Multiprogramming is a rudimentary form of parallel processing in which several programs are run at the same time on a uniprocessor. Since there is only one processor, there can be no true simultaneous execution of different programs. Instead, the operating system executes part of one program, then part of another, and so on. To the user it appears that all programs are executing at the same time.

If the machine has the capability of causing an interrupt after a specified time interval, then the operating system will execute each program for a given length of time, regain control, and then execute another program for a given length of time, and so on. In the absence of this mechanism, the operating system has no choice but to begin to execute a program with the expectation, but not the certainty, that the program will eventually return control to the operating system.

If the machine has the capability of protecting memory, then a bug in one program is less likely to interfere with the execution of other programs. In a system without memory protection, one program can change the contents of storage assigned to other programs or even the storage assigned to the operating system. The resulting system crashes are not only disruptive, they may be very difficult to debug since it may not be obvious which of several programs is at fault.

See also: multithreading, multitasking, multiprocessing