Operating systems, security and networks (207SE) Lab 12: Jobs

# Your task

1. W rite around ½ of a side of A4 comparing **multiprogramming and multitasking.** You should define what these **approaches are**, **consider how they have evolved overtime** and **identify their similarities and differences**. You should including **diagrams** to support your findings.

**MULTIPROGRAMMING & MULTITASKING**

There was a time when there was a restriction in technology advances, there were older system, the prices were expensive and the computers were slow and processes needed to use peripheral devices. This meant that the CPU was sitting idle of a long period of time. With time and technological advances, a solution was developed and it is called ‘Batch Processing’ which is a new term used for modern operating systems. Multi-programming is an old term because in modern operating systems the whole program is not loaded completely into the main memory.

In a multiprogramming operating system, there are one or more programs (processes) resident in computer’s main memory ready to execute. Only one program at a time can be executed in the CPU, while the others wait their turn to use the CPU. The whole idea of having a multi-programmed system is to optimize system utilization, more specifically CPU time. For example, the program currently being executed gets interrupted by the operating system between tasks and transfer control to another program in line. Running program keeps executing until it voluntarily gives the CPU back or when it blocks for IO. The design goal is very clear as processes waiting for IO should not block other processes which wastes CPU time. The idea is to keep the CPU busy, if there are processes ready to execute. (shown in diagram1)

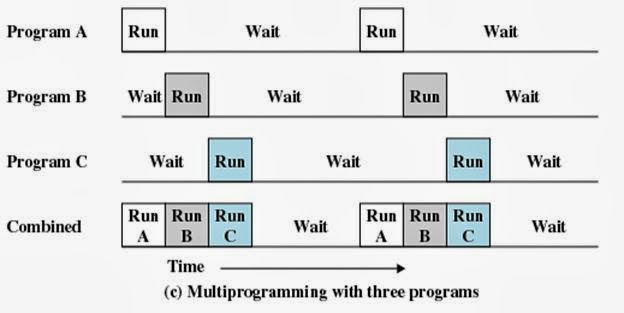


Diagram 1

Multitasking has the same meaning of multiprogramming but in a more general sense, as it refers to having multiple (programs, processes, tasks, threads) running at the same time. This term is used in modern operating systems when multiple tasks share a common processing resource (e.g., CPU and Memory). At any time, the CPU is executing one task only while other tasks waiting their turn. The illusion of parallelism is achieved when the CPU is reassigned to another task (i.e. process or thread context switching).  
There are subtle differences between multitasking and multiprogramming. A task in a multitasking operating system is not a whole application program but it can also refer to a “thread of execution” when one process is divided into sub-tasks. Each smaller task does not hijack the CPU until it finishes like in the older multiprogramming but rather a fair share amount of the CPU time called quantum. Both multiprogramming and multitasking operating systems are (CPU) time sharing systems. However, while in multiprogramming (older OSs) one program keeps running until it blocks, in multitasking (modern OSs) time sharing is best manifested because each running process takes only a fair quantum of the CPU time. (shown in diagram2)

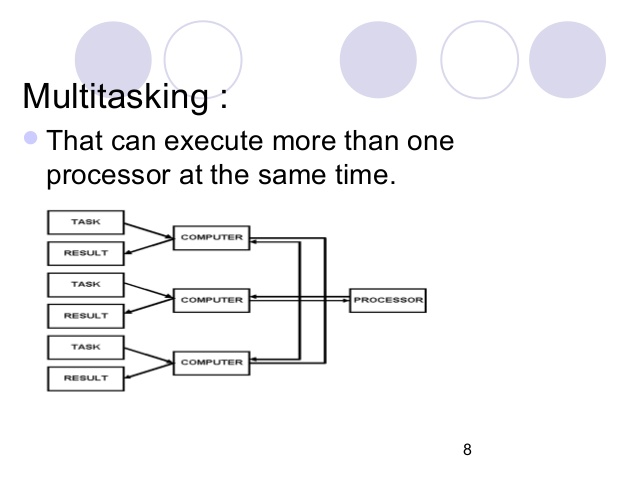


Diagram 2

In conclusion, Multiprogramming operating system allows multiple processes to reside in main memory where only one program is running. The running program keeps executing until it blocks for IO and the next program in line takes the turn for execution. The goal is to optimize CPU utilization by reducing CPU idle time. However, In multitasking time sharing is best manifested because each running process takes only a fair quantum of the CPU time.

1. In no more than a paragraph, describe what is an operating systems scheduler, **why is it needed** and **what are the different approaches to measure how good the scheduler is.**

[Operating system](http://www.wisegeek.org/what-is-an-operating-system.htm) scheduling is the process of controlling and prioritizing messages sent to a processor. An internal operating system program, called the scheduler, performs this task. The goal is maintaining a constant amount of work for the processor, eliminating highs and lows in the workload and making sure each process is completed within a reasonable time frame. Scheduling is typically broken down into three parts: long, mid, and short-term scheduling. Long-term scheduling revolves around admitting programs to the scheduling process. When a new program initiates, the long-term scheduler determines if there is enough space for the new entrant. If there isn’t, then the scheduler delays the activation of the program until there is enough room. The midterm scheduler decides which processes have been idle and which are active. It leaves the active processes alone and writes idle ones to the [hard drive](http://www.wisegeek.com/what-is-a-hard-drive.htm). This frees up memory for other programs to come in through the long-term scheduler. When the mid- and long-term schedulers are combined, instead of delaying activation of a new process, the scheduler simply swaps it into storage. The short-term scheduler is the part that works directly with the processor. This portion activates processes, sets priorities and oversees the processor’s workload. The short-term scheduler is constantly trying to anticipate computer needs to keep the processor running smoothly.

## **References**

N.p., 2017. Web. 10 Apr. 2017.

"Batch Processing". *En.wikipedia.org*. N.p., 2017. Web. 10 Apr. 2017.

"Multiprogramming, Multiprocessing, Multitasking, And Multithreading". *Gabriele Tolomei*. N.p., 2017. Web. 10 Apr. 2017.

twitter., Follow. "Difference Between Multiprogramming, Multitasking, Multithreading And Multiprocessing". *8 BIT AVENUE*. N.p., 2017. Web. 10 Apr. 2017.

"What Is Multiprogramming? - Definition From Whatis.Com". *WhatIs.com*. N.p., 2017. Web. 10 Apr. 2017.

"What Is Operating System Scheduling? (With Pictures)". *wiseGEEK*. N.p., 2017. Web. 10 Apr. 2017.