Exercise 1

In a service-oriented architecture:

A. Services are independent from each other and communicate via interfaces (x)
B. Services only expose REST APIs
C. Every service is explicitly exposed to the user
D. Horizontally scaling is not possible

Exercise 2

A Webservice is a software that:

A. Uses only REST as its underlying technology
B. Exposes functionality that is accessed using the HTTP protocol (x)
C. Can only be called from web applications
D. Uses only JSON as payload format

3. SOA

A RESTful service uses an endpoint /api/accounts to access, create, retrieve and update bank account data. This endpoint is the only one that exists for dealing with accout data. How would a client application update account information using this service in a RESTful way?

A. PUT /api/accounts/:id (x)
B. GET /api/accounts
C. POST /api/accounts/:id
D. GET /api/accounts/update

https://forms.office.com/r/zhGJPgvbBJ

Exercise 4

In a Dockerfile the following line is contained:

EXPOSE 7999

This means that:

A. The application will automatically accept connections on the host via port 7999
B. The application listens in the container port 7999 (x)
C. Port 7999 is critical and has to be protected using SSH

Exercise 5

In a data analysis application, which is implemented using MapReduce, there are 100 images of animals which are classified into images of dogs, cats, birds and reptiles. The application counts how many images are there of which type. How many executions of the Map and Reduce functions each can be run in parallel at most?

A. 4 executions of Map and 100 executions of Reduce
B. 104 executions of Map and 4 executions of Reduce
C. 100 executions of Map and 4 executions of Reduce (x)
D. 208 executions of Map and 4 executions of Reduce

Exercise 6

A service in a service-oriented architecture is required to process as many requests per second as possible (maximize throughput). The processed data is not particularly critical regarding data protection and security. What would be the most efficient way of implementing a request in such a service?

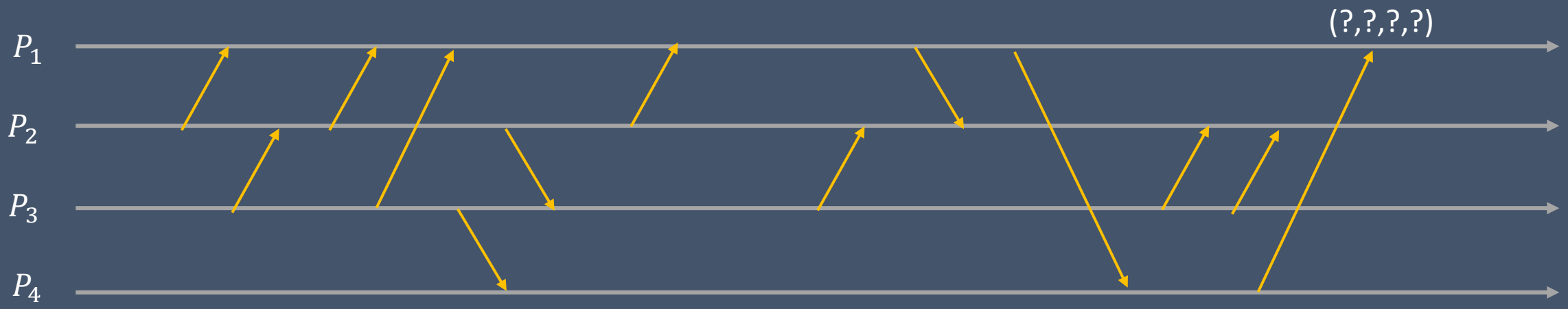
A. Using threads, because context-switch is more lightweight (x)
B. Using processes, because context-switch is handled by the operating system
C. Using virtual machines, because they handle every aspect in a transparent way

Exercise 7

In a Chord Distributed Hash Table (DHT) with a name space of $m = 4$ bits there are exactly 8 nodes evenly spaced (with IDs = 0, 2, 4, …, 14). Assuming that the finger table of each node is already constructed, how many jumps are needed to look up the value 11 from node 2?

A. 4
B. 2 (x)
C. 1
D. 0

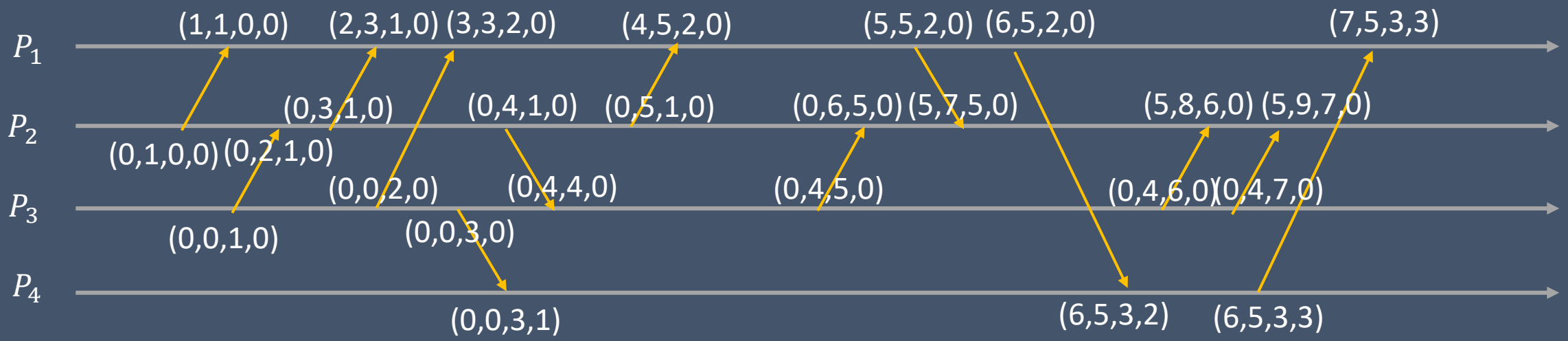
Exercise 8



What is the value of the final vector clock of P_1?

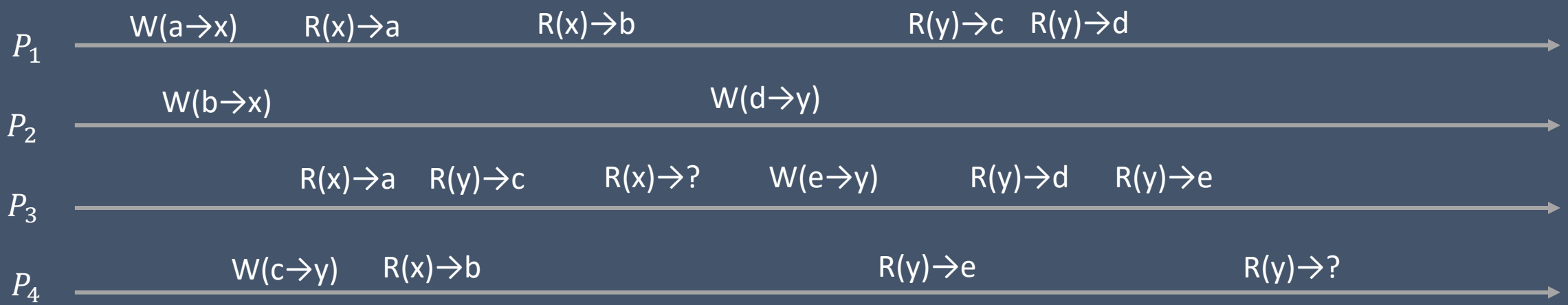
A. (6,5,4,3)
B. (7,5,4,3)
C. (7,5,3,3) (x)
D. (6,5,5,3)

Exercise 8 (Explanation)



Exercise 9

Let R(x)→a denote a read operation for the value a of variable x from a data store. Let W(a→x) denote a write operation where the value a is written to the variable x. Consider the following diagram:

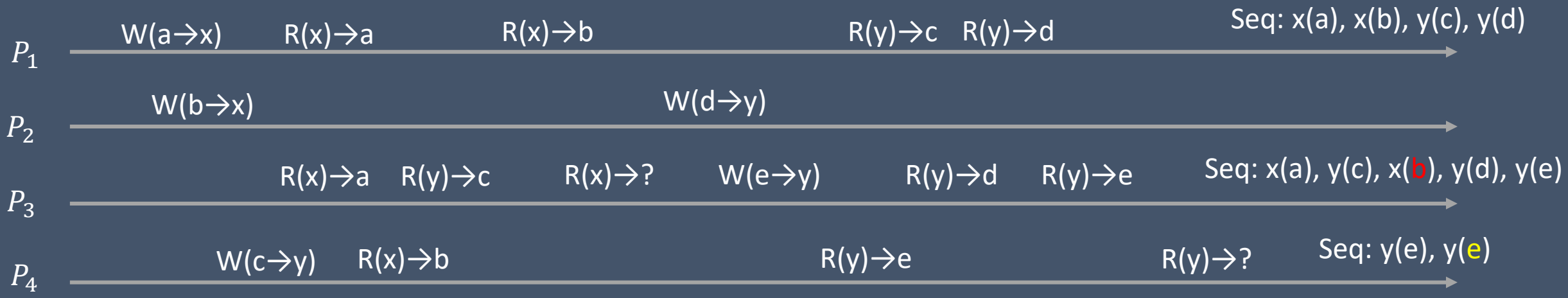


What are the required values for x and y to satisfy sequential consistency?

A. R(x)→b, R(y)→d
B. R(x)→b, R(y)→e (x)
C. R(x)→a, R(y)→e
D. R(x)→b, R(y)→c

Exercise 9 (Explanation)

Let R(x)→a denote a read operation for the value a of variable x from a data store. Let W(a→x) denote a write operation where the value a is written to the variable x. Consider the following diagram:



What are the required values for x and y to satisfy sequential consistency?

A. R(x)→b, R(y)→d
B. R(x)→b, R(y)→e (x)
C. R(x)→a, R(y)→e
D. R(x)→b, R(y)→c

- *The value of x has to be b because P_3 has already seen the value c of y and P_1 determines that before reading c, the value b was already read from x.*
- *The value of y has to be e because the value e has already been read before and P_1 and P_3 combined see y(c), y(d), y(e)*

Exercise 10

Consider the basic Paxos algorithm. In one instance, assume that there are 2 proposers P_1 and P_2, 3 acceptors A_1, A_2 and A_3 and 3 learners L_1, L_2 and L_3. A proposal with ID 1 and timestamp 1 from proposer P_1 was promised by acceptors A_1 and A_3. What happens if P_1 fails now?

A. The algorithm halts until P_1 restores
B. As soon as P_2 sends a new proposal, it will be accepted by the acceptors
C. As soon as P_2 sends a new proposal, a majority of acceptors will inform P_2 that a proposal with a lower timestamp was already promised.(x)
D. The learners will execute proposal with ID 1 because it was already accepted.