1 Process

A process is an **running instance** of a program. A process is registered with the OS using its *control structures*: i.e. an entry in the OSs process table to a process control blocks (PCB). *The process control block* contains all information necessary to administer the process and is essential for context switching in multiprogrammed systems.

1.1 Memory image of a process

A process *memory image* contains:

- The program code (could be shared between multiple processes running the same code)

- A data segment, stack and heap

Every process has its own *logical address space*, in which the *stack and heap* are placed at opposite sides to allow them to grow.

1.2 Process control block

A process control block contains three types of attributes:

- Process identification (PID, UID, Parent PID)

- Process control information (process state, scheduling information, etc.)

- Process state information (user registers, program counter, stack pointer, program status word, memory management information, files, etc.)

Process control blocks are kernel data structures, i.e. they are protected and **only accessible in kernel mode**! Allowing user applications to access them directly could compromise their integrity The operating system manages them on the users behalf through system calls (e.g. to set process priority)

1.3 Tables

An operating system maintains information about the status of resources in tables

- Process tables (process control blocks)

- Memory tables (memory allocation, memory protection, virtual memory)

- I/O tables (availability, status, transfer information)

- File tables (location, status)

The process table holds a process control block for each process, allocated upon process creation
Tables are maintained by the kernel and are usually cross referenced

2 Context switching

When a *context switch* takes place, the system saves the state of the old process and loads the state of the new process (creates overhead).

- Saved the process control block is updated

- (Re-)started the process control block read

A trade-off exists between the length of the time-slice and the context switch time:

- Short time slices result in good response times but low effective utilisation

- Long time slices result in poor response times but better effective utilisation

Reference section

logical address space
> The address generated by CPU is called logical (or virtual) address space. It is set of logical addresses that are generated by a program.

stack and heap
> Stack is used for static memory allocation and Heap for dynamic memory allocation, both stored in the computer's RAM . Variables allocated on the stack are stored directly to the memory and access to this memory is very fast, and it's allocation is dealt with when the program is compiled.

context switch
> In computing, a context switch is the process of storing the state of a process or of a thread, so that it can be restored and execution resumed from the same point later. This allows multiple processes to share a single CPU, and is an essential feature of a multitasking operating system.