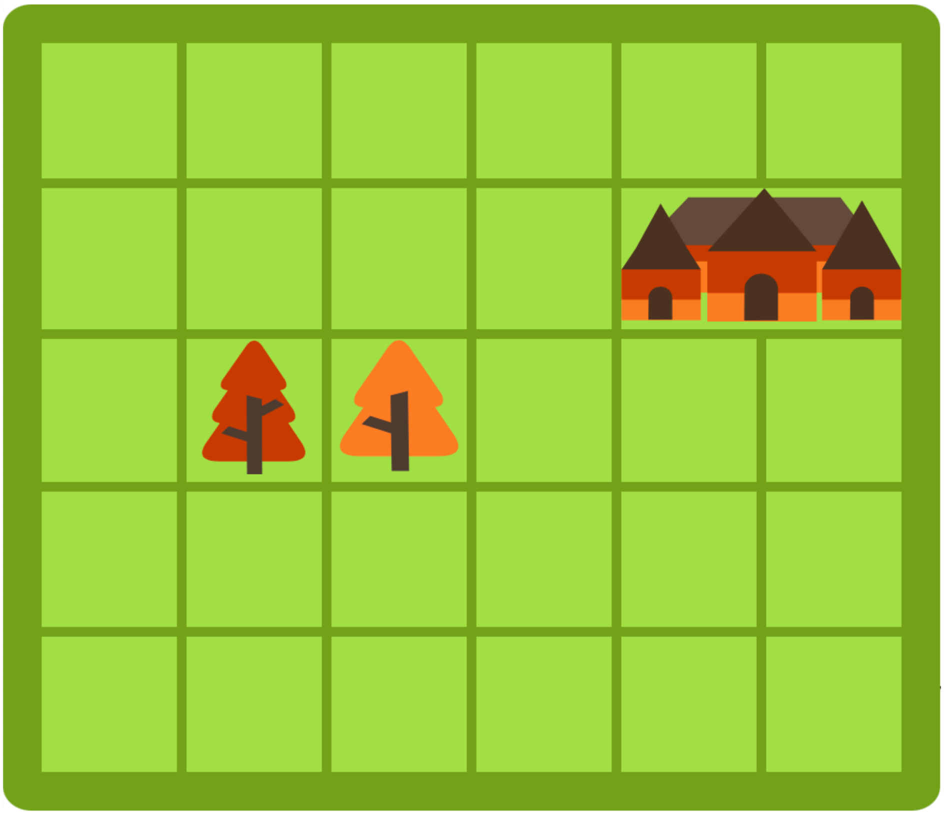
We can combine “True” and “False” values in special ways using logical operations. This task is a special form of combining the values of these properties with AND and OR logical operations. Each rule from the friend consists of properties combined using AND. But the different rules are combined using OR.

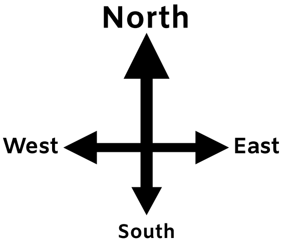
# Planting Flowers

Sabrina the flower planting robot is programmed to plant as many plants as possible in the garden below obeying certain rules.

Rules:

- Plants must NOT be planted in a square directly north of a building or tree.
- Plants with star-shaped, red flowers must be planted in lines of 3 or more (not diagonally).
- Plants with round, yellow flowers must be planted in groups of 4 in a 2 x 2 square.
- There must be a gap of at least one square between each group of plants with round, yellow flowers.
- There must be an equal number of plants with star-shaped and round flowers.





## Question

What is the largest number of plants Sabrina can plant?

*Continued on next page*

# Planting Flowers – continued

## Answer

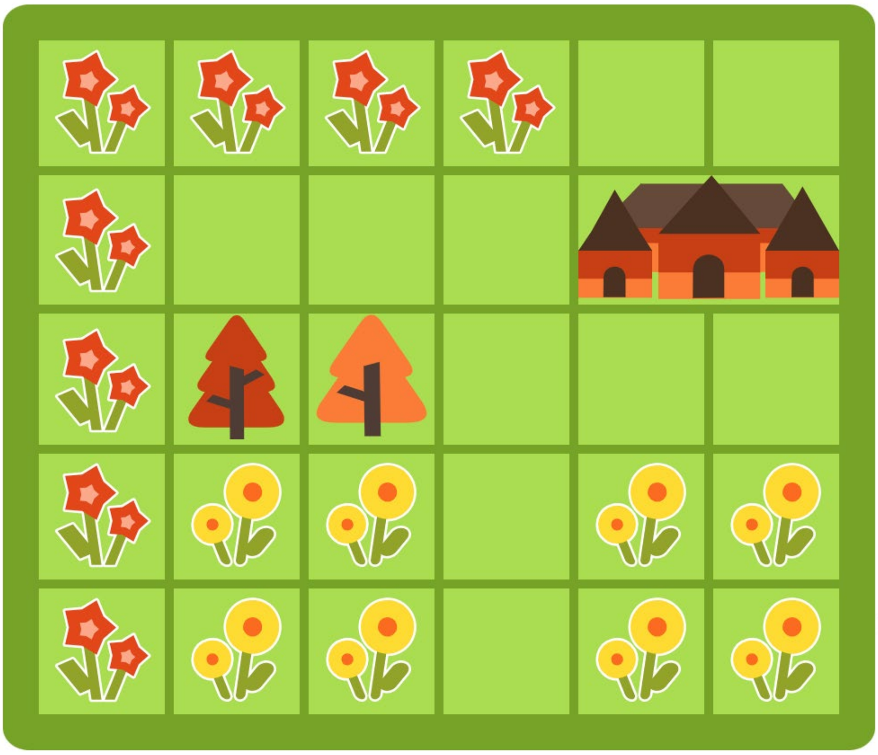
**The answer is: 16**

The key to this problem is realising that there is going to be a limit on the number of groups of round, yellow flowers.

In fact there it is only possible to place two groups of four when the rules are applied. It then becomes easy to find spaces to plant 8 star-shaped, red flowers to obey the rules.

Two sets of 8 gives us 16 flowers as the maximum allowed.

One possible arrangement:



## It's Computational Thinking

**Computational Thinking Skills:** Pattern Recognition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms

Logic is the foundation of many aspects of Computer Science. This problem can be solved with logical reasoning.

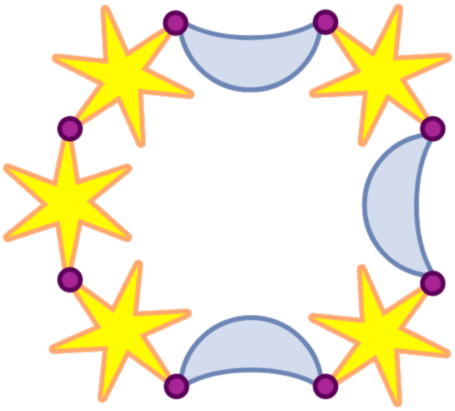
Here we look at the rules and decide which is going to be the most limiting. Finding a partial solution that satisfies the most limiting factor enables us to solve the whole problem more easily.

Many classic games are based on this kind of logic, such as, chess where each type of chess piece can only move according to its own set of rules.

The concept of this task is also related to a maximisation problem of job assignment with constraints.

# Stars and Moons

Marie would like a bracelet like the one shown on the right.
She gives Stephan the following instructions:



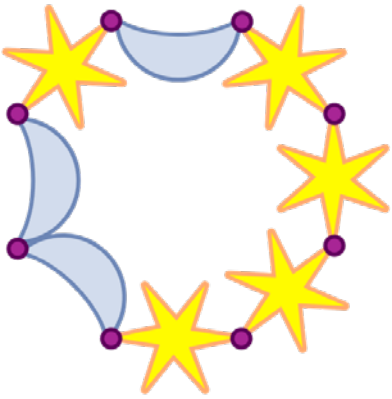
1. Join a star piece and a moon piece.
2. Repeat step 1 two more times.
3. Join the three parts made so far into a single chain.
4. Add two more stars at one end of the chain.
5. Join both ends of the chain to make a bracelet.

Unfortunately, if Stephan does not have a picture of the bracelet, he may end up with a bracelet that looks quite different, even if he follows the instructions exactly.

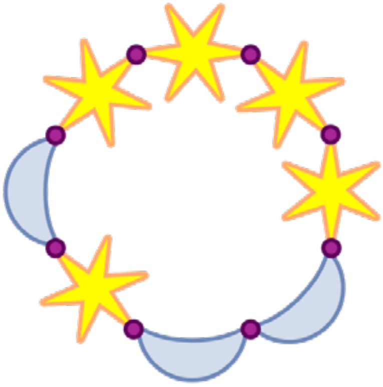
## Question

Which of the bracelets shown below can NOT be made by following Marie's instructions?

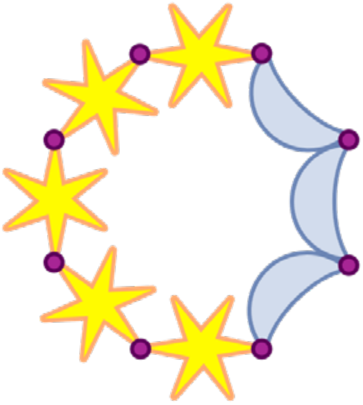
*Select your answer from the options below.*



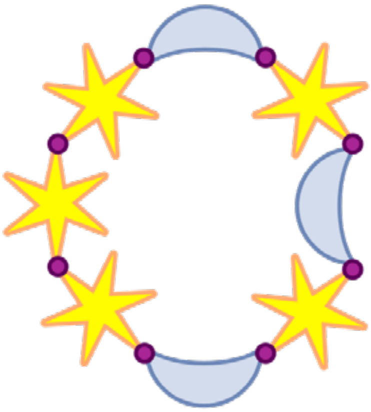
A



B



C



D

*Continued on next page*

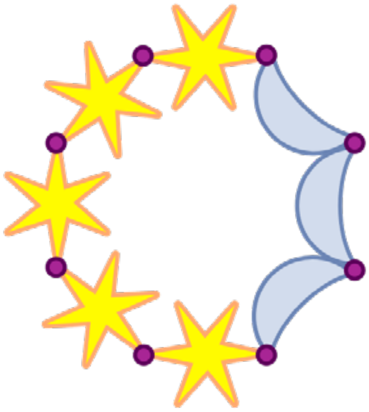
This question comes from
Belgium

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

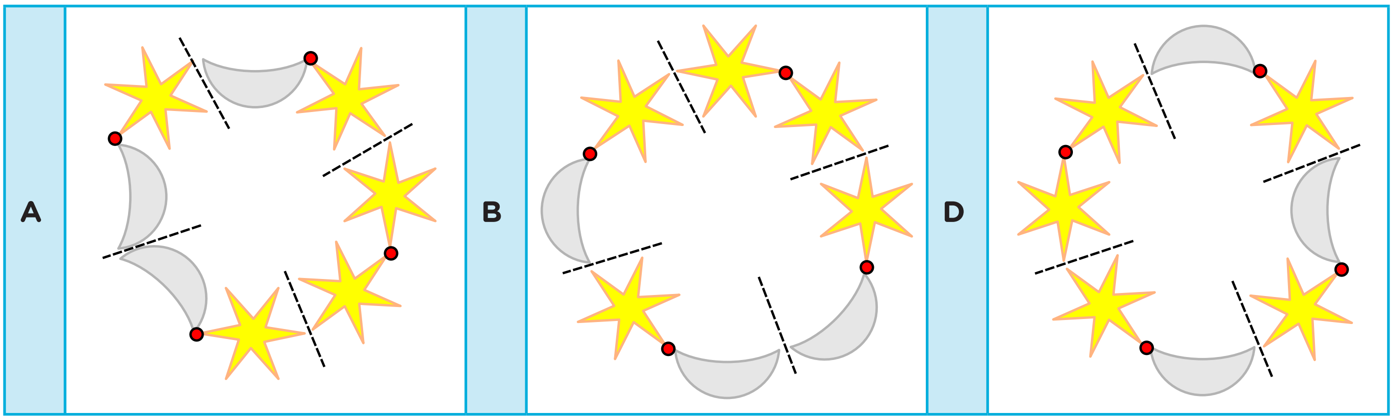
# Stars and Moons – continued

## Answer

**The answer is: C**



The pictures below show how each of the other three examples can be split into three star-moon pairs and one star-star pair.



Because every moon in the bracelet must be next to a star (according to the first instruction), it is not possible to have 3 moons in a row if you follow the instructions. Bracelet C has three moons in a row.

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Algorithms, Evaluation

**Concepts:** Data Representation, Data Interpretation, Specification, Algorithms, Implementation, Digital Systems

When programmers give instructions to a computer, it is important that they exactly specify what the computer has to do, otherwise the result may not be what is wanted.

For example, in the list of Mary’s instructions, she forgot to state exactly how the three star-moon pairs must be joined together. In the bracelet she wanted, a moon is always surrounded by stars. So, although the instructions looked quite specific, something was missing. If a computer was driving the bracelet making machine, Mary’s instructions would not have been sufficient.

Luckily, in reality computers will usually just stop and say ‘I have no idea what you mean’ if your instructions are not sufficiently clear. Always having to write everything out in detail becomes tedious and yields very long programs. Therefore computer program instructions provide several types of shortcuts.

For instance, like in Mary’s instructions, it is possible to say that the computer has to repeat an action several times, instead of having to copy the instructions for these actions over and over again. But, this can be done in very specific way, in order to be understood by the computer.

This question comes from
Austria

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

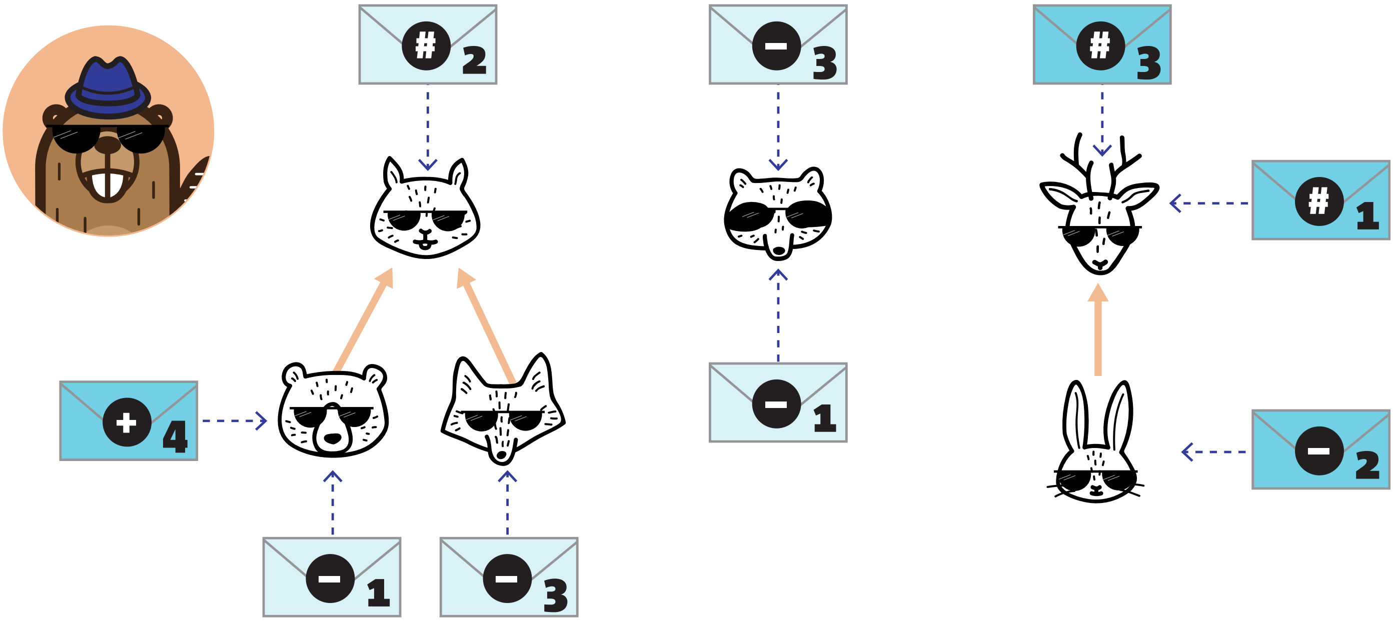
# Secret Message

Big Beaver Boss has a secret message for his six agents. Unfortunately, Big Beaver Boss does not trust his agents equally.

He split the message into four pieces (1 to 4) and added levels of secrecy to each piece.

- A piece of the message sealed with the symbol **–** can only be read by the agent who receives the piece.
- A piece of the message sealed with the symbol **#** can be read by the agent that receives it and their team
- A piece of the message sealed with the symbol **+** can be read by every agent.

The following picture shows the agents and the pieces of the message they have received. The teams are shown underneath their boss, the orange arrows point to the head agent.



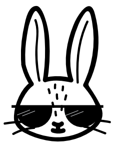
## Question

Which agent has access to all four parts of the message?

*Select your answer from the options below.*

Raccoon

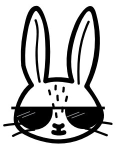
Squirrel


Rabbit

Bear

## Answer

**The answer is: Rabbit**



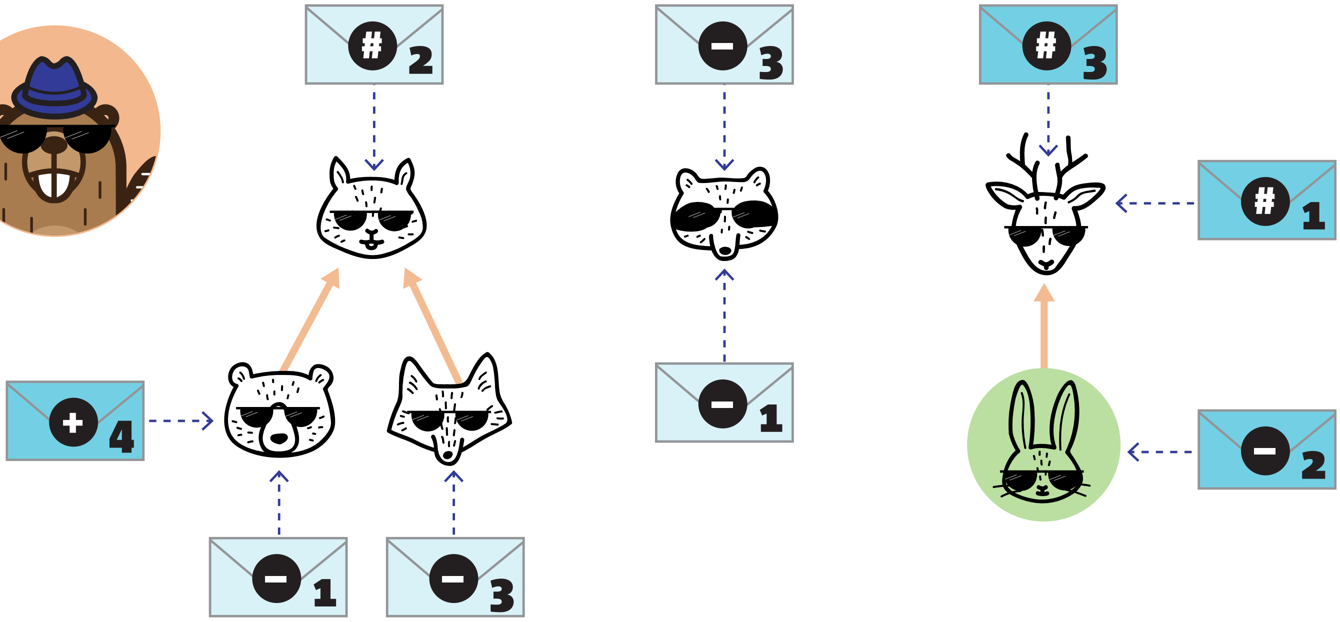
Rabbit received the second piece of the message.

Her boss, Deer, received pieces 1 and 3 with the # symbol on the seal – meaning these messages could be read by Rabbit too.

Piece 4 of the message is visible to all agents because it is sealed with the + symbol.

*Continued on next page*

# Secret Message – continued



Squirrel is wrong because she cannot read pieces 1 and 3 of the message.

Fox is wrong because he cannot read piece 1 of the message.

Bear is wrong because he cannot read piece 3 of the message.

Raccoon is wrong because he cannot read piece 2 of the message.

## It’s Computational Thinking

**Computational Thinking Skills:** Decomposition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Digital Systems, Interactions

Most object-oriented programming languages are class-based, meaning that objects are instances of classes.

In this task, the different agents represent different classes. Every class has access to the relevant objects with the corresponding type of visibility. **Public** (+) objects can be accessed from every class in the program. **Private** (–) objects are only visible and changeable by the class itself. **Protected** (#) objects can be accessed by the class itself and by its subclasses.

In this example: the subordinates can read # messages sent to their bosses. Different types of visibility are very useful: a class does not have to deal with objects and properties from other classes. In this way, some programming errors can be avoided.

This question comes from China

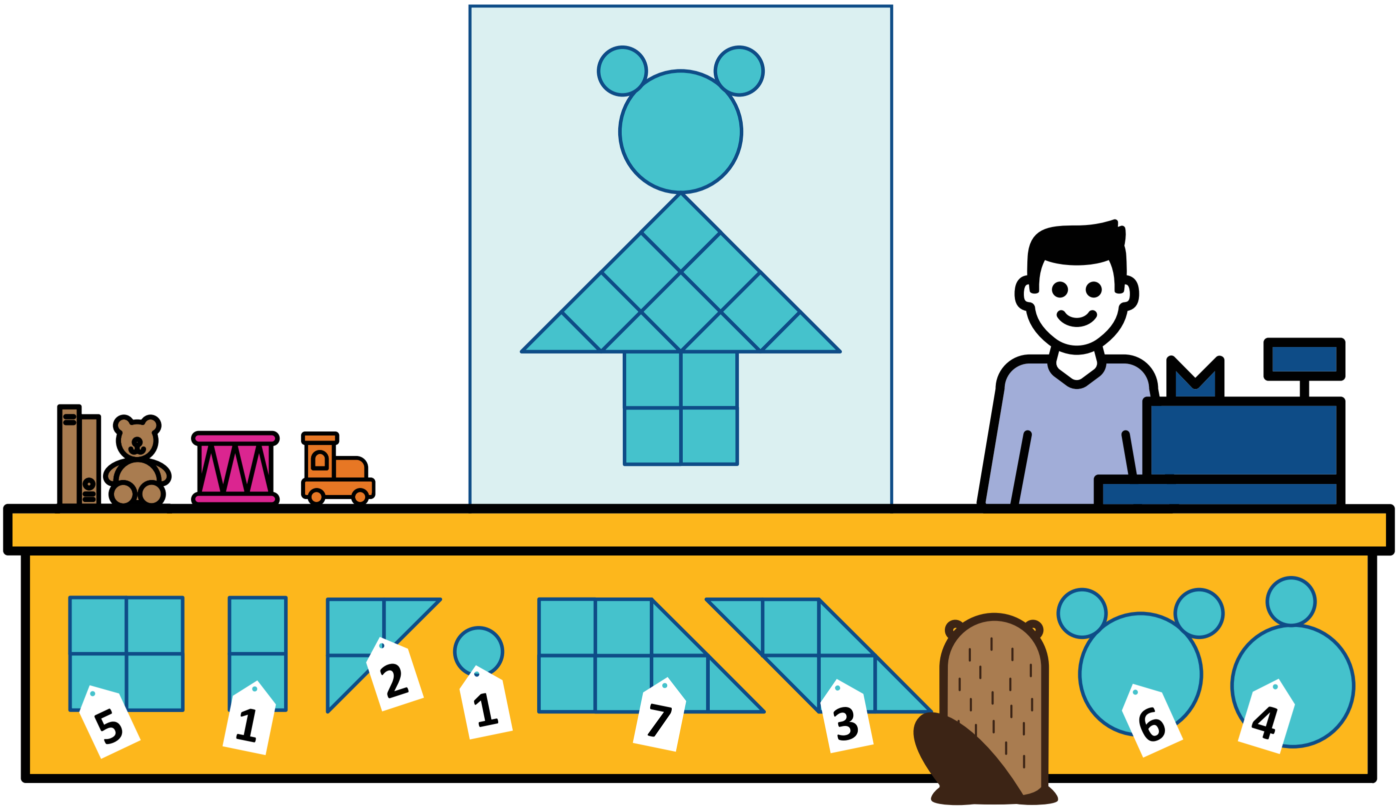
Years 3+4
Years 5+6 Hard
Years 7+8
Years 9+10
Years 11+12

# Jigsaw Puzzle

Jamal wants to assemble a toy with the same form as the one in the picture on the toy shop wall.

He needs to make the toy from the shapes offered on the counter, each with a different price, ranging from 1 coin to 7 coins.

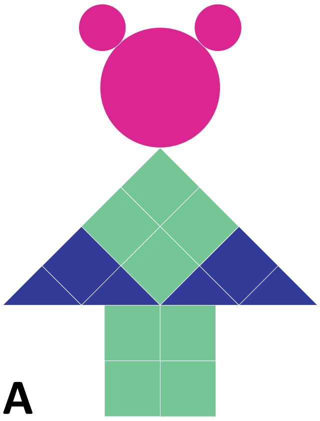
He can buy as many shapes as he wants of each shape type and rotate them in any way.

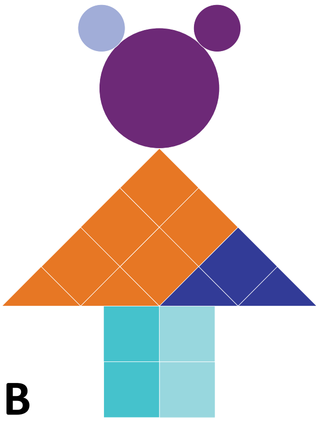


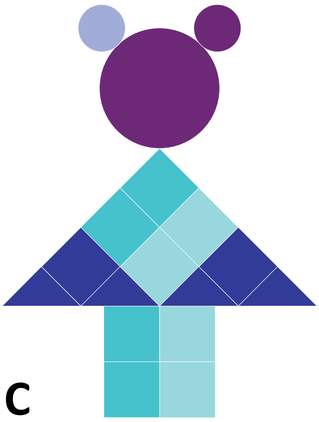
## Question

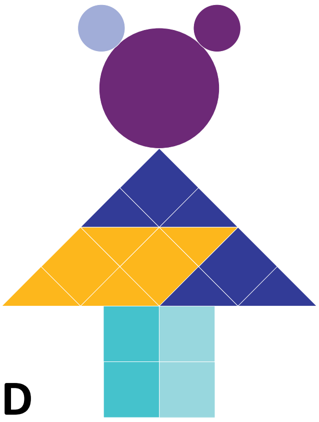
Which combination of pieces will allow Jamal to build the toy for the least amount of coins?

*Select your answer from the options below.*


A


B

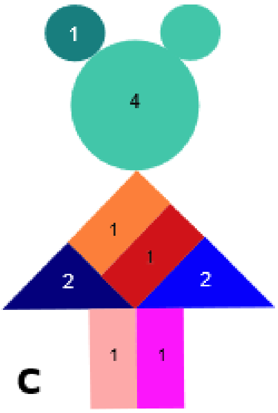

C


D

*Continued on next page*

# Jigsaw Puzzle – continued

## Answer



**The answer is: c) 13 coins**

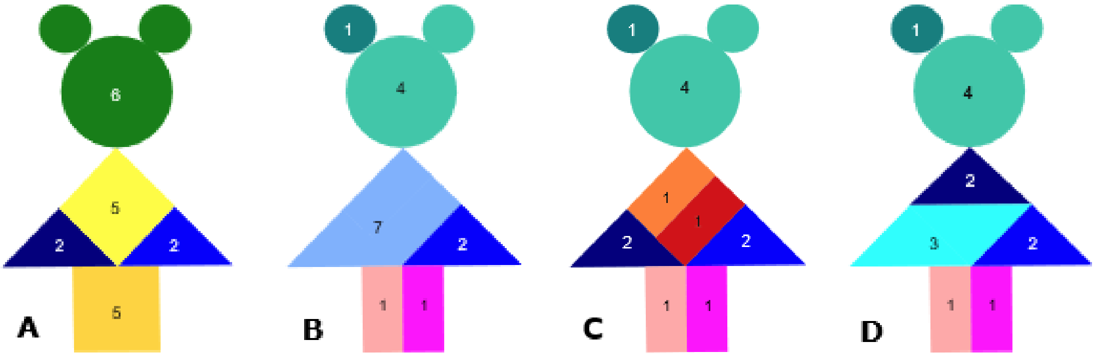
To find the correct answer, first we break down a complex problem into smaller parts. We start by assigning the shapes that potentially make up the figure of the toy to two fundamental categories: they are either angular or rounded.

On the basis of this, the figure can be broken up into two main parts: the head, only containing rounded shapes, and the body (torso and legs), consisting of angular shapes. The body can then be further subdivided into two parts: the torso, which may contain a square, rectangles, triangles or parallelograms, and the legs, which may only consist of rectangles or a square.

The next step is to determine the minimum amount of beaver coins needed for each part.

- The head is the easiest part: using the shape identical with the head (6 coins) is more expensive than using a combination of the two other rounded shapes (5 coins in total).
- The legs are similarly easy to determine: using two rectangular shapes (2 coins in total) for that part is far cheaper than using one square shape (5 coins).
- The most difficult part is the torso because there are many possible combinations of shapes. The best way to proceed is to use the cheaper shape types as much as possible, trying to built the torso from them.
- Starting with the rectangular shape type for the middle of the torso and complementing the torso with the two triangles.

Following this logic, we eventually arrive at the least expensive combination, which is shown in figure C.



All the other answer options also represent possible combinations of puzzle pieces as illustrated by figures A (20 coins), B (16 coins) and D (14 coins) respectively. The total cost of each of them would be higher, however, than the total cost of option C. So C is the most cost effective option.

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Pattern Recognition, Abstraction, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Digital Systems, Interactions, Impacts

For solving this task you can use the divide and conquer algorithm, which means that you are repeatedly breaking down a problem into two or more sub-problems of the same or a related type until these become simple enough to be solved directly.

The solutions to the sub-problems are then combined to give a solution to the original problem.

The only proven strategy for solving tangram problems is trial and error – rearranging the shapes in multiple combinations until the answer is reached.

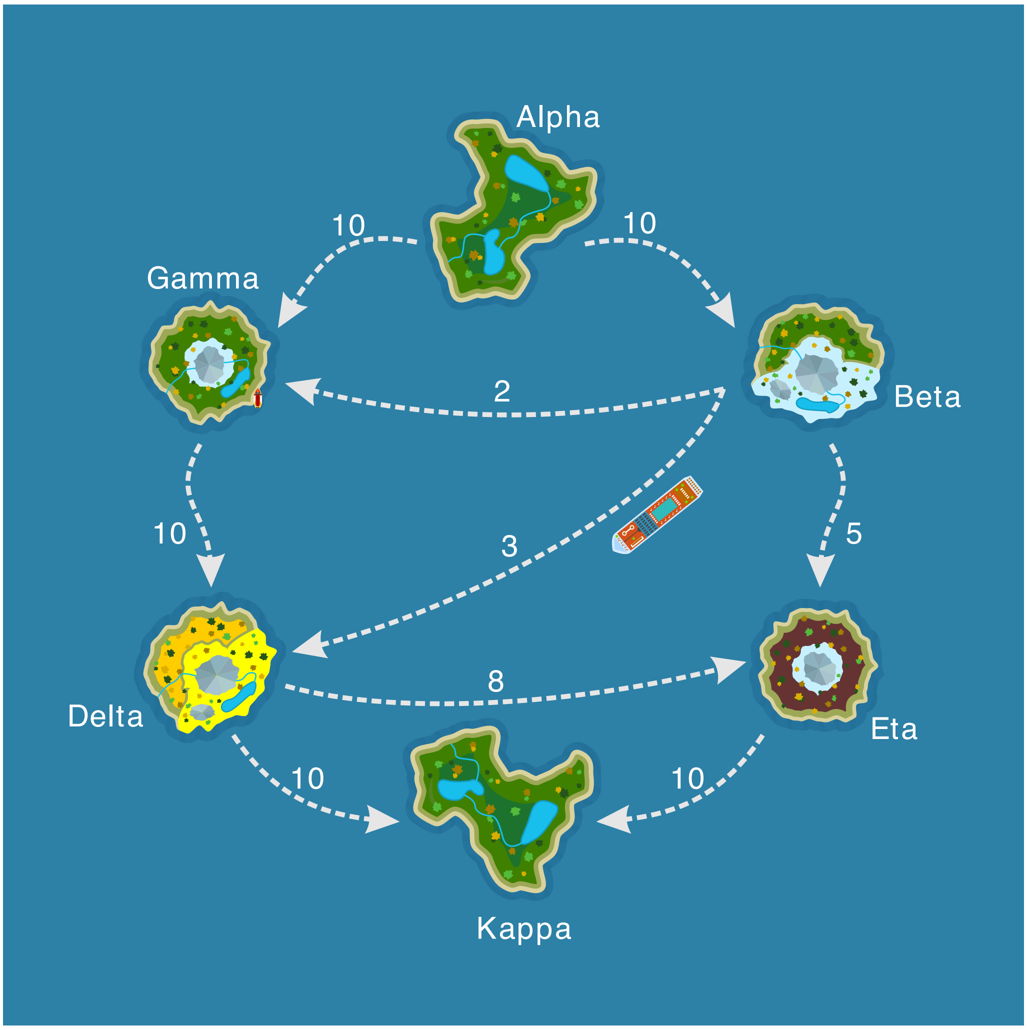
# Dangerous Virus

Alpha Island is the capital of a nation consisting of six islands. The names of the other islands are Beta, Gamma, Delta, Eta, and Kappa.

A dangerous virus is spreading on Kappa Island so it needs medical supplies, but they are only produced on Alpha Island.

To prevent the virus from spreading, the government is limiting all existing transportation routes to those shown below.

The number on each route is the maximum weight of supplies (in kg) that can be sent along that route each day.



Each route can only be used once per day but there are no restrictions on the order in which the routes can be used.

## Example

From Beta Island, 2kg can be sent to Gamma Island, 3kg to Delta Island, and 5kg to Eta Island.

## Question

What is the maximum weight of supplies that can arrive on Kappa Island in one day?

*Select your answer from the options below.*

20kg 18kg 15kg 12kg

*Continued on next page*

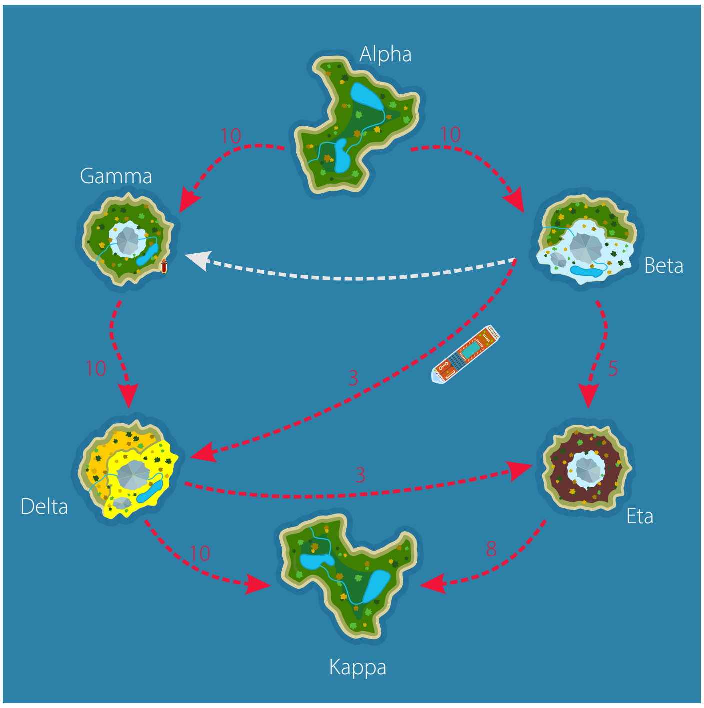
# Dangerous Virus – continued

## Answer

**The answer is: b) 18kg**

From Alpha Island, 10kg of medical supplies can be sent to Gamma Island and from there continues to Delta Island and then on to Kappa Island. Maximum weight of medical supplies that can arrive via this route in one day = 10kg

The second route is more complicated.



From Alpha Island, 10kg of medical supplies can be sent to Beta Island.

From Beta Island, 3kg of medical supplies are sent to Delta Island and 5kg to Eta Island.

*There is no point sending 2kg to Gamma Island as the maximum daily outbound weight has already been used for that island.*

*Only 10kg of supplies can be sent directly from Delta Island to Kappa Island each day and this has already been used.*

The 3kg medical supplies still on Delta Island must first be sent to Eta Island.

There it combines with the 5kg sent from Beta Island for a total of 8kg. Maximum weight of medical supplies that can arrive via this route in one day = 8kg

**Therefore, the total medical supplies that can arrive on Kappa Island in one day is 18kg.**

## It's Computational Thinking

**Computational Thinking Skills:** Decomposition, Abstraction, Modelling & Simulation, Algorithms, Evaluation

**Concepts:** Abstraction, Data Representation, Data Interpretation, Specification, Algorithms, Digital Systems, Interactions, Impacts

A graph is a data structure consisting of nodes and edges. Graphs are used to represent many real-life problems. In this problem, the graph is used to represent the network flow. The islands are represented by nodes and transportation routes as edges. Finding daily maximum capacity between islands is a maximum flow problem.

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
digitalcareers.csiro.au

Australia’s National Science Agency

CSIRO digital careers