UMASS CICS

CS677 Spring 2019

Prof. Prashant Shenoy

Assignment #2

Due Date: Feb 25, 2019 23:55 hrs.

Submission Instructions

- Generate a PDF solution and upload it to Gradescope
- Please make sure you highlight your solutions appropriately i.e. Mark which pages of your PDF solution correspond to questions of the assignment
- Check here for helpful tips on uploading PDF assignments to Gradescope: https://gradescope.com/help#help-center-section-student-workflow

Part 1

1. What is it important to respect cache affinities of processes and threads in multiprocessor scheduling?
2. Explain in a few sentences as to why distributed scheduling will not provide much benefits at both light and heavy utilization levels.
3. Why does type 1 hypervisor not need a host operating system when booting up? Explain in 2-3 sentences.
4. Does a Type 1 hypervisor using paravirtualization need special hardwarde support from the CPU like normal type 1 hypervisors? Why or why not?
5. Does Docker use a form of hardware-level or OS-level virtualization? Explain your answer in 2-3 sentences.
6. Why is OS virtualization more lightweight than hardware virtualiation?
7. What is the primary difference between process and code migration?
8. Why does VM migration not cause active network socket connections of processes to break even though the IP address of the physical machine does not move with the VM?

Part 2

1. Which VirtualBox features did you try?
2. What is the size of the virtual disk on the host? Issue df in the VM and report the size of the file-system that the VM sees.
3. How long (approximately) does it take to pause and unpause a running VM?
4. Which networking option (NAT/Bridged) worked for you?
5. How much RAM and disk space did you allocate to the VM?
6. What configuration options did you examine for the VM?
7. Did you install docker in the VM or try the online tutorial?
8. Which docker commands did you try?
9. What pre-packaged software / database did you try? What advantages did you observe?
10. Compare the time required to start a container vs. starting a VM.
11. Explore file system inside the container and examine if the root(/) is shared between the container and the system.
12. Compare the ps output inside the container vs when run for the entire system.
13. What is the CPU/memory allocated to the container?
14. What project did you pick for the volunteer computing part?
15. What resource usage did you observe when the machine was idle?