CMI/2024/Mid-Semester Exam 01/03/2024

Distributed Computing and Big Data

Chennai Mathematical Institute

Duration: 90 Mins. Max: 25 Marks.

Instructions

- This is a closed book exam.
- This is an individual task. Do not discuss with anyone.
- No electronic devices are allowed. Wherever heavy calculation is involved, you need not evaluate it to the final number unless it is explicitly asked for. For example, it is acceptable to leave the answer as $\frac{1}{1+\frac{5}{32}}$. You need not evaluate it to 0.865.
- Clearly mention your name and roll number in your answer sheet.

Section 1: Correct answers carry 1 mark each. Answer True/False. Wrong answers carry -0.5 marks each.

Question 1. In the model of the distributed system as discussed in the class, there is no common global memory. **True**/False?

Question 2. While computing average access time, head switching time is often considered negligible. **True**/False?

Question 3. Data lakes are schemaless. **True**/False?

Question 4. Grid computing infrastructure refers to the use of heterogeneous systems. **True**/False?

Question 5. Scalar time is strongly consistent. True/**False**?

Question 6. Hadoop uses a Write Once and Read Once model. True/**False**?

Question 7. Hadoop data nodes send periodic heartbeat signals and block reports to name node. **True**/False?

Section 2: Correct answers carry 2 marks each. No negative marks.

Question 8. Assume disk size = 512 GB, block size = 8 KB. How much space (in MB) will we need to store the free space bitmap?

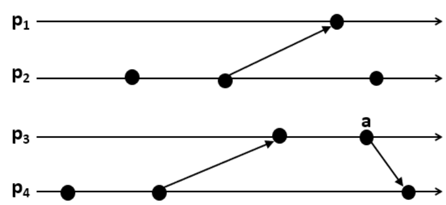
(Solution) 512 GB = $2^9 * 2^{30}$ Bytes. Block size is $2^3 * 2^{10}$ Bytes. For each block, we use one bit in the free space bitmap. Therefore, we need $\frac{2^{39}}{2^{13}}$ bits. We need 2^{26} bits = $\frac{2^{26}}{2^3 * 2^{20}}$ = 8 MB.

Question 9. As per Amdahl's law, What is the best achievable speed up if only 25% of the job can be parallelized, and we have 4 processors?

(Solution) R = 75% cannot be parallized i.e., $\frac{3}{4}$. p = 4.

Best speedup = $\frac{1}{\frac{3}{4}+\frac{1-\frac{3}{4}}{4}=16/13} = 1.23x$

Question 10. If we were to annotate the following space-time execution diagram with vector time stamps, how would we annotate the event marked as 'a'?



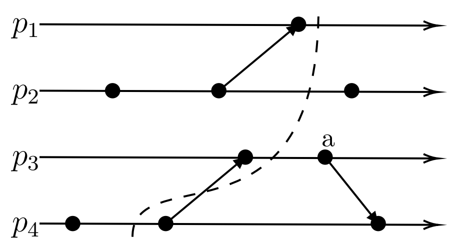
(Solution) $\begin{bmatrix} 0 \\ 0 \\ 2 \\ 2 \end{bmatrix}$

Question 11. For the same diagram given above, annotate the events in p_3 with scalar time.

(Solution) 3 and 4.

Question 12. For the same diagram given above, identify an inconsistent cut not involving the event marked as 'a' i.e., 'a' must be in the future of the cut.

(Sample Solution)



Question 13. In the muddy children puzzle, as discussed in the class, what would the children say during the first and second rounds if n=4 and k=3? i.e., there are four children, and three of them have muddy forehead and to start with, they are told that at least one of them have muddy forehead. Assume that each child can only say 'yes', 'no' or 'dont know'. Use the following format to provide your answer.

```
Round1: <1st child response>,<2nd ...>,<3rd ...>,<4th ...>
Round2: <1st child response>,<2nd ...>,<3rd ...>,<4th ...>
```

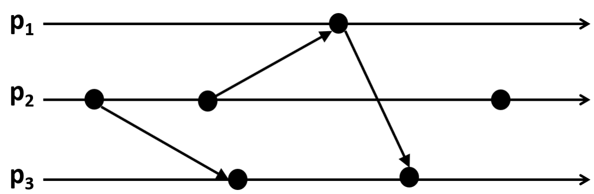
(Solution)

Round1: DN DN DN DN

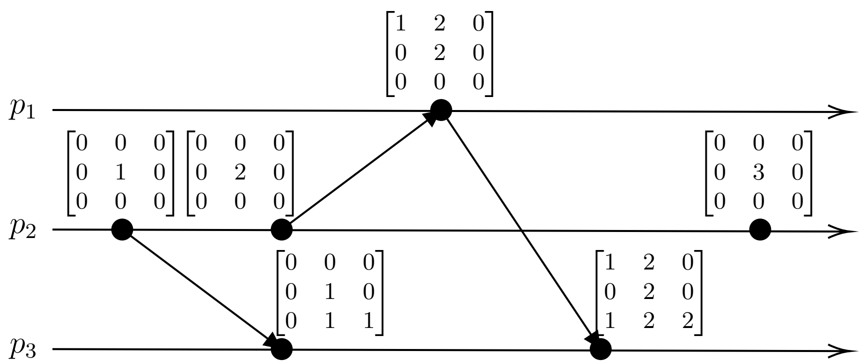
Round2: DN DN DN DN

Section 3: Question carries 3 marks. No negative marks.

Question 14. Annotate all the events in the following diagram with matrix time.



(Solution)



Question 15. You are given a large text file. You need to find all the distinct words that have no vowels. For example, if the input text file has the content, “my phone is ringing and ringing again”, there are six distinct words (my, phone, is, ringing, and, again). The word without vowel is only one, “my”. Therefore, the output should be a single word, “my”.

Describe an approach using map reduce logic. No need to write any code.

(Sample Solution)

Map: Mapper breaks the input text into words, for each word checks if it has vowels, if not emits a key-value pair with the word as key and 0 as value (value could be anything as it is not relevant to the reducer). Example:

my cat can fly =[Mapper]=> <my, 0> <fly, 0>

my dog can spy =[Mapper]=> <my, 0> <spy, 0>

Reduce: After shuffle and sort, the key-value pairs are combined (using Combiner), where all pairs with the same key are combined to a new key-value pair where the key is the same and the value is the list of prior values. This is the input to the reducer, so it just needs to output the keys, which are already distinct. Example:

<fly, {0}> <my, {0, 0}> <spy, {0}> =[Reducer]=> fly my spy