

NYU, Tandon School of Engineering

Bridge to Computer Science Program

4th Exam

Thursday, 10 September 2020

- You have two hours
- There are 100 points total.
- Note that there are longer problems at the end. Be sure to allow enough time for these.
- We supplied you with a file, named 'solutions.txt', where you should type all your answers.
- For editing this file, you are allowed to use only plain text editors (Notepad for Windows users, or textEdit for Mac users).
- You may use Visual Studio, XCode or CLion for a compiler
- Calculators are not allowed.
- This is a closed-book exam. No additional resources are allowed.
- Pay special attention to the style of your code. Indent your code correctly, choose meaningful names for your variables, define constants where needed, choose most suitable control statements, etc.
- In all questions you may assume that the users enter inputs as they are asked. For example, if the program expects a positive integer, you may assume that users will enter positive integers.
- No need to document your code in this exam, but you may add comments if you think they are needed for clarity.
- Read every question completely before answering it.

- 1) (3 pts) Choosing which page to remove from main memory when a page fault has occurred is part of which concept
 - a. Working set management
 - b. Load control
 - c. Resident set management
 - d. The UNIX fork function
- 2) (3 pts) On a Operating system which doesn't understand threading, what can be used to made use of the concept of "modular program structure"?
 - a. Kernel Level threads
 - b. User Level threads
 - c. A hybrid approach
 - d. Light weight processes
- 3) (3 pts) If a process with PID 14 calls the fork function and new process will be created with PID 27. In process 27, what is the return value of the fork function?
 - a. 14
 - b. 27
 - c. 0
 - d. undefined
- 4) (3 pts) A deadlock prevention algorithm which allows the operating system to forcefully remove locks would be eliminating?
 - a. Hold-and Wait
 - b. Circular Wait
 - c. No Preemption
 - d. Mutual Exclusion
- 5) (3 pts) When using a semaphore which is uninitialized, list the function(s) which would be called to initialize it and list the function(s) which would be called at the beginning and end of a critical section.
- 6) (10 pts) Explain how Peterson's algorithm makes use of the fundamental system to provide for mutual exclusion.
- 7) (10 points) Converting a logical memory address to a physical memory address is a time-consuming process which has to be done VERY frequently. What are some of the decisions that have been made regarding paging and some of the hardware that makes this the conversions more efficient?
- 8) (10 pts) For page replacement algorithms, LRU would be effective, but isn't possible to implement in most environments today. Explain why.

- 9) (15 pts) When using Wireshark, you discover that there are a large number of SYN packets being sent to a machine in which each packet has a destination port one higher than the previous packet (incrementing by one each time). The machine responds with a FIN packet for all ports except the ones it's listening on. However, for the ports that the server is listening on, the client sends a FIN after receiving the reply from the server. Obviously, this results in no connections ever being made, explain what you think is happening in this situation.
- 10) (10 pts) When using TCP to communicate with a machine which is, physically, very distant, we see a lot of packets being sent and then a long delay occurs before more information is sent. Explain what's happening and why.
- 11) (15 pts) A company has setup two networks, one in New York and the other in Westchester. There is a router and WAN link between these two offices. The IP address of a workstation in NY is 10.0.1.26/29 (SM: 255.255.255.248). This workstation is printing to a printer with an IP address of 10.0.1.30. Determine on which of the two networks the printer is located, please show your work.
- 12) (15 pts) A hash table is a data structure which allows for insertion and searching in constant time. It does this by providing a "hash" function generates a unique output for all inputs. The most basic hash function, for integers, would be to simply, mod the number. Obviously, this will result in "collisions" when two input integers hash to the same value. A simple implementation of a hash table would be to store a vector of lists of integers.

Please create a class called HashTable. You should initialize the vector size as 100 and your hash function will be a simple: `hashValue = input%100`. Create functions to insert and search. Resolve collisions by adding the items to the list.