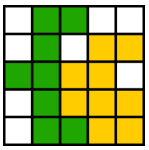
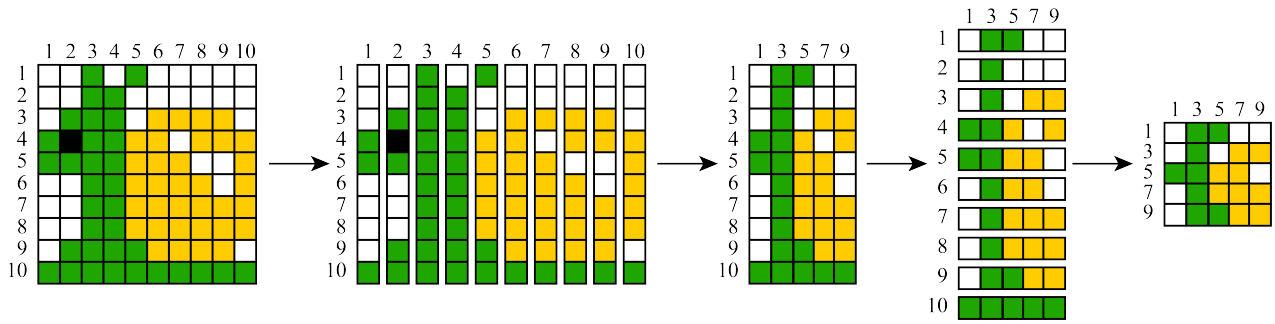
Answer

(D) 

Explanation of Answer

The steps the beaver uses are illustrated below.



Alternatively, you can solve this multiple-choice question by considering the four options. Since the dark "eye" of the snail is in an even-numbered horizontal strip, it will be removed. This eliminates Options A and B. Similarly, the bottom horizontal strip (numbered 10) is the only one with more than four green cells. Since it is an even-numbered strip, it will be removed. This eliminates Option C because it contains five green cells in its bottom row. Therefore, of the options, Option D must be correct.

Connections to Computer Science

This task is focussed on *image compression*, which is the process of reducing the size of an image file by eliminating or minimizing some of the information in the image while still maintaining sufficient image quality for use. Reducing the size of an image can save both storage space and improve the time needed to transfer the image between devices, such as between two phones.

In this task, the total amount of information stored is only 25% of the original amount of information. However, some of the details of the image are removed. This type of compression is an example of a *lossy compression method* that works well for certain types of images that don't have much variation.

There are also *lossless compression methods*, which allow the original information to be stored in a smaller size completely, without any loss of information. Two well-known examples of lossless compression are ZIP files and PNG files.

Country of Original Author

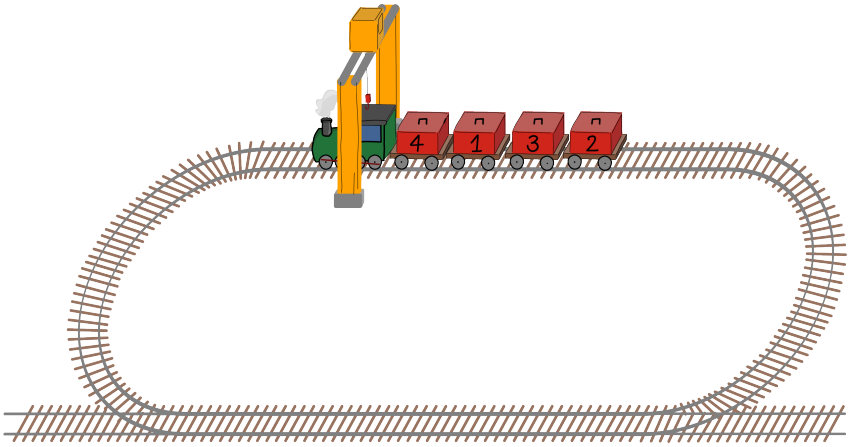
Vietnam

Unloading

Story

Freight trains consist of an engine followed by wagons, each holding a numbered box. The boxes must be unloaded in increasing order, starting from box 1. To unload a box, its wagon must be positioned directly below the crane.

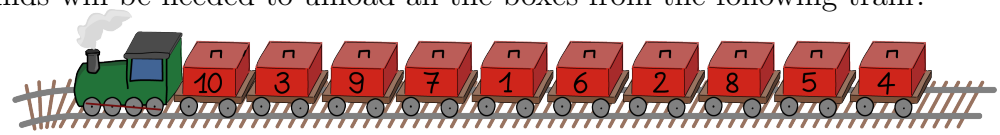
The crane is in a fixed position and trains can only move forward on a loop. Usually, this means that several rounds are needed to unload all the boxes. Each round begins with the engine directly under the crane.



In the example shown, the boxes have to be unloaded in the order 1, 2, 3, 4 and three rounds are needed to do this. In the first round of unloading, the train moves forward to skip box 4, unload box 1, skip box 3, and unload box 2. The train then goes around the track until the engine is under the crane again. In the second round of unloading, the train moves forward to skip box 4, skip the empty wagon, and unload box 3. The train then has to come back for a third round in order to unload box 4.

Question

How many rounds will be needed to unload all the boxes from the following train?



(A) 5

(B) 6

(C) 7

(D) 8

Answer

(C) 7

Explanation of Answer

We can simulate the unloading of the boxes in the required order 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 in as few rounds as possible. This just means that a box is unloaded if it is directly below the crane and all lower-numbered boxes have already been unloaded. Carefully following this procedure, we can determine that boxes are unloaded as shown in the following table.

Round	Boxes Unloaded
1	1,2
2	3,4
3	5
4	6
5	7,8
6	9
7	10

Alternatively, one might discover a general rule that determines exactly how many extra rounds (rounds after the first round) are needed. For each number in the sequence 1, 2, ...,10, if the next number in the sequence comes to the left of it on the train, an extra round is needed. For instance, if 3 appears to the left of 2, then 3 will be skipped to unload 2 and so an extra round is needed to bring 3 under the crane. For this question, there are six such pairs that are out of order on the train: (2,3), (4,5), (5,6), (6,7), (8,9), and (9, 10). Therefore, six extra rounds are needed, for a total of seven rounds.

Connections to Computer Science

This task has the concept of counting *inversions* as the key idea. Generally, an inversion in a list of numbers is where a number, say 6, has the next largest number, 7, written earlier in the list. In the context of this task, an inversion is where box K has box $K + 1$ somewhere to the left of it on the train. For each such inversion, an extra round is needed. If we count the number of inversions, we get the answer.

Counting inversions with respect to a desired sequence has many applications in computer science. For some *sorting algorithms*, such as *insertion sort*, the number of inversions tells us how many swaps we need to sort a given sequence. Consider another example from online shopping, where two customers rank the same set of items in order of preference. Then, the number of inversions in their rankings tells us how aligned their tastes are, and this information can be used by online retailers to identify "similar" customers to make product recommendations.

Country of Original Author

India

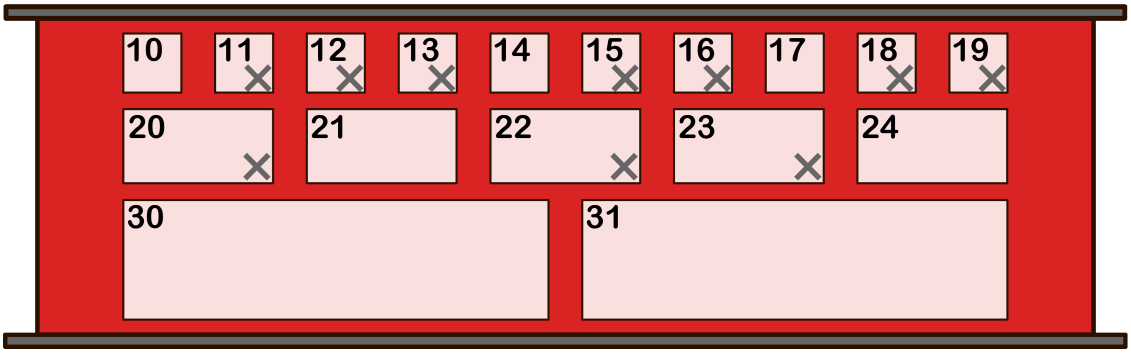
Part C

Lockers

Story

When packages arrive at the post office they are placed in lockers to await pick up. The top row of lockers can only hold small packages. The middle row of lockers can hold small or medium packages. The bottom row of lockers can hold packages of any size. Each locker can only hold one package at a time.

The following image shows what the lockers at the post office currently look like. Lockers marked with an X are holding a package.



When a new package arrives, it is placed in the lowest-numbered available locker in which it can fit. When a customer arrives to pick up a package from a locker, the locker becomes available again.

The post office has opened for the day and the following five events occur in this order:

- Four small packages arrive.
- The packages in lockers 11 and 19 are picked up.
- Two medium packages arrive.
- The packages in lockers 20 and 21 are picked up.
- Two small packages arrive.

Question

Then one more small package arrives. In which locker is it placed?

(A) 20

(B) 19

(C) 24

(D) 17

Answer

(A) 20

Explanation of Answer

We can keep track of the available lockers before and after each of the five events.

Event	**Notes**	**Available Lockers**
Post office opens		10, 14, 17, 21, 24, 30, 31
4 small packages arrive	They occupy lockers 10, 14, 17, 21	24, 30, 31
Pickup from lockers 11 and 19	This frees lockers 11, 19	11, 19, 24, 30, 31
2 medium packages arrive	They occupy lockers 24, 30	11, 19, 31
Pickup from lockers 20 and 21	This frees lockers 20, 21	11, 19, 20, 21, 31
2 small packages arrive	They occupy lockers 11, 19	20, 21, 31

The small package that arrives next is placed in the lowest-numbered available locker, which is locker number 20.

Connections to Computer Science

The key computer science concept outlined in this task is *memory allocation strategies*. Memory allocation strategies are used by the *operating system*, which executes various *applications* or *programs* and assigns each application/program an amount of memory in *random access memory (RAM)*. One such strategy is the *best-fit* strategy, where the smallest available block of memory that is large enough for the program/application is the one that is chosen. The best-fit strategy is what is used in this task: the lowest-numbered empty locker that is large enough to hold the package is chosen.

There are other allocation strategies, such as *first-first*, where the first available slot is chosen, and *worst-fit*, where the largest available slot is chosen. It is worth noting that for **any** allocation strategy, there is a sequence of allocations and deallocations that will cause an *out of memory* error, even though there is enough available memory. For example, in this task, if 7 small packages arrive, then all lockers will be allocated. Then, even if all the lockers from 10 to 19 are emptied, there is no room for a medium sized package. When this happens, we say that memory has been *fragmented*. In operating systems, fragmentation can be be resolved by *paging* blocks of allocated memory into smaller units, or by *compacting* blocks to free up memory.

Country of Original Author

Iceland

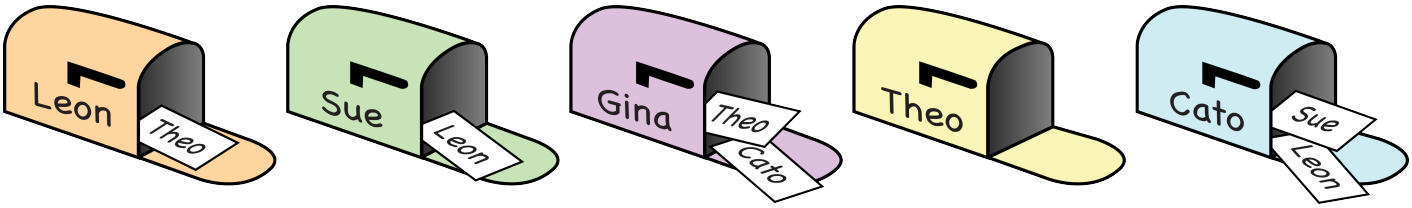
Delivering Mail

Story

When beavers want to send mail to each other they leave the mail in their own mailbox. Percy, the mail-delivery beaver, then does two things at each mailbox:

- He collects any mail in the mailbox and puts it in his bag.
- He delivers all the mail for that beaver from his bag by putting it in their mailbox.

At the start of Percy's shift, his bag is empty and the beavers' mailboxes contain the following mail:



At the end of Percy's shift, all the mail has been delivered. Percy also notes that he only had to visit each mailbox exactly once (which is unusual for Percy).

Question

In which of the following orders must Percy have visited the mailboxes?

(A) Gina $\longrightarrow$ Cato $\longrightarrow$ Leon $\longrightarrow$ Sue $\longrightarrow$ Theo

(B) Gina $\longrightarrow$ Sue $\longrightarrow$ Cato $\longrightarrow$ Theo $\longrightarrow$ Leon

(C) Gina $\longrightarrow$ Cato $\longrightarrow$ Sue $\longrightarrow$ Leon $\longrightarrow$ Theo

(D) Cato $\longrightarrow$ Gina $\longrightarrow$ Sue $\longrightarrow$ Leon $\longrightarrow$ Theo

Answer

(C) Gina $\longrightarrow$ Cato $\longrightarrow$ Sue $\longrightarrow$ Leon $\longrightarrow$ Theo

Explanation of Answer

Since Percy begins his shift with an empty bag and visits each mailbox exactly once, it follows that when Percy visits a beaver's mailbox, no other mailboxes can contain mail for that beaver.

The first mailbox visited must belong to a beaver who is not receiving any mail. This can only be Gina.

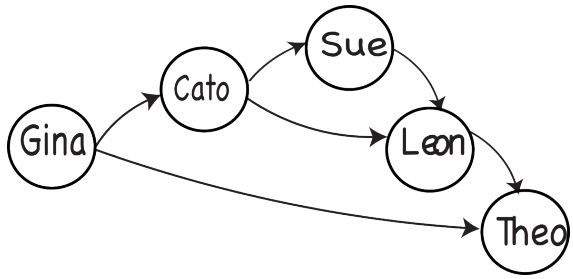
Now the remaining mailboxes contain mail for Theo, Leon, and Sue. Since Cato is the only remaining beaver without mail for them in a mailbox, Percy must have visited Cato next.

Now Leon, Sue, and Theo's mailboxes are left. Sue's mailbox has mail for Leon, and Leon's mailbox has mail for Theo. Therefore, Percy must have visited these three mailboxes in the order Sue, Leon, and Theo.

Therefore, Percy must have visited the mailboxes in the order:
Gina $\longrightarrow$ Cato $\longrightarrow$ Sue $\longrightarrow$ Leon $\longrightarrow$ Theo

Connections to Computer Science

The mail that is initially in the mailboxes before Percy starts delivering the mail provides information about the order in which Percy can visit the mailboxes. We can put this information together to create the following diagram called a *graph*. A graph consists of *nodes*, which in this diagram are the circles with names, and *(directed) edges* between the nodes, indicated as arrows from one node to another. In this graph, an arrow from node A to node B means person A wants to send mail to person B.



This diagram is called a *directed acyclic graph*, or DAG for short. A graph is "acyclic" because it doesn't contain any *cycles*: that is, there is no path starting and ending at the same node. A DAG is commonly used to describe relationships in computer science, such as for *scheduling*, *parallel processing*, and *data compression*. This task asks to find the order in which Percy must visit the mailboxes. This is called a *topological ordering* of the corresponding DAG, where the names are arranged in a line with all arrows pointing forward.

A good exercise is to show that there is only one topological ordering for this DAG. That is, show why Percy *must* visit the mailboxes in the order given by Option (C).

DAGs are used everyday to solve task scheduling problems. For example, cooking usually requires multiple tasks. We make a mental list of the order of the tasks: some tasks cannot start until others are completed, while others can start at any time.

Country of Original Author

Canada

Logic Treasure

Story

An island contains three treasure chests: one by the mountains, one under the palm tree, and one on the beach. At the start of the day all three treasure chests were empty. Then, at some point during the day, Pirate Beaverbeard filled one of the chests with gold.

Three treasure hunters explored the island. One of them did all of their exploring *before* Beaverbeard filled a chest with gold. The other two treasure hunters did all of their exploring *after* Beaverbeard filled a chest with gold. None of the treasure hunters ever found the gold; they only discovered empty chests, as shown below.

Treasure Hunter	**Empty Chest(s) Discovered**
Alice	on the beach
Bob	on the beach / under the palm tree
Clark	by the mountains / under the palm tree

Question

Which treasure chest was the gold in?

(A) The chest under the palm tree

(B) The chest on the beach

(C) The chest by the mountains

(D) The chest cannot be determined

Answer

(C) The chest by the mountains

Explanation of Answer

Since there are three possible places for the gold to be, we can explore each possibility to see which one satisfies the details of the story.

If the gold was in the chest under the palm tree, then both Bob and Clark must have explored the island before Pirate Beaverbeard arrived. The story tells us that only one treasure hunter explored the island before, so this is a contradiction. The chest under the palm tree cannot contain the gold.

If the gold was in the chest on the beach, then both Alice and Bob explored the island before Pirate Beaverbeard arrived. This again contradicts the story. The chest on the beach cannot contain the gold.

If the gold was in the chest by the mountains, then only Clark needs to have explored the island before Pirate Beaverbeard arrived. Assuming that Alice and Bob explored the island after, the story is satisfied.

Therefore, Pirate Beaverbeard filled the chest by the mountains with gold.

Connections to Computer Science

This task involves *logical reasoning*. Specifically, reasoning about the three chests using logical expressions involving *AND*, *OR*, or *NOT*.

Each observation can be thought of as a NOT expression: that is, the chests examined by the treasure hunters are known to NOT contain treasure when they opened the chests.

The information gathered by each person can be connected together with an AND. For example, Clark states that there is no treasure in the chest by the mountains AND no treasure in the chest under the palm tree.

We know that one of the chests will have gold at some point. In other words, either the gold is in the chest on the beach OR in the chest under the palm tree OR in the chest by the mountains.

Logic plays a key role in computer science in a huge variety of areas: databases, programming languages, artificial intelligence, hardware and software design and verification, etc. Logic is one of the fundamental concepts that underlies *computational thinking*: being able to take information, model it logically, and make *logical conclusions* from that model.

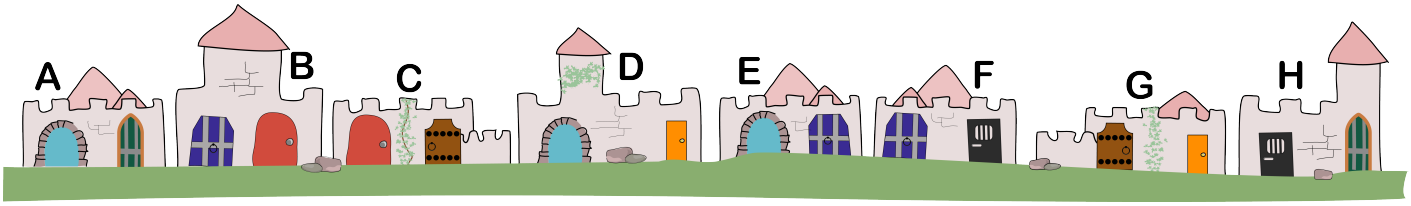
Country of Original Author

Uruguay

Magical Doors

Story

There are eight buildings, labelled A through H, along a road as shown below.



The only way to travel between the buildings is by using magical doors. There are seven different types of doors:

Each building has two different doors. When you exit a building through one of its doors, you can then enter any of the other buildings that have a door of the same type.

For example, if you exit building A via the leftmost door , then you can enter either building D or building E, and if you exit building A via the rightmost door , then you will enter building H.

Question

If you passed through the fewest buildings possible starting in building A and ending in building C, how many **types of doors** did you travel through?

(A) 2

(B) 3

(C) 4

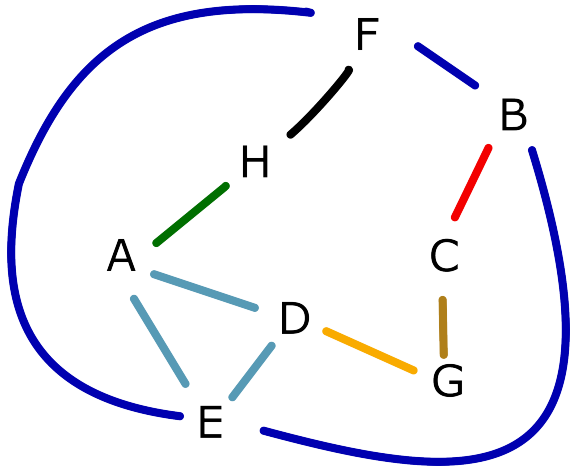
(D) 5

Answer

(B) 3

Explanation of Answer

The following diagram can help us solve and visualize this problem. It consists only of the labels of the buildings and lines connecting two buildings exactly when they have a door of the same type. Notice how the colours of these lines match the colour of the corresponding door.



Now it is easier to see that the only buildings you can reach from building A using one door are buildings H, D and E. From there, the only new buildings you can reach using a total of two types of doors are buildings F, G and B. This means that you cannot travel from building A to C using fewer than three types of doors. However, you can travel from building A to building C using a total of three types of doors. One way to do this is to travel from building A to building D and then to building G before using a third type of door to enter building C.

Connections to Computer Science

As shown in the Explanation of Answer, this task involves finding the *shortest path* in a *graph*. Graphs are composed of a set of *nodes* and a set of *edges* between nodes. In this task, the nodes are the buildings, and the edges connect two buildings which have a door of the same type.

One major application of graphs is to represent maps for driving: each street address is a node and roads are edges. When driving, finding the shortest path is the most common problem to solve. There are several *graph algorithms* which can help us find the shortest path. Two examples of shortest path algorithms are *Dijkstra's algorithm* (used in common GPS systems) and the *Warshall-Floyd algorithm* (used in Google Maps).

Country of Original Author

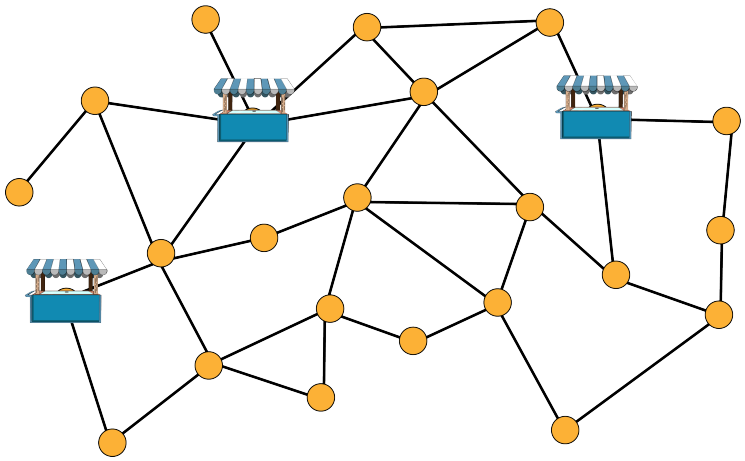
Japan

Juice Carts

Story

To celebrate National Juice Day, three juice carts have been placed in a huge park. The juice carts will distribute free juice.

The image below shows a map of the park. The juice cart icons represent points of interest where a juice cart has been placed, circles represent other points of interest, and the lines represent paths.



It takes 5 minutes to walk between any two points of interest which are directly connected by a path. There are some points of interest from which you have to walk more than ten minutes in order to reach a juice cart. This is considered too far to walk for free juice. Therefore, a point of interest must be chosen for another juice cart. After this fourth juice cart is placed, it must be possible to reach a juice cart from any point of interest in at most 10 minutes.

Question

How many different points of interest can be chosen for the location of the fourth juice cart?

(A) 1

(B) 2

(C) 3

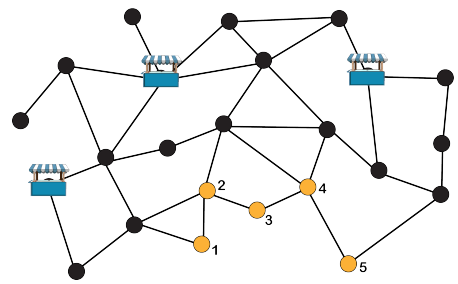
(D) 4

Answer

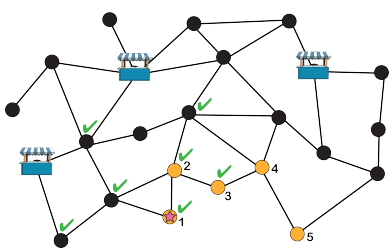
(B) 2

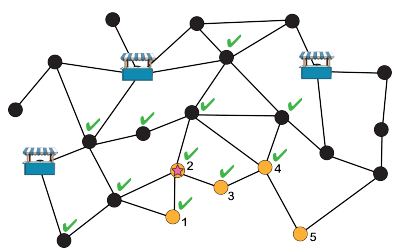
Explanation of Answer

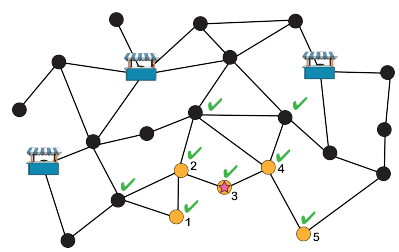
The points of interest where you can currently get free juice by walking at most ten minutes have been coloured black. The five points of interest where you must walk more than ten minutes are numbered. The placement of another juice cart must make it possible to get free juice by walking at most ten minutes from these five points of interest.



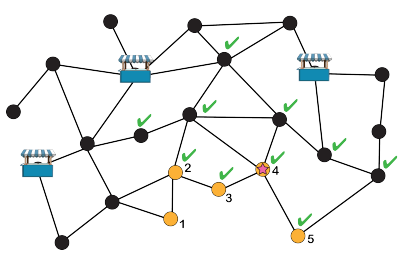
To get free juice from point of interest 1, a juice cart must be placed either on point of interest 1, or five or ten minutes away. The first picture below shows all these places using checkmarks. The second picture shows all the places where a juice cart could go in order to get free juice from point of interest 2. The third picture shows all the places where a juice cart could go in order to get free juice from point of interest 3.

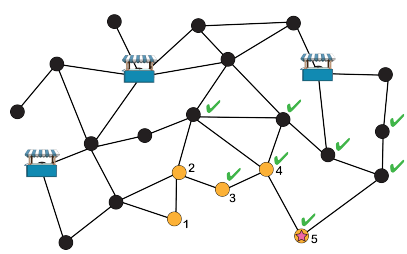






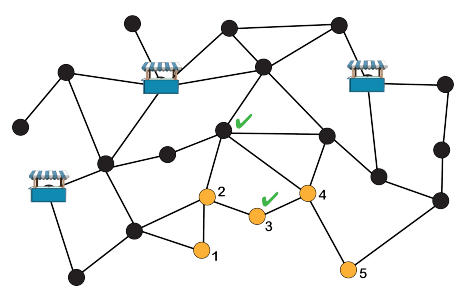
The first picture below shows all the places where a juice cart could go in order to get free juice from point of interest 4. The second picture shows all the places where a juice cart could go in order to get free juice from point of interest 5.





There are two points of interest that have checkmarks in all five of the pictures. These represent the places where a juice cart could go so that it is possible to reach a juice cart from any point of interest in at most 10 minutes.

Therefore, there are two different choices for the placement of the fourth juice cart.



Connections to Computer Science

This task deals with *graphs*. A graph is composed of a set of *nodes* and a set of *edges* connecting pairs of nodes. Graphs are used in many different contexts: for example, streets in maps, relationships between people, states in a closed system, or words organized hierarchically.

Some properties of graphs are relatively easy to determine, such as whether a graph is *connected*. Other properties of graphs can be more difficult to determine. In this task, we are trying to determine a variant of a *dominating set* of nodes. A set of nodes is dominating if every node in the graph is in the dominating set or is connected by an edge to a node in the dominating set. In this task, we looking for a *2-distance dominating set*. A 2-distance dominating set has the property that evey node is at most distance two away from a node in the set. That is, a node is either in the set, connected by an edge to a node in the set, or connected by an edge to a node which is connected by another edge to a node in the set. Finding the size of the smallest dominating set is a very difficult problem: it is in a category of problems called *NP-complete*, for which it is believed there is no efficient algorithm to solve the problem.

Country of Original Author

Brazil