

Question 1 (10 marks)

Write a complete C program to create a new file named "foo" in the current directory and generate 100 random numbers and save these numbers to file "foo".

Question 2 (10 marks)

Given the following Unix code excerpt:

```
for (int i=0; i<4; ++i)
    fork();
```

explain how many child processes would be created by the above for loop statement. In addition, draw a diagram to illustrate the parent-child relationship among these processes (you may insert a hand drawing diagram into the Word document).

Question 3 (20 marks)

Write a complete Unix program in C to list the name of each file directly under the current directory. In addition to the file name, the program should also display its i-node number and its file type, one file per line. In the end, print the total number of files directly under the current directory.

Question 4 (20 marks)

Write a complete Unix program in C to continuously generate a sequence of random numbers at the rate of one random number per second and save each random number into a file named "bar" as soon as the number is generated. When the program runs on the foreground, it should print out its PID on the console every time the user types Ctrl-C on the keyboard and apart from this PID, the program should continue to generate the random numbers as before. However, the program would print the message "Bye-bye for now!" and then quit if the user types Ctrl-\ on the keyboard.

Question 5 (20 marks)

Explain what the Per Process Table of Open Files of a process is in the classical Unix system. With the help of a C program fragment, show how the standard error of a process may be redirected away from the terminal screen to a log file.

Question 6 (20 marks)

Design a paging scheme for a virtual memory system on a 24-bit computer system. You should provide a sufficient number of details to allow us

- 1) to translate a logical address to a physical address if the page table is given and
- 2) to design a page replacement algorithm (no requirement for you to design a page replacement algorithm though).

To help us understand your paging scheme, please provide us with an example page table and an example showing how to translate a logical address to its physical address using the page table if the page containing the address is in the physical memory. Furthermore, explain how to translate a logical address to its physical address if the page containing the address is not in the physical memory.

END OF QUESTIONS

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs