2024-AT-03-eng Distributed Ledger

6yo–8yo:	*8yo–10yo:*	*10yo–12yo:*	*12yo–14yo:*	*14yo–16yo:*	*16yo–19yo:*
–	hard	hard	medium	–	–

Answer Type: Click To Choose (☐ keep order of multiple-choice/-select)

Categories (click to choose):

☐ algorithms and programming	☐ computer processes and hardware
☐ data structures and representations	☐ communication and networking
	☐ interactions, systems and society

Body

Three beaver friends enjoy playing with marbles and often borrow marbles from each other. To keep track of who owes how many marbles to whom, they decided to write down all the marble transfers on a single piece of paper:

Example of a shared transfer list (not relevant to the question):

Lender	**Borrower**	**Quantity**
Anna	Tim	5
Tim	Bernd	3
...	...	...

After some time the beaver friends started having doubts. They accused each other of cheating and believed that some of the transfers never happened.

The next day, they decided not to rely on a single list to which everyone has access. Every beaver should create their own list to detect fraud attempts. After one week, the beavers compare their lists. Was there another attempt to cheat?

Anna's List:

Lender	**Borrower**	**Quantity**
Tim	Anna	2
Bernd	Tim	1
Bernd	Anna	3
Anna	Tim	2

Tim's List:

Lender	Borrower	Quantity
Tim	Anna	2
Bernd	Tim	1
Tim	Anna	3
Anna	Tim	2

Bernd's List:

Lender	Borrower	Quantity
Tim	Anna	2
Bernd	Tim	1
Anna	Tim	2
Bernd	Anna	3

Question / Challenge

Who cheated?

Answer Options / Interactivity Description

1) Anna

2) Bernd

3) Tim

4) Nobody

Answer Explanation

Looking at the sequences (2, 1, 3, 2) only, it seemed that Bernd cheated because he is the only one that has a different sequence (2, 1, 2, 3) but that’s not the case. Bernds list only has a different order of transactions but the same overall result. It was Tim that cheated by changing the transaction (Bernd, Anna, 3) to (Tim, Anna, 3). If that had gone unnoticed, he would have gained 3 marbles from Bernd.

This is Informatics

The idea that all transactions are not just stored on a single central server of a trusted party, but are distributed across a peer-to-peer network, is one of the basic ideas behind blockchain technology. The approach mentioned in the question only works if most participants tell the truth. Blockchain technology goes one step further by implementing a consensus algorithm,

which would require, for example, that the malicious actor controls more computing power than all other participants combined. This is impossible if the network is large enough. However, the question already shows how trust can be strengthened in an untrusted environment by distributing power among all participants instead of relying on a single source of truth.

This is Computational Thinking

EDIT HERE (delete this): explain which computational thinking competencies are necessary to solve this task. Use about 3 to 8 sentences. Do not explain the correct answers of the task, also do not explain the computer science behind the task. You may, however, relate your explanation to the “Answer Explanation” section and the “This is Informatics” section.

Informatics Keywords and Websites

EDIT HERE (delete this): give some keywords, useful to find further information on the web. The target are the teachers and interested students. // Indicate websites (in English and other languages) which offer further information on the particular domain of informatics mentioned in the “This is Informatics” section. Wikipedia is often a good start.

Computational Thinking Keywords and Websites

EDIT HERE (delete this): give some keywords, useful to find further information on the web. The target are the teachers and interested students. // Indicate websites (in English and other languages) which offer further information on the particular domain of computational thinking mentioned in the “This is Computational Thinking” section.

Wording and Phrases

EDIT HERE (delete this): list of words and phrases used to name important things mentioned in the task body (actors, activities/processes, concepts, definitions, objects, names, etc.) This is to ensure the consequent use of terminology in the task body and to facilitate translation.

Comments

Working group L7, 2024-05-16: We decided to file this task under Work Needed because of the following reason:

- The text is too long, more than one page. Although the concept is interesting, the story needs to be improved to make more sense. Also, we need to exclude the possibility that two people could have cheated in collusion.

EDIT ABOVE (keep this template and add above this line): author, e-mail, date (YYYY-MM-DD): *Comment, including documentation of and reasons for changes, and precautions to take when further developing this task. Especially mention if this task is based upon or directly connected to another task (even from the past).*

Graphics and Other Files

`2024-AT-03-eng.odt` This file.

EDIT ABOVE (keep this template and add above this line): filename, author, source/self-made, license

Please indicate:

- Source of each image or of each other file to produce the task image(s): (URL and/or author)
- License for each image.

Note: If the source of an image is unclear, it might be copyrighted and cannot be used in the task.

Authors, Contributors, and Editors (incl. Graphics)

Andreas Zottl, andreas.zottl@ocg.at, Austria

License

2024-AU-01-eng Hidden pictures

6yo–8yo: –	*8yo–10yo:* –	*10yo–12yo:* –	*12yo–14yo:* –	*14yo–16yo:* hard	*16yo–19yo:* hard
Answer Type: Multiple-Choice with Images (☐ keep order of multiple-choice/-select)					
Categories (click to choose): algorithms and programming ☐ data structures and representations			☐ computer processes and hardware communication and networking ☐ interactions, systems and society		

Body

Leo has invented a new method for encrypting images using the operations H (horizontal) and V (vertical).

An image is essentially a rectangle divided into rows and columns of square cells called pixels, with each pixel storing a color.

In each application of the H operation:

- Every pixel in the 1st row remains in place (i.e., they do not move).
- Every pixel in the 2nd row moves 1 place to the right.
- Every pixel in the 3rd row moves 2 places to the right.

  …
- Every pixel in the *n*th row moves *n* − 1 places to the right.

When pixels in any row are pushed beyond the right edge of the image, they are kept in order and moved as a group to the available space at the left end of the row.

Similarly, in each application of the V operation:

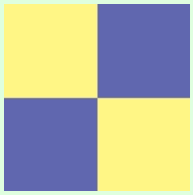
- Every pixel in the *n*th column moves *n* − 1 places down, and pixels pushed beyond the bottom edge are moved to the top.

Here is an example of a 3 × 3 image with colors labeled 1 to 9:

Here is how the sequence HVHV can be used to encrypt a 25 × 25 image of the Mona Lisa:

Question / Challenge

Leo encrypts the following 1000 × 1000 image by applying V then H:



Which of the following best matches the result?

Answer Options / Interactivity Description

A.  B.  C.  D.  E. 

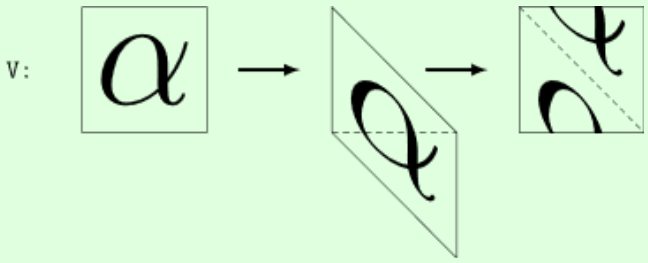
Answer Explanation

The answer is option E.

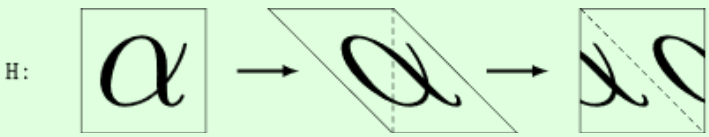
The image has high resolution, so the movement of individual pixels can be reasonably well approximated by the geometrical transformations of the shapes represented in the image.

In the 3 × 3 example, we see that the first column (1, 4, 7) becomes the top-left-to-bottom-right diagonal, while the top-right-to-bottom-left diagonal (3, 5, 7) becomes the last column. This suggests that a shear is taking place, which is further supported by the first image in the Mona Lisa example. We will analyze the geometrical effects of the two operations in more detail, starting with V since it will be applied first.

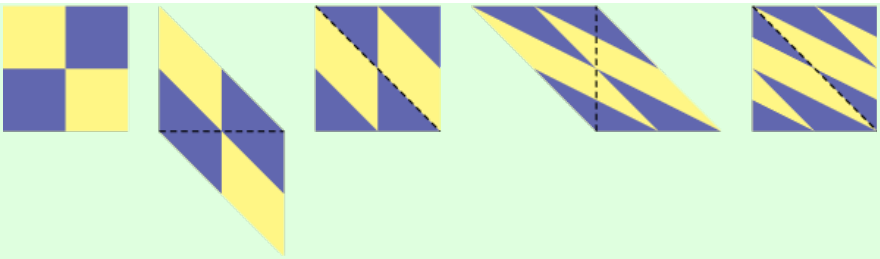
The effect of V can be broken into two stages. First, shear the rectangle vertically towards the bottom right to form a parallelogram with a distorted image. Then, vertically translate the overhanging triangle at the bottom to form a new rectangular image of the original size:



Similarly, the effect of H is a horizontal shear followed by a partial horizontal translation:



Applying these steps to the given image, we have the following sequence:



This is Informatics

When transferring information, it may be important to encrypt the data so that anyone intercepting your information cannot easily determine the original data. In this task, you have been given an encryption algorithm that appears simple, is easily reversible, yet still generates a new image that does not immediately reveal the original. In fact, without knowledge of the encryption mechanism, there could be many potential source images. However, it is important to note that there is no secret key required; all you need to know is the encryption algorithm, and you could decrypt any image encrypted in this manner.

During this process, we convert the image into a matrix, where the cell values represent a level of color. This allows us to represent the data in a way that can be easily operated on.

This is Computational Thinking

The process of abstraction often allows us to understand a problem or algorithm through more heuristic means – grasping the 'gist' of the process rather than absorbing all of the fine details. For example, regardless of the image type (portrait, landscape, photography, etc.) or size. Also, distilling an algorithm down to its essential arithmetical properties can allow us to test the relationship between pairs of inputs and outputs without necessarily conducting a full analysis. The novelty in this problem is that the heuristic approach relies on geometry rather than arithmetic.

Algorithmic thinking is also another skill of Computational Thinking. In this task, the algorithm has two main steps (H and V), and in each of these, the cells or pixels move some places, depending on the row numbering. It can also be used to decrypt the process.

The process of representing images as a matrix is a data representation process, which is a Computational Thinking skill. Choosing the best representation can help solve the problem faster.

Finally, you can observe that generalization (or transferability) is also used in the task. In the first example of the body, a 3 x 3 matrix is shown; next, the example of Mona Lisa has increased the size to 25 x 25; and, in the question, we use the same process with a 1000 x 1000 matrix. This also includes the possibility that the image could be non-square (rectangular).

Informatics Keywords and Websites

Encryption, algorithms, computer graphics

Computational Thinking Keywords and Websites

Abstraction, algorithm thinking, generalization

Wording and Phrases

Mona Lisa.

Comments

Chris Wetherell, chris.wetherell@amt.edu.au, 2024/05/03. Clarified moving 0 places is equivalent to not moving, in response review comment from Pedro Ribeiro [*Perhaps adding something to make clear that "moved 0 places to the right" is equivalent to not moving at all.*]

Chris Wetherell, chris.wetherell@amt.edu.au, 2024/05/03. Removed classification for 12-14 yo in response to several reviewer comments: Pedro Ribeiro [*Ok as a hard task for 14y-16y and 16y-14y, might be a bit too hard for 12y-14y (but still acceptable)*] Seiichi Tani [*The difficulty level of the task may be too challenging for the Cadet age group [12yo – 14yo].*] Angy Coronel-Suarez [*I think it is difficult or time consuming to visualize the solution]* Aho Truu [*Maybe just a little too long for the youngest proposed age group (12-14yo).*]

Chris Wetherell, chris.wetherell@amt.edu.au, 2024/05/03. Added second paragraph to Explanation, in response to reviewer comment by Angy Coronel-Suarez [*I don't know if it is completely evident that the effect of the transformations is the one described or if something else needs to be shown to visualize it better*]

Javier Bilbao, javier.bilbao@ehu.eus, 2024/05/14. Modified some sections: TiI, TiCT, Keywords, Wording.

Praphan Pavarangkoon, praphan@it.kmitl.ac.th, 2024/05/15. Improved the answer options and answer explanation.

Graphics and Other Files

2024-AU-01-eng.odt This file.

2024-AU-01-02-monalisa.svg, Chris Wetherell, self-made (TikZ via Excel, from Photoshop'd low-res version of public-domain Wikipedia image https://en.wikipedia.org/wiki/Mona_Lisa#/media/File:Mona_Lisa,_by_Leonardo_da_Vinci,_from_C2RMF_retouched.jpg), CC

all remaining images (2024-AU-01-01-3by3.svg, 2024-AU-01-03-source.svg, 2024-AU-01-03-OptionA.svg, 2024-AU-01-03-OptionB.svg, 2024-AU-01-03-OptionC.svg, 2024-AU-01-03-OptionD.svg, 2024-AU-01-03-OptionE.svg, 2024-AU-01-03-Vexpln.svg, 2024-AU-01-03-Hexpln.svg, 2024-AU-01-11-solution.svg), Chris Wetherell, self-made (TikZ), CC

Authors, Contributors, and Editors (incl. Graphics)

Chris Wetherell (problem, solution, graphics), Australia, chris.wetherell@amt.edu.au

Eljakim Schrivers ('This is informatics' section), Netherlands, eschrijvers@eljakim.nl

Javier Bilbao ('This is informatics' and 'This is Computational Thinking' sections, at BW2024), Spain, javier.bilbao@ehu.eus.

Praphan Pavarangkoon, ('Answer Options' and 'Answer Explanation' sections, at BW2024), praphan@it.kmitl.ac.th

License

2024-BE-01a-eng Keep Switching

6yo–8yo:	*8yo–10yo:*	*10yo–12yo:*	*12yo–14yo:*	*14yo–16yo:*	*16yo–19yo:*
–	–	hard	medium	easy	–

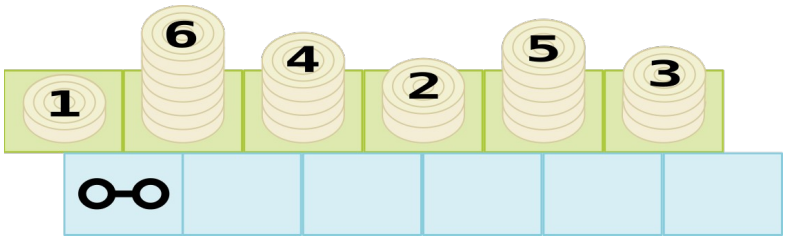
Answer Type: Multiple-Choice with Images (☐ keep order of multiple-choice/-select)

Categories (click to choose):

algorithms and programming
☐ data structures and representations
☐ computer processes and hardware
☐ communication and networking
☐ interactions, systems and society

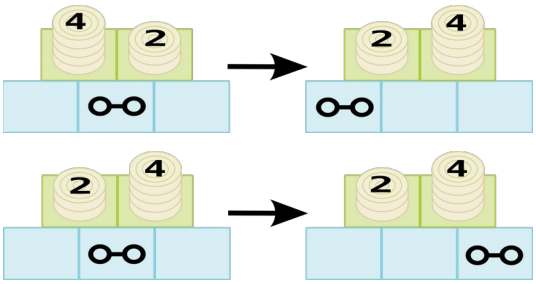
Body

On a board there are two rows of 6 cells, arranged as in the picture below. In the cells on the top row there are stacks of discs, each of different height(1, 6, 4, 2, 5, 3). In the bottom row there is a marker pointing to two cells in the top row.



Starting from the configuration above you keep repeating the following ‘moves’

1. If the left cell of marker is *taller* than the right cell, you switch two cells and move the marker to the *left*, if possible.
2. If the left cell of marker is *shorter* than the right cell, you don’t need to switch and move the marker to the *right*.



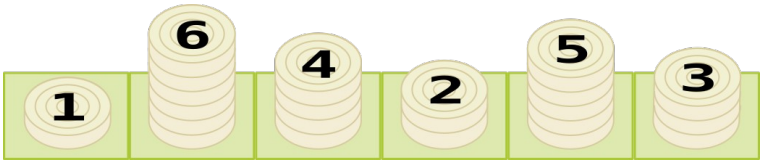
You *stop* repeating these moves when the marker is in the *rightmost* cell.

Question / Challenge

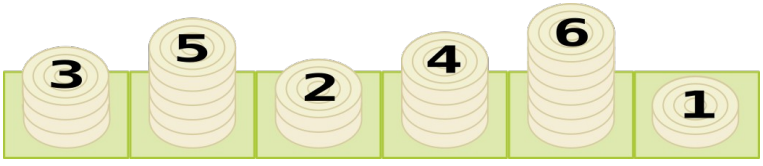
What will the top row look like when the marker ends up in the rightmost cell?

Answer Options / Interactivity Description

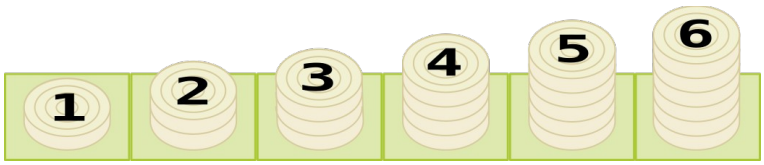
A.



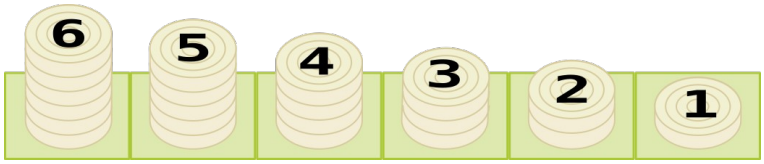
B.



C.



D.



E. There is no solution – the marker never reaches the rightmost cell.

Answer Explanation

The correct answer is **C**.
You can get this answer by carefully making each move as described and seeing where you end up, but a faster way is to realize that what this task does is sorting the stacks.

In fact, as we move the marker, at each point the stacks to the left of the counter are sorted from low to high. Using move 2 the counter will move to the right as long as this order remains correct. If a stack is encountered that is not in the correct position, a series of moves 1 will ‘carry’ that stack into the correct position and the counter can start moving to the right again.

C is the answer in which all stacks are sorted correctly.

This is Informatics

In informatics, it is often necessary to rearrange data so that it is sorted in a particular order, from small to large, from cheap to expensive, and so on. One reason for this is that finding data in a sorted sequence is considerably much faster than finding data in an unsorted sequence.

The method described here is called 'Gnome Sort'. It is easy to understand and execute, and also easy to program, but it is not used in practice because it is much slower than the more sophisticated algorithms that are traditionally used (e.g. quick sort or merge sort).

This is Computational Thinking

Students must understand and execute an algorithm to find the answer to this question, although the clever step is to see that the algorithm does not have to be executed in full to answer the question.

Informatics Keywords and Websites

Sorting, gnome sort, https://en.wikipedia.org/wiki/Gnome_sort

Computational Thinking Keywords and Websites

Algorithm, pattern recognition

Wording and Phrases

Counter, stack, cell, move (noun)

Comments

Kris Coolsaet (BE), kris.coolsaet@ugent.be, 2024-03-17

- There is a related question 2024-BE-01b that uses the same story but is more difficult.
- Because my drawing skills are not so good, I used stacks of counters. The standard gnome sort story is about a garden gnome sorting a line of flower pots. If you find someone that can draw garden gnomes and flower pots, that might be a nice alternative. (On the other hand, for those students that have already heard of Gnome sorting, this might be a give-away.)
- It is perfectly possible to use only 4 of the 5 multiple choice options

Kris Coolsaet (BE), kris.coolsaet@ugent.be, 2024-04-27
Answer to reviews.txt for both 2024-BE-01a and 2024-BE-01b (not copied here)

- Darkened edges of counters #e9ddaf → #d8cc9e
- Moved difficulty up one age group because 2 out of 3 reviewers found the task harder than indicated
- Added ‘if possible’ add the end of the first rule. Because I made sure the case would not occur in which the counter was at the leftmost position and it would be asked to move even further to the left, I did not explain this situation. But I did not realize this could be done with only two words – thank you Tom! I dropped the last paragraph of the TiT section which explained that this exception is needed in the ‘official’ gnome sort.
- As a consequence I removed a sixth MC-option ‘There is no solution – the black counter must at some point move to the left of the leftmost cell, and this is not possible’, but anyhow, I personally would drop the ‘there is no solution’ options anyway.
- I did not implement the suggestion of making pictures for the first few sorting steps as part of the answer explanation because of the amount of work this requires (and the amount of space this will take on the page). Also it is not clear whether this should be done only for the correct answer or also for the incorrect ones. (If the working group specifies exactly what pictures needs to be made, I am willing to do help them with that.)
  It is stated in the answer explanation that actually performing the moves is not really necessary to find the correct answer!

Dongyoon Kim and hongjin Yeh(KR), dykim@ajou.ac.kr, hjyeh@ajou.ac.kr, 2024-05-15

- Slightly reworded and simplify the sentences.

EDIT ABOVE (keep this template and add above this line): author, e-mail, date (YYYY-MM-DD): *Comment, including documentation of and reasons for changes, and precautions to take when further developing this task. Especially mention if this task is based upon or directly connected to another task (even from the past).*

Graphics and Other Files

`2024-BE-01a-eng.odt` This file.

All SVG files in the `graphics` folder - author Kris Coolsaet (BE), kris.coolsaet@ugent.be, 2024-03-17, self-made from scratch

All SVG files are modified to add numbers and a special character as a marker - hongjin Yeh(KR), hjyeh@ajou.ac.kr, 2024-05-15

EDIT ABOVE (keep this template and add above this line): filename, author, source/self-made, license

Please indicate:

- Source of each image or of each other file to produce the task image(s): (URL and/or author)
- License for each image.

Note: If the source of an image is unclear, it might be copyrighted and cannot be used in the task.

Authors, Contributors, and Editors (incl. Graphics)

Kris Coolsaet (BE), kris.coolsaet@ugent.be, original author

Dongyoon Kim and hongjin Yeh(KR), dykim@ajou.ac.kr, hjyeh@ajou.ac.kr, contributors (incl. graphics)

EDIT ABOVE (keep this template and add above this line): author / contributor / editor, e-mail, country.

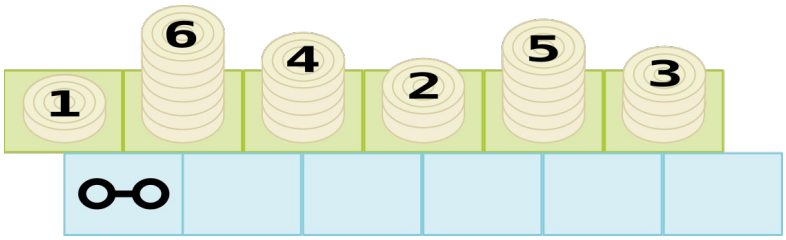
License

2024-BE-01b-eng Keep Switching

6yo–8yo:	*8yo–10yo:*	*10yo–12yo:*	*12yo–14yo:*	*14yo–16yo:*	*16yo–19yo:*
–	–	hard	medium	medium	medium
Answer Type: Multiple-Choice with Images (☐ keep order of multiple-choice/-select)					
Categories (click to choose): algorithms and programming ☐ data structures and representations			☐ computer processes and hardware ☐ communication and networking ☐ interactions, systems and society		

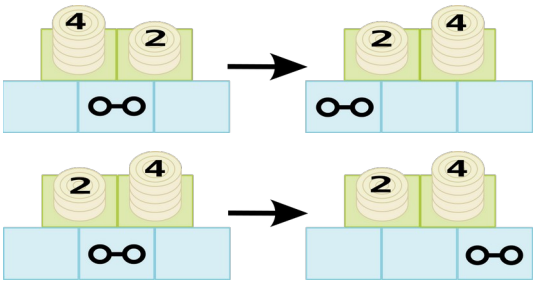
Body

On a board there are two rows of 6 cells, arranged as in the picture below. In the cells on the top row there are stacks of discs, each of different height(1, 6, 4, 2, 5, 3). In the bottom row there is a marker pointing to two cells in the top row.



Starting from the configuration above you keep repeating the following ‘moves’

1. If the left cell of marker is *taller* than the right cell, you switch two cells and move the marker to the *left*, if possible.
2. If the left cell of marker is *shorter* than the right cell, you don’t need to switch and move the marker to the *right*.



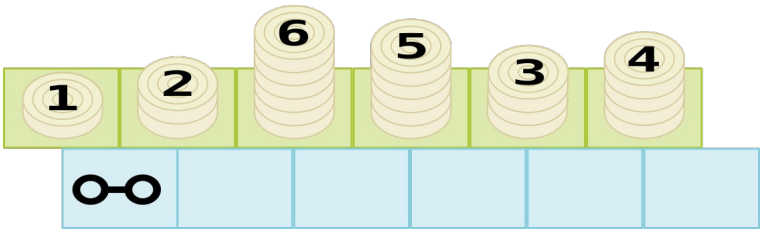
You *stop* repeating these moves when the marker is in the *rightmost* cell.

Question / Challenge

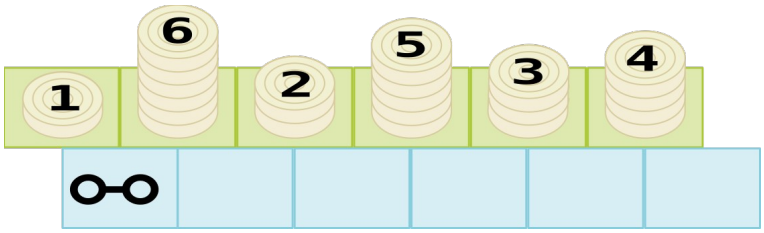
Below are four different start configurations. Which of these will take the fewest moves for the marker to reach the rightmost cell?

Answer Options / Interactivity Description

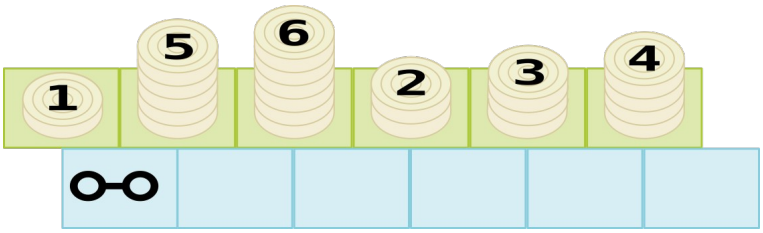
A.



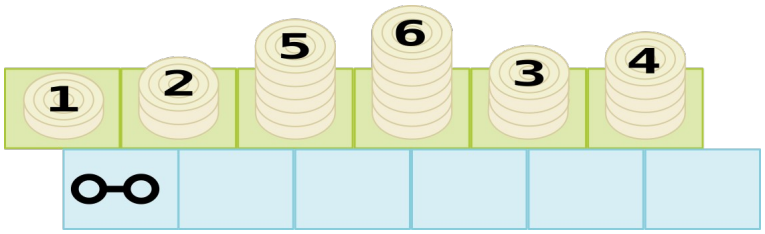
B.



C.



D.



Answer Explanation

The correct answer is **D**.
To find the correct answer it is not necessary to execute each move as described in the task and count how many moves you need before the marker reaches the rightmost cell (and that for each of the four possibilities). Instead you can compare the number of moves between the four options without actually counting them all.
To simplify our explanation, we shall represent the stacks by numbers (how many discs there are in that stack) and the position of the marker by a dot. As follows:

Option A	1 ● 2 6 5 3 4	Option C	1 ● 5 6 2 3 4
Option B	1 ● 6 2 5 3 4	Option D	1 ● 2 5 6 3 4

Have a look at the first moves for option B. The marker first moves to the right (move 2.) then switches stacks and moves to the left (move 1.), i.e.,

1 ● 6 2 5 3 4 → 1 6 ● 2 5 3 4 → 1 ● 2 6 5 3 4

After 2 moves, the configuration is exactly the same as in configuration A. Hence B takes longer than A and cannot be the answer.

Now have a look at what the first moves are for option A

1 ● 2 6 5 3 4 → 1 2 ● 6 5 3 4 → 1 2 6 ● 5 3 4 →1 2 ● 5 6 3 4

and this is where option D will end up after exactly one move. Hence A takes longer than D.

These are the first few moves for option C

1 ● 5 6 2 3 4 1 5 ● 6 2 3 4 → 1 5 6 ● 2 3 4 → 1 5 ● 2 6 3 4 → 1 ● 2 5 6 3 4

which is the configuration in which D starts. Hence D takes the least number of moves.

This is Informatics

Note that all the options would eventually end up with 1 2 3 4 5 6 ●. In fact, it is easy to see that what this task does is sort the stacks by height.

In informatics, it is often necessary to rearrange data so that it is sorted in a particular order, from small to large, from cheap to expensive, and so on. One reason for this is that finding data in a sorted sequence is considerably much faster than finding data in an unsorted sequence.

The method described here is called 'Gnome Sort'. It is easy to understand and execute, and also easy to program, but it is not used in practice because it is much slower than the more sophisticated algorithms that are traditionally used (e.g. quick sort or merge sort).

This is Computational Thinking

Students must understand and execute an algorithm to find the answer to this question, although the clever step is to see that the algorithm does not have to be executed in full to answer the question.

Informatics Keywords and Websites

Sorting, gnome sort, https://en.wikipedia.org/wiki/Gnome_sort

Computational Thinking Keywords and Websites

Algorithm

Wording and Phrases

Counter, stack, cell, move (noun)

Comments

Kris Coolsaet (BE), kris.coolsaet@ugent.be, 2024-03-17

- There is a related question 2024-BE-01b that uses the same story but is easier.
- Because my drawing skills are not so good, I used stacks of counters. The standard gnome sort story is about a garden gnome sorting a line of flower pots. If you find someone that can draw garden gnomes and flower pots, that might be a nice alternative. (On the other hand, for those students that have heard of Gnome sorting, this might be a give-away.)

Kris Coolsaet (BE), kris.coolsaet@ugent.be, 2024-04-27

- Answer to reviews.txt for both 2024-BE-01a and 2024-BE-01b (not copied here)
- Darkened edges of counters #e9ddaf → #d8cc9e
- Changed the difficulty from easy to medium for the two oldest age groups
- Added ‘if possible’ add the end of the first rule. Because I made sure the case would not occur in which the counter was at the leftmost position and it would be asked to move even further to the left, I did not explain this situation. But I did not realize this could be done with only two words – thank you Tom! I dropped the last paragraph of

the TiT section which explained that this exception is needed in the ‘official’ gnome sort.

Dongyoon Kim and hongjin Yeh(KR), dykim@ajou.ac.kr, hjyeh@ajou.ac.kr, 2024-05-15

- Slightly reworded and simplify the sentences.

EDIT ABOVE (keep this template and add above this line): author, e-mail, date (YYYY-MM-DD): *Comment, including documentation of and reasons for changes, and precautions to take when further developing this task. Especially mention if this task is based upon or directly connected to another task (even from the past).*

Graphics and Other Files

`2024-BE-01b-eng.odt` This file.

All SVG files in the `graphics` folder - author Kris Coolsaet (BE), kris.coolsaet@ugent.be, 2024-03-17, self-made from scratch

All SVG files are modified to add numbers and a special character as a marker - hongjin Yeh(KR), hjyeh@ajou.ac.kr, 2024-05-15

EDIT ABOVE (keep this template and add above this line): filename, author, source/self-made, license

Please indicate:

- Source of each image or of each other file to produce the task image(s): (URL and/or author)
- License for each image.

Note: If the source of an image is unclear, it might be copyrighted and cannot be used in the task.

Authors, Contributors, and Editors (incl. Graphics)

Kris Coolsaet (BE), kris.coolsaet@ugent.be, original author

Dongyoon Kim and hongjin Yeh(KR), dykim@ajou.ac.kr, hjyeh@ajou.ac.kr, contributors (incl. graphics)

EDIT ABOVE (keep this template and add above this line): author / contributor / editor, e-mail, country.

License

2024-BG-01b-eng Balls

6yo–8yo:	*8yo–10yo:*	*10yo–12yo:*	*12yo–14yo:*	*14yo–16yo:*	*16yo–19yo:*
–	–	–	–	hard	medium

Answer Type: Interactive (Drag & Drop) (☐ keep order of multiple-choice/-select)

Categories (click to choose):

algorithms and programming
☐ data structures and representations
☐ computer processes and hardware
☐ communication and networking
☐ interactions, systems and society

Body

A sequence of red and blue balls is given:

We count the number of blue balls from left to right starting from the first ball, then starting from the second ball and so on, and obtain the following sequence:

3, 3, 2, 1, 1, 1.

Now we write 0, if the number is even and write 1, if the number is odd and get the following string:

110111

Question / Challenge

Having the next string of 0's and 1's (meaning even and odd):

01110100

construct the sequence of color balls that corresponds to the above string.

Interactivity Description

The participant have to use an interactive software with "drag" and "drop" tools to construct the sequence of color balls.

It is also possible to set the task to be in "open text answer" type. The participant has to write a sequence of 'r' (for red) and 'b' (blue) representing the sequence of color balls.

Answer Explanation

Correct answer:

We can devise two simple algorithms to reach this solution. One that processes the sequence from left to right, and one that processes it right to left.

To process the sequence from left to right, we can use a loop in which we successively take the characters from the given string of zeros and ones. We maintain two variables *c* and *cc*, which at each step contain the values of two consecutive elements of the given string. At the beginning, the variable *c* contains the starting element, and the variable *cc* contains the next one.

When *c*=='0', it means that the number of blue balls from the current position to the end is even. Then there are two cases.

First case: *cc*=='0', which implies that on pass from the current position to the next position, the number of blue balls will not change, i.e. the current position is marked with a red ball.

Second case: *cc*=='1', which implies that on pass from the current position to the next position, the number of blue balls will change, i.e. there is a blue ball at the current position.

When *c*=='1', the reasoning is analogous, with the difference that at *cc*='0' there is a blue ball at the current position, and at *cc*='0' there is a red one.

We can write the above algorithm in pseudo-code that sequentially outputs ‘r’ for red ball and ‘b’ for blue ball. We assume that the length of input string is *n*:

```
get c;
for(i=0;i<n-1;i++)
{
  get cc;
  if(c=='0') {
     if(cc=='0')
          w='r';
     else
          w='b';
     }
  else {
     if(cc=='0')
          w='b';
     else
          w='r';
     }
  output w;
  c=cc;
}
```

Finally, we need to process the last element *c* of the input string. In output, there can be either red, or blue ball. When *c*=='0', it means the number of blue balls is 0 i.e. there will be a red ball (recall that zero is an even number), and vice versa, if *c*=='1', there will be a blue ball.

We can also process the sequence from right to left, offering a simpler algorithm to find the sequence:

The number of blue balls changes when the sequence changes from 1 to 0 or from 0 to 1, going backwards as you can see on the table below.

If the final number of the sequence is 0 the last ball is red, and if the final number of the sequence is 1, the last ball is blue.

0	1	1	1	0	1	0	0
4 blue	3 blue	3 blue	3 blue	2 blue	1 blue	0 blue	0 blue

Then the sequence of blue balls is **43332100.**

This is Informatics

In informatics various encoding are used. Encodings help us to describe the situation to computers. In many ways we can omit some information, make the encoding easier and still not lose the original information. In this task, the proposed encoding didn't store the exact numbers of red and blue balls, but stored it as a sequence of 0s and 1s (computers can use less space with only 0 and 1 in code, in comparison to other numbers). Nonetheless, we are still able to restore the original information – the sequence of balls. Therefore, for each encoding you need to know how it was created (which algorithm was used) so you can use it correctly.

These algorithms are known as compression algorithms. They help lowering the size of a file by reducing the number of bits they are composed of, without losing information. For example, the PNG image format uses a combination of compression algorithms in order to lower the size of the file.

One way to construct the sequence of balls is by the means of a one-pass algorithm, also known as "read and destroy," which processes data in a single pass, involving reading and discarding it without the need for revisiting. This approach helps improve efficiency and reduce resource consumption when encrypting, compressing/decompressing, and in general processing large amounts of data.

This is Computational Thinking

Efficiently manage data processing tasks through sequential reading and immediate disposal. This requires an understanding of algorithmic efficiency, data management, and optimization techniques to perform tasks with minimal resource usage.

Informatics Keywords and Websites

Data compression

https://en.wikipedia.org/wiki/Data_compression

One-pass algorithm

https://en.wikipedia.org/wiki/One-pass_algorithm

Computational Thinking Keywords and Websites

Algorithm. Loop.

Wording and Phrases

Comments

Emil Kelevedjiev (BG), keleved@gmail.com, 2024-03-26:

There is a related task 2024-BG-01a that uses the similar story, but it is easy – you just have to count and compare.

==========
By: Zsuzsa Pluhár (svn: pluhar)
on 2024-04-06
Non-assigned review
Nice idea for data representation.
Body, texting: if we have "sequence" of numbers first, then I would use "sequence" for the "111011..." as well (and not string)
Answer explanation: must be extended with the 1/0 sequences pairing with the images.

TiI: Poor and not CS, it is only "counting some objects"- it doesn't contain CS background - the keywords are mostly from math - maybe the "prefix array" could be explained in TiI.
(but maybe it is only a special prefix array. - "A prefix sum array, also known as a cumulative sum array, is a derived array that stores the cumulative sum of elements in a given array. Each element in the prefix sum array represents the sum of all elements up to that index in the original array")
If you write something in the Wording, please explain the important words.

Graphics: only png files are included. The images must be created in svg format.
==========
By: Susanne Datzko (svn: sdatzko)
on 2024-04-14
Assigned review
[AGE] appropriate
[IDEA] I like the idea of the task. It is quite simple to understand but not so easy to solve. I like the interactive version of the task, when the participants have to construct an array following the "101-sequence"
[TEXT] The text has to be simpler und more straight forward - but probably use an example to explain it step by step.
[EXPL] Explanation has to be step by step. Your explanation is too technical. Probably use it in the TiI section.
[TiI] Could you probably explain with this, how you need to use a for-loop to index an array?
[TiCT] I hope this is not true - This would be then a very boring task.
[GRAPH] Very easy graphic to do in svg - you could write me an e-Mail to make you those graphics.
==========

By: Willem van der Vegt (svn: vandervegt)
on 2024-04-16
Assigned review
Though the use of odd and even is a bit far fetched, the task of retrieving the order from an encoded description is really challenging and a nice Bebras task.
[AGE] Appropriate
[IDEA] Nice
[TIL] The It's informatics section should be expanded. The one pass algorithm is also known as read-and-destroy. It is also about a form of encryption and that can be included in the It's Inf section.
==========
By: Tom Naughton (svn: naughton)
on 2024-04-22
Assigned review
[AGE] good

[IDEA] A nice idea. It would be great if it could be improved by bringing into the story some real world aspect that would make it meaningful to only keep track of even/oddness; the problem of equally sharing sweets between two children, maybe?

[TEXT] Text is clear.

[EXPL] A technical explanation is given for one way to solve it, however, two things seem to be missing that would make this an excellent task:

1) There should be a way to speed up finding a solution through a deeper understanding of the task. A task's difficulty should not be based on how tedious it is to solve. Is a short-cut possible here, or is it the case that even the brightest students must laboriously follow the algorithm?

2) There should be an intuitive explanation why this is the ONLy correct solution for the binary sequence 01110100. Was this sequence chosen very carefully -- if the sequence is changed slightly might there be multiple correct solutions?

[TiI] Needs work.
[TiCT] Needs work.
[GRAPH] good
[AUTH] good
[NAME] good
[OTHER]
==========

WG L3, Reem, reem.h.alotaibi@hotmail.com, 2024-05-15: *Expand the informatics section . The one pass algorithm is also known as read-and-destroy.*

WG L3, Masiar Babazadeh, masiar.babazadeh@supsi.ch, 2024-05-16: *Expanded even further the TiI. Adjusted the pseudocode in the answer description. Made clear that there are two different algorithms.*

EDIT ABOVE (keep this template and add above this line): author, e-mail, date (YYYY-MM-DD): *Comment, including documentation of and reasons for changes, and precautions to take when further developing this task. Especially mention if this task is based upon or directly connected to another task (even from the past).*

Graphics and Other Files

`2024-BG-01b-eng.odt` This file.

All files in the graphic folder are self-made. The author is Emil Kelevedjiev (BG), keleved@gmail.com,

EDIT ABOVE (keep this template and add above this line): filename, author, source/self-made, license

Please indicate:

- Source of each image or of each other file to produce the task image(s): (URL and/or author)
- License for each image.

Note: If the source of an image is unclear, it might be copyrighted and cannot be used in the task.

Authors, Contributors, and Editors (incl. Graphics)

Emil Kelevedjiev (BG), keleved@gmail.com, original author

WG L3 – Reem Alotaibi, reem.h.alotaibi@hotmail.com, SA

WG L3 – Eugenio Bravo, eugenio.bravo@ehu.es, ES

WG L3 – Masiar Babazadeh, masiar.babazadeh@supsi.ch, CH

EDIT ABOVE (keep this template and add above this line): author / contributor / editor, e-mail, country.

License

2024-BR-04-eng Organizing bracelets

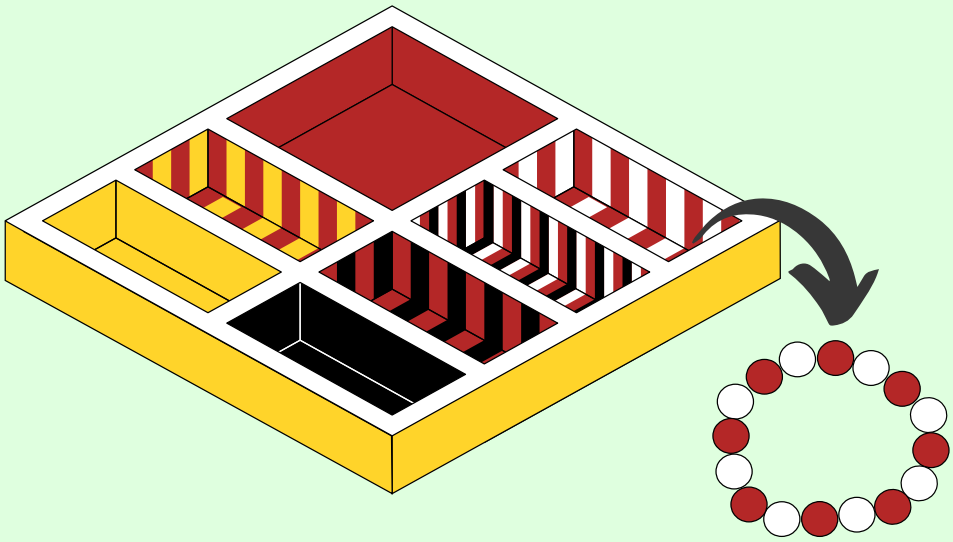
6yo–8yo: medium	*8yo–10yo:* –	*10yo–12yo:* –	*12yo–14yo:* –	*14yo–16yo:* –	*16yo–19yo:* –
Answer Type: Click To Choose (☐ keep order of multiple-choice/-select)					
Categories (click to choose): ☐ algorithms and programming data structures and representations			☐ computer processes and hardware ☐ communication and networking ☐ interactions, systems and society		

Body

Victoria has a box where she organizes her bracelets.

There are seven compartments in this box and Victoria puts each bracelet in a compartment with the same color pattern.

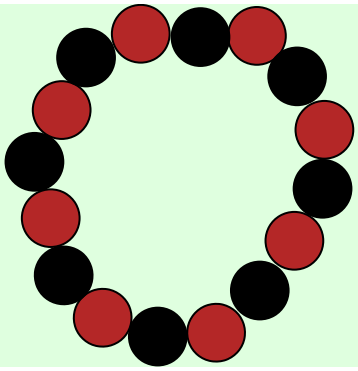
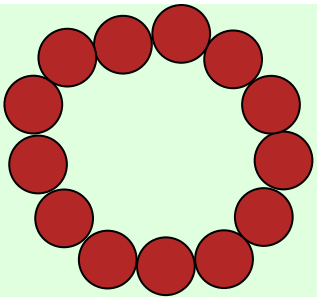
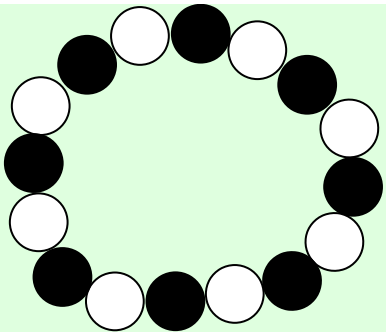
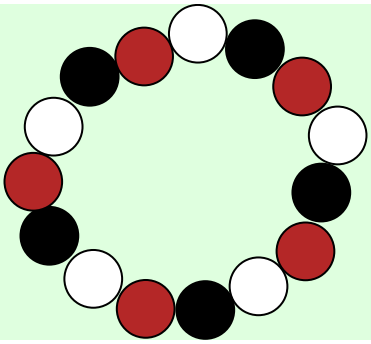
The image below shows where she puts one of her bracelets.



Question / Challenge

Which of these bracelets does **NOT** match any of the compartments in Victoria's box?

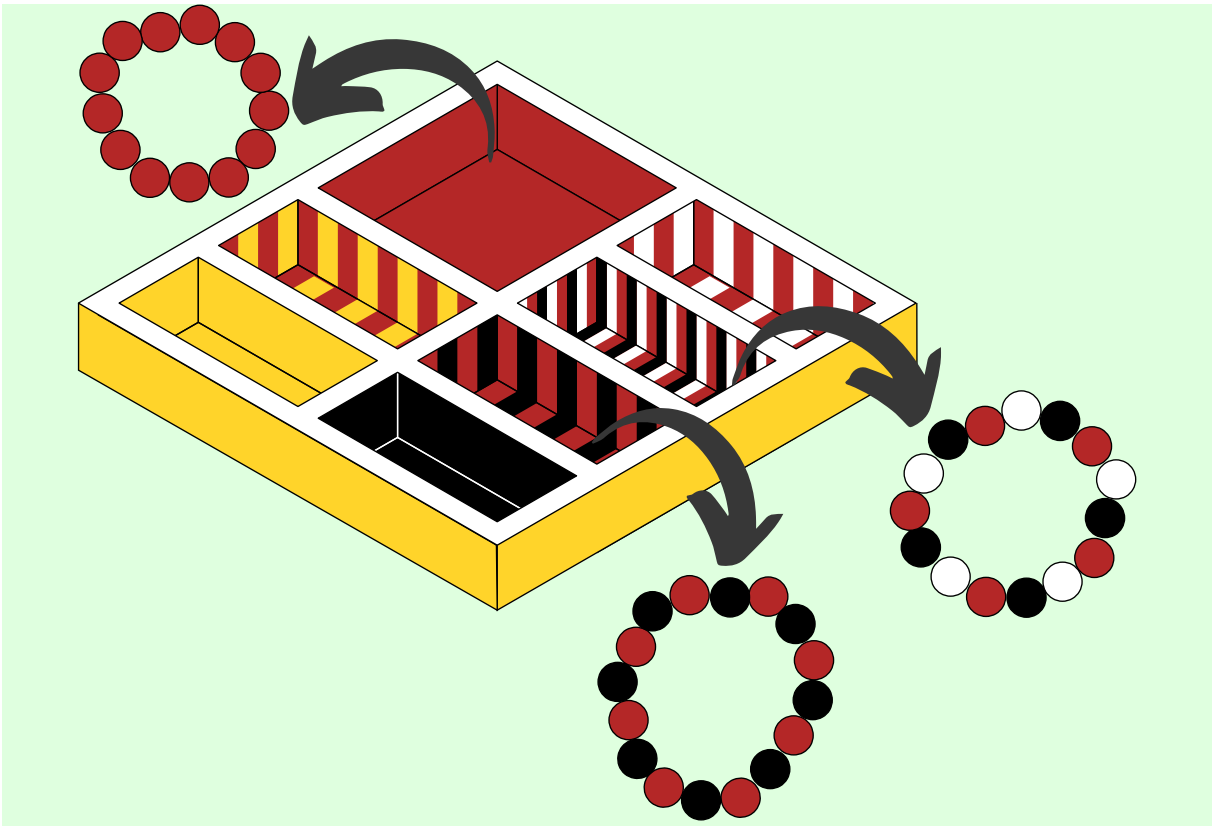
Answer Options / Interactivity Description

A	B
	
C	**D**
	

Answer Explanation

The correct answer is (C).

The image below shows the compartments where Victoria puts the bracelets from options (A), (B) and (D). Observe that the bracelets and the compartments have the same color pattern.



Note there is no compartment with the color pattern black-white in Victoria's box, so there is no compartment for the bracelet in option (C) to match.

This is Informatics

We often have a lot of data to work with when trying to solve a question or problem. Just like in this task, where there are many aspects to the bracelets we could use to put them into the correct boxes. It is important to be able to filter out the data we don't need, so it is easier to find the important data and find a solution. In this task, we only need to look at the colors, it doesn't matter what shape the beads are, how many there are or in what pattern they occur. We don't even need to see the whole bracelet to solve the question, only seeing a part of the images would be enough.

This is Computational Thinking

This task uses a simple form of abstraction. We need to ignore the unnecessary details of the bracelets in order to find the information we need to solve the task. In this task, it is important that we realize that it is the color that is important, that is the only information we need about bracelets to solve the task.

Informatics Keywords and Websites

EDIT HERE (delete this): give some keywords, useful to find further information on the web. The target are the teachers and interested students. // Indicate websites (in English and other languages) which offer further information on the particular domain of informatics mentioned in the "This is Informatics" section. Wikipedia is often a good start.

Computational Thinking Keywords and Websites

Keywords:

Pattern recognition

Websites:

https://www.learning.com/blog/pattern-recognition-in-computational-thinking/#:~:text=Pattern%20recognition%20as%20part%20of,have%20one%20or%20multiple%20similarities

Wording and Phrases

Victoria is a common name in Brazil. Fell free to change it, if necessary.

Comments

```
By: Sarah Chan (svn: chan)
on 2024-04-04
Assigned review
This task crashed my LibreOffice repeatedly. When I did manage to get it
open, all the bracelet images appeared the same (solid black beads). I
think there is an issue with the images, but I think I understand what the
task is intended to be.

Are the patterns all meant to be in grayscale? Or is that a problem with my
LibreOffice rendering?  As I am seeing it, the grayscale patterns are not
different enough from each other. More points of difference between the
patterns (such as colours or different shapes) would really help.

I personally find pattern matching to be a weak Informatics connection
although I accept it as a computational thinking skill. I also accept that
this type of task can work well for younger students.
```

Thank you very much for your suggestions, Sarah. I have implemented all of them.

```
By: doyong kim (svn: doyong)
on 2024-04-05
Assigned review
I love this task. This problem gives to me a variety of imaginations in
pattern matching. However, the Reason for the correct answer needs to be
clearer.
```

Thank you very much for your suggestions, Doyong Kim. I have put more information into the solution.

```
By: LucÃa Crivelli (svn: crivelli)
on 2024-04-15
Non-assigned review
[AGE] Ok
[IDEA] Interesting
[TEXT] Ok
[EXPL] Ok
[TiI] OK
[TiCT] OK
[GRAPH] It would be nice to add color to the graphics.
```

Thank you very much for your suggestions, Lucia. I have implemented colors on the pictures.

By: Workgroup L8 (Alieke Stijf), astijf@eljakim.nl, 2024-05-15:

We agreed that this task is not about pattern recognition. We changed the TiI section to try to strengthen its ties to informatics. We also changed the TiCT section to tie the task to abstraction. We know the TiI link is quite weak, but for this young age group, it nicely covers the very beginnings of the informatics concepts.

Unfortunately we had to redraw all the pictures, as the original files were huge (MBytes), made open office very slow and could not be properly printed.

EDIT ABOVE (keep this template and add above this line): author, e-mail, date (YYYY-MM-DD): *Comment, including documentation of and reasons for changes, and precautions to take when further developing this task. Especially mention if this task is based upon or directly connected to another task (even from the past).*

Graphics and Other Files

`2024-BR-04-eng.odt` This file.

2024-BR-04i.svg

2024-BR-04-A.svg … 2024-BR-04-D.svg

2024-BR-04iii.svg

Graphics were originally made by Emily Cavalcanti and Leonardo Cavalcante, emily.cavalcanti@upmat.com.br, leonardo.cavalcante@upmat.com.br, Brazil but have been redrawn entirely by Kris.Coolsaet@ugent.be (Belgium) during de 2024 workshop (see comments)

EDIT ABOVE (keep this template and add above this line): filename, author, source/self-made, license

Please indicate:

- Source of each image or of each other file to produce the task image(s): (URL and/or author)
- License for each image.

Note: If the source of an image is unclear, it might be copyrighted and cannot be used in the task.

Authors, Contributors, and Editors (incl. Graphics)

Author: Emily Cavalcanti, emily.cavalcanti@upmat.com.br, Brazil

Contributor: Leonardo Cavalcante, leonardo.cavalcante@upmat.com.br, Brazil

Contributor (workshop): Alieke Stijf, astijf@eljakim.nl, Netherlands

EDIT ABOVE (keep this template and add above this line): author / contributor / editor, e-mail, country.

License