Section 3: Shell Scripting

Answers

1. a
2. a,b,c, and d
3. b
4. a
5. c
6. d
7. b
8. b
   A given process has at least these file descriptors, but it can possess more as needed.
9. a
10. b
11. a
12. c
13. b
    You have to leave no spaces when assigning a variable in BASH. For example:
    $user="johndoe" NOT $user = "johndoe"
14. b
    while answer c might be somewhat true if we take into consideration that the interpreter will have to parse the strings inside the double quotes to check for any variables and resolve them, but the speed difference will negligible. After all the system admin is not looking for speed as much as he’s looking for readability and reuse.
15. a
16. b
17. b
    While it’s perfectly legal to direct file contents to the command’s standard input using the < sign, this is redundant because cut accepts a file object as its first argument.
18. d
19. c
20. b
    Lines must be sorted before being passed to the uniq command
21. d
22. a
23. a
24. c
25. b
26. b
27. b
    BASH separates commands by using the newline character. if you want to place more

than one command on the same line, then you have to use the semicolon to separate them

28. b
    You can execute a BASH script by preceding the scrip file with sh or bash. Only if you want the script to be self-executable do you have to start the file with the shebang #!/bin/bash
29. b
    File extensions have no effect on how a file gets executed in UNIX and Linux
30. a
    The echo command can handle those characters but using –e switch
31. b
32. a
33. b
    BASH does not require parentheses after the function name both when defining it and when calling it.
34. c
35. c
36. a
37. b
38. a
39. b
40. d
    $# alones will give you the highest index of the items in the array. Add 1 to it to get the total items count.
41. a
42. b
43. b
44. b
    Python recognizes different command
45. a
46. a
47. a
48. b
    A tuple cannot be changed one created while a list can
49. a
50. a