Q1. What is the problem the authors are aiming to solve in this article? (0.5 point)

The author was trying to manage massive and complex computing infrastructure that are running significant amount of application and programs. As per the article, it will be the chances of correlated failures and minimize the fault recovery time as well.

Q2. In Section 2 the authors state "many end-user-facing service jobs see a diurnal usage pattern" What is a diurnal usage pattern? Can you provide one example (0.5 points)

The diurnal cycle refers to the pattern or cycle that repeats with in the specific period of the time. The example of the diurnal usage pattern could be internet consumption rate of any area which remains at the peak at the day time and decreases during the night.

Q3. The authors state that "Borg programs are statically linked to reduce dependencies on their runtime environment." The following contains a snippet of a C++ program. Can you compile the code and provide the size of the binary with a) static linking and then b) without static linking? What is the reason for the size difference? (0.5 points)

```
# include <iostream>
int main() {
std::cout << "Hello World" << std::endl;
return 0;
}
```

```
bivek@bivek-ThinkBook-14-G2-ITL:~/Desktop$ g++ -static hello.cpp -o hello_static
bivek@bivek-ThinkBook-14-G2-ITL:~/Desktop$ g++ hello.cpp -o hello_dynamic
bivek@bivek-ThinkBook-14-G2-ITL:~/Desktop$ ls -lh hello_static hello_dynamic
-rwxrwxr-x 1 bivek bivek  17K Feb 15 20:16 hello_dynamic
-rwxrwxr-x 1 bivek bivek 2.3M Feb 15 20:16 hello_static
bivek@bivek-ThinkBook-14-G2-ITL:~/Desktop$
```

The size of the binary with the static linking is 2.3M where as the size of binary with dynamic linking is 17K. The main reason of the size difference is, In dynamic linking, binary contains reference to the required libraries which are loaded at runtime. However in static linking all libraries are directly included in binary.

Q4. Please describe Figure 1 in 5-10 sentences (2.5 points)

The given figure is related to very high overview of borg architecture. It is showing different components and their relationships as well as interaction mechanism with in

the borg architecture. The figure also illustrates some communication channels between borglets and borgmaster. Borgmaster seems to be the core that manages all the operation and execution including scheduling, persistent storage, etc. From the figure it seems that use can have some kind of interface through which use submit jobs . Also it can be possible to submit via programmatic APIs.

Q5. Describe Figure 5 (b) and explain why are we seeing steps in the CDF and what do the error bars represent? (1.5 points)

The CDF graphs are forming the steps that may be because of the timeout limit.

They will stop handling workload after certain time has been elapsed. It totally depends on the workload on the machines. The errors bar represent the complete range of values from the trial performed. It provides the range of values used in the trails for prod and non prod workloads.

Q6. In Figure 8, we observe that the X-axis represents the "Requested limit in cores" and we see the X-axis range from 0.01 to 10000. As mentioned in 5.4 they use a unit of milli-cores. What is a milli-core? (0.5 points)

The term milli-cores refers to the unit especially used to indicate the CPU resource allocation. It is found that the using the milli-core as the resource allocation unit enables efficient and flexible resource allocation.

Q7. Summarize the results of Section 5 in 5-6 sentences. Please describe which subset of results you picked and why?(2 points)

The section includes several topics including resource utilization, task completion graphs and jobs scheduling policies. It shows that borg system provides the high level of resource utilization with the minimum amount of resource waste. The section demonstrated that borg system is able to scale effectively to very large cluster. The section also talked about evaluation methods including cell compaction, which identifies smallest cell size for the particular workload.

Q8. How can Borg use the results of OpenFlow, F10, and VXLans? (1 point)

Borg can use the results to Openflow, F10 and VXLans to improvise the performance and efficiency of the network. Using those results borg can ensure the high availability and optimize the traffic routing. Similarly, borg can use the fault tolerant mechanism

specified by the results of openflow article. Also the openflow could monitor the flows of the network. F10 can be used for the security and inspection of packets and VXLans can be used to provide flexibility and inter network connection.

Q9. What are the key weaknesses of this work in your opinion, and how would you address them? (1 point)

The conclusion of the article is really short and unclear. Majority of articles and research paper provides the clear conclusion so that, the conclusion part can clearly reflect the summarize view of the entire article. However, this paper is missing that. The author have not mentioned anything clearly in lesson learned section. I would have better written the conclusion in much better way. So that it can reflect overall articles.