




2023-CA-01 Besonderer Baum

6yo–8yo: —	8yo–10yo: hard	10yo–12yo: medium	12yo–14yo: easy	14yo–16yo: —	16yo–19yo: —
Answer Type: multiple choice					
Computer Science Areas: <input checked="" type="checkbox"/> algorithms and programming <input type="checkbox"/> data structures and representations			<input type="checkbox"/> computer processes and hardware <input type="checkbox"/> communication and networking <input type="checkbox"/> interactions, systems and society		
Equivalent Tasks: --					

Body

Ben hat einen besonderen Apfelbaum im Garten:

- Landet ein Vogel  auf dem Baum, wachsen sofort zwei neue Äpfel.
- Klettert ein Eichhörnchen  auf den Baum, fällt ein Apfel runter. Wenn kein Apfel am Baum hängt, passiert nichts.
- Besucht eine Schlange  den Baum, verschwinden alle Äpfel sofort.

Heute Morgen hängen 25 Äpfel am Baum. Dann besuchen einige Tiere nacheinander den Baum, zuletzt ein Eichhörnchen. Ben hat ihre Reihenfolge genau aufgeschrieben:



Question/Challenge

Wie viele Äpfel hängen danach am Baum?






Answer Options/Interactivity Description








- A) 3 Äpfel
- B) 7 Äpfel
- C) 17 Äpfel
- D) 31 Äpfel






Answer Explanation

Antwort B ist richtig. Nachdem das letzte Eichhörnchen auf den Baum klettert, hängen noch 7 Äpfel am Baum.

Man kann für jeden Tierbesuch ausrechnen, wie viele Äpfel im Moment am Baum hängen:

Tier:	Start					
Anweisung:	–	+2	+2	-1	+2	reset
Anzahl Äpfel:	25	27	29	28	30	0

Tier:	Übertrag								
Anweisung:	–	–	–	+2	+2	+2	-1	+2	reset
Anzahl Äpfel:	0	0	0	2	4	6	5	7	0

Tier:	Übertrag					
Anweisung:	–	+2	+2	+2	+2	-1
Anzahl Äpfel:	0	2	4	6	8	7

Da alle Äpfel sofort verschwinden, wenn eine Schlange den Baum besucht, können wir alles ignorieren, was vor der Ankunft der zweiten (und letzten) Schlange passiert. Wie in der Tabelle gezeigt, landen nach dem Besuch der letzten Schlange vier Vögel auf dem Baum. Danach hängen am Baum $4 \times 2 = 8$ Äpfel. Dann klettert ein Eichhörnchen auf den Baum, wodurch ein Apfel herunterfällt und $8 - 1 = 7$ Äpfel übrigbleiben.

This is Informatics

Der Besuch eines Tieres verändert den Zustand des magischen Apfelbaums – aber nur auf ganz bestimmte Weise: Nur die Anzahl der Äpfel, die am Baum hängen, wird geändert. Auf andere Eigenschaften des Baumes, etwa die Anzahl der Blätter, die Länge einzelner Äste oder die Form der Baumkrone, hat der Besuch eines Tieres keinen Einfluss. Für diese Biberaufgabe ist es also ausreichend, die Anzahl der Äpfel zu betrachten.

Auch ein Computerprogramm hat einen Zustand, der von den einzelnen Anweisungen des Programms verändert wird. Als Zustand werden meist die von einem Programm gespeicherten Daten betrachtet; diese Daten speichert das Programm in den bei der Programmierung eingeführten *Variablen*.

Die Folge der Tierbesuche auf dem Baum in dieser Biberaufgabe ist wie ein Computerprogramm: Jeder Tierbesuch ist eine Anweisung, die den Zustand des Apfelbaums verändert. Dieser Zustand – also die Anzahl der Äpfel, siehe oben – kann in einer einzigen Variable gespeichert werden.

Beim Lösen der Aufgabe ist dir vielleicht aufgefallen, dass du nicht das ganze “Programm” anschauen musstest, sondern nur den Teil nach dem letzten Vorkommen der Schlange. Durch genaue Betrachtung der Auswirkungen der einzelnen Anweisungen auf den Zustand des Programms konntest du eine besondere Eigenschaft des Programms herausfinden. Eine solche Analyse von (Computer-)Programmen gehört zu den häufigen Tätigkeiten von Informatikerinnen und Informatikern.

This is Computational Thinking

The solution of the task requires algorithmic thinking.

Informatics Keywords and Websites

- Variablen: [https://de.wikipedia.org/wiki/Variable_\(Programmierung\)](https://de.wikipedia.org/wiki/Variable_(Programmierung))
- (Programm-)Zustand: <https://media.kswillisau.ch/in/zustand/zustand.html>

Computational Thinking Keywords and Websites

- Abstraction: [https://en.wikipedia.org/wiki/Abstraction_\(computer_science\)](https://en.wikipedia.org/wiki/Abstraction_(computer_science))
- Algorithms: <https://en.wikipedia.org/wiki/Algorithm>

Wording and Phrases

- Äpfel wachsen - Äpfel werden dem Baum hinzugefügt
- Äpfel fallen - Äpfel werden abgezogen
- Äpfel verschwinden - alle Äpfel werden ohne Verzögerung “gelöscht”

Comments

- **WGL1:** Valentina Dagienè and Bence Gaál, 2023-05-15: *TiCT part and related keywords have been added with sources. We also add a new keyword: instantly.*
- **Pre-Workshop Review by Jean-Philippe Pellet:** I think it can work as ‘hard’ for 8-10, too. Very nice idea and implementation. Keeping track of state is essential in programming. Having a long list of ‘commands’, most of which can be ignored by a smart solving process is really good. The math is little, but the CT is highly present, so this is a really good task IMO. As a non-native speaker, I find the ‘their’ possessive for a single person to be rather confusing from a clarity point of view. Explanation is complete and concise. The link to the variable is clear. Between that and the condition, one could

mention a program as a list of instructions. To me, the link to program is clearer than the one to the condition. One would need the condition to actually implement a small interpreter for the 3-symbol minilanguage invented here, but it doesn't seem to be central here. Graphics are nice and UXWing licence seems totally appropriate. [TiCT] Missing.

- **Author Response:**

- I have adjusted the age group difficulty levels.
- The use of a singular 'their' was intentional in order to present a diverse task that includes a gender-neutral pronoun. Countries are welcome to change it if that pronoun is not well-understood but I prefer that the change be made on a local level after the workshop.
- I agree there are other informatics connections that could be made but my preference is to focus/highlight just one or two rather than overwhelm the reader with a buffet of ideas.
- Since the TiCT section was optional I opted to leave it blank.

- **Pre-Workshop Review by Raluca Constantinescu:** Age/difficult level specified and appropriate for the task. Very interesting idea, very attractive. The text is very clear and understandable. Words and expressions are used consistently in the entire task. [EXPL] Available and complete. Very good answer explanation. [Til] Available and very well and clearly described. All the figures are clear and appealing. They are suitable for the age group.

- **Author Response:**

- Thank you for the positive review.

- **Pre-Workshop Review by Mărtiș Opmanis:** It seems that task is appropriate also for the second younger (8 - 10 yo.) age group. Easy task with nice catch that you should count apples only after the last appearance of a snake. Tale is a bit artificial (bird's visit leads to appear apples).

- **Author Response:**

- I have adjusted the age group difficulty levels.
- The task is artificial, but many Bebras tasks are. The artificialness helps to emphasize the informatics being presented. I don't think having a silly story is a reason to downrate a task.

- **Pre-Workshop Review by Atheer Khabti:** Very interesting idea, I really like it. [TEXT] Clear and understandable. [EXPL] Very good. [Til] Very good. In the body section, the three options that can change the tree can be written in 3 points under each other to be more clear for younger ages.

- **Author Response:**

- I tried writing the body section using a bulleted list as you suggested, but in the end decided it was shorter and easier to read in paragraph form.

- **Pre-Workshop Review by Samuel Rosario:** The age and the level are appropriate. It is a good concept to bring early computation and algorithms. The text of the task is well written. The task explanation is complete and direct. The size of the symbols representing the animal should be bigger.
- **Author Response:**
 - I increased the size of the images slightly.

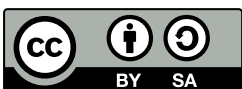
Contributors

- Sarah Chan, sarah.chan@uwaterloo.ca, Canada (author,graphics)
- Valentina Dagienė, valentina.dagiene@mif.vu.lt, Lithuania (contributor)
- Bence Gaál, gaalbenc@inf.elte.hu, Hungary (contributor)
- Christian Datzko, christian@bebras.services, Hungary (contributor)
- Wilfried Baumann, baumann@ocg.at, Austria (translation from English into German)
- Susanne Datzko-Thut, datzko@bwinf.ch, Germany (graphics)
- Wolfgang Pohl, pohl@bwinf.de, Germany (contributor)

Support Files

- *.svg from <https://uxwing.com/> (CC BY-SA 4.0), improved by Sarah Chan, improved by Susanne Datzko-Thut

License



Copyright © 2023 Bebras – International Contest on Informatics and Computer Fluency. This work is licensed under a Creative Commons Attribution – ShareAlike 4.0 International License. <https://creativecommons.org/licenses/by-sa/4.0/>