Operating System 2021
Quiz #1

- Concurrency is about dealing with lots of things at once. Parallelism is about doing lots of things at once., Concurrent execution is possible on a single processor., Concurrent execution and parallel execution can be combined.
- A hardware trap caused by dereferencing an invalid pointer is synchronous to a running process., A software interrupt generated by a terminal program on user request is asynchronous to a running process., The kernel handles asynchronous interrupts that may not be visible by any running processes., A kernel has to exploit concurrency in order to be efficient.
- C programmer expected to handle runtime error situations
  - The function void perror(const char *s) defined in stdio.h can be used to write a message s to the standard error stream followed by a message describing the last error encountered during a call to a system or library function., The function strerror() converts an error code into a string describing the error code., Error messages do not belong on the standard output and should be written to the standard error.
- libraries and dynamic linking
  - A dynamically linked executable executes the dynamic linker before the main() function is called., Deploying an updated dynamic library requires the restart of all affected programs but no explicit call of the linker., A dynamically linked program may need more main memory than the corresponding statically linked program., Dynamic libraries may be shared between multiple programs that use the same dynamic libraries.
- operating system architectures
  - A microkernel architecture is essentially a layered architecture with just two layers., A virtual machine kernel architecture benefits from having three or more privilege levels., A modular kernel architecture enables smaller kernels by loading only the modules that match the hardware., A modular architecture simplifies the development of new kernel modules such as driverrs or filesystems.
- system calls and programs report runtime error conditions
  - System calls often return the int status code -1 to indicate that an error occurred., The global symbol errno declared in errno.h resolves to an int value indicating the error code of the last error that has occurred., A program exiting with a non-zero exit code is indicating an error.
- The separation ensures that systems can be used in a wide range of different situations., An operating system should support mechanisms to enforce access control but leave the access control policies to be configured flexibly.
- for standard input, output, error streams
  - The standard input can be redirected to read the content of a file or the output produced by other programs., The C library's stream interface to the standard output is buffered and hence write failures may be detected late (for example, when the stream is flushed or closed)., In a terminal session, the standard output and the standard error are by default both printed to the terminal., In a terminal, the standard input is usually line buffered.
- direct system calls
  - int open(const char *pathname, int flags, mode_t mode);
  - ssize_t write(int fd, const void *buf, size_t count);
  - int pause(void);

- A loaded kernel module executes in system mode., Regular programs execute in user mode., Transition from user mode into system can be caused by an interrupt or a system call.