University of Glasgow | School of Computing Science

Networks & Operating Systems Essentials

Dr Angelos Marnerides

<*angelos.marnerides@Glasgow.ac.uk*>

School of Computing Science

NETWORKS

Key concepts

- General
  - OSI reference architecture:
    - What are the layers
    - What service/input is every layer assuming to get from the layer below
    - What service/output is every layer offering to the layer above

Key concepts

- Physical layer
  - What does it offer to the layers above?
  - Baseband data encoding schemes (NRZ, NRZI, Manchester)
  - Carrier modulation basics (AM, FM, PM, spread spectrum communication)
  - Characteristics governing the maximum transmission rate of a channel
- Data link layer
  - What does it offer to the layers above?
  - Basics of addressing, framing, synchronization, error detection and correction
  - Contention-based MAC (Aloha, CSMA, CSMA-CD)

Key concepts

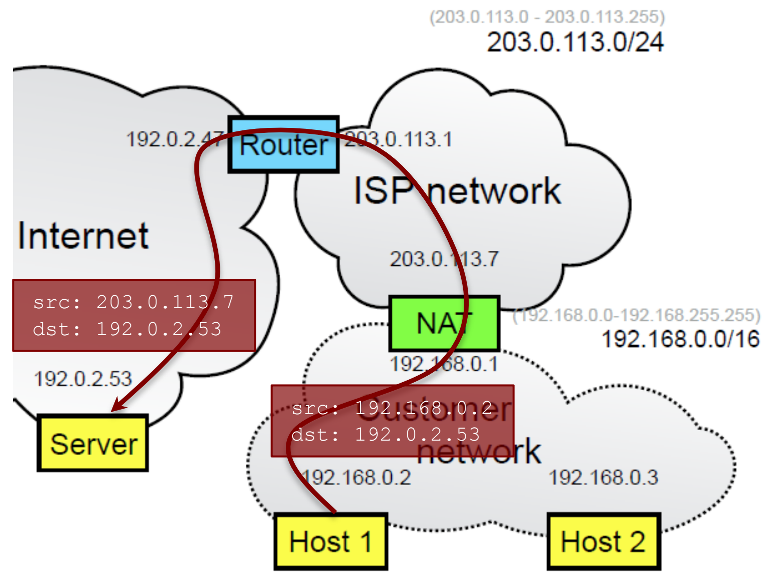
- Network layer
  - What does it offer to the layers above?
  - Best-effort
  - ASs, LANs, routers/gateways, internet
  - IP: service model, addresses (IPv4), fragmentation
  - Differences between IPv4 and IPv6, considerations regarding IPv6 adoption
- Routing
  - Basics of inter-domain routing
  - Distance Vector Routing Protocol
  - Link-State Routing Protocol
  - Comparison between DV and LS

Key concepts

- Transport layer
  - What does it offer to the layers above?
  - End-to-end vs host-to-host: the end-to-end principle
  - Framing, Congestion control
  - UDP:
    - What does it offer on top of IP?
  - TCP:
    - What does it offer on top of IP?
    - How does it compare to UDP?
    - 3-way handshake
    - Sequence numbers, ACKs, detection of lost packets, reordering of packets
    - Congestion control/conservation of packets/AIMD
    - AE1: idiosyncrasies of TCP’s byte stream model and what that means for applications using TCP
  - NAT:
    - How it works? What problem does it solve? What problem(s) does it create?

Key concepts

- What actually happens…
  - Customer acquires a NAT box, which gets the customer's previous IP address
  - Customer gives each host a private address
  - NAT performs address translation
    - Rewrites packet headers to match its external IP address
    - Likely also rewrites the TCP/UDP port number
- The NAT hides a private network behind a single public IP address
  - Private IP network addresses:
    - 10.0.0.0/8 (Class A - Host addresses in the range: 10.0.0.1 - 10.255.255.254)
    - 172.16.0.0/12 (Class B - Host addresses in the range: 172.16.0.1 - 172.31.255.254)
    - 192.168.0.0/16 (Class C - Host addresses in the range: 192.168.0.1 - 192.168.255.254)
- Gives the illusion of more address space



Key concepts

- Higher layers
  - What do they offer to the layer above?
  - What is content negotiation? How is it performed?
  - Text vs binary data transfers
  - Remember what you did for AE1…
  - DNS zones and name resolution; how does it work?
  - How is the internet different to the world wide web?
  - Basics of URIs
  - Basics of HTTP communication

Key concepts

- Client-server application implementation
  - Sockets (TCP vs. UDP)
  - Send() vs sendall()
  - See slides on AE1 : common issues on Moodle.

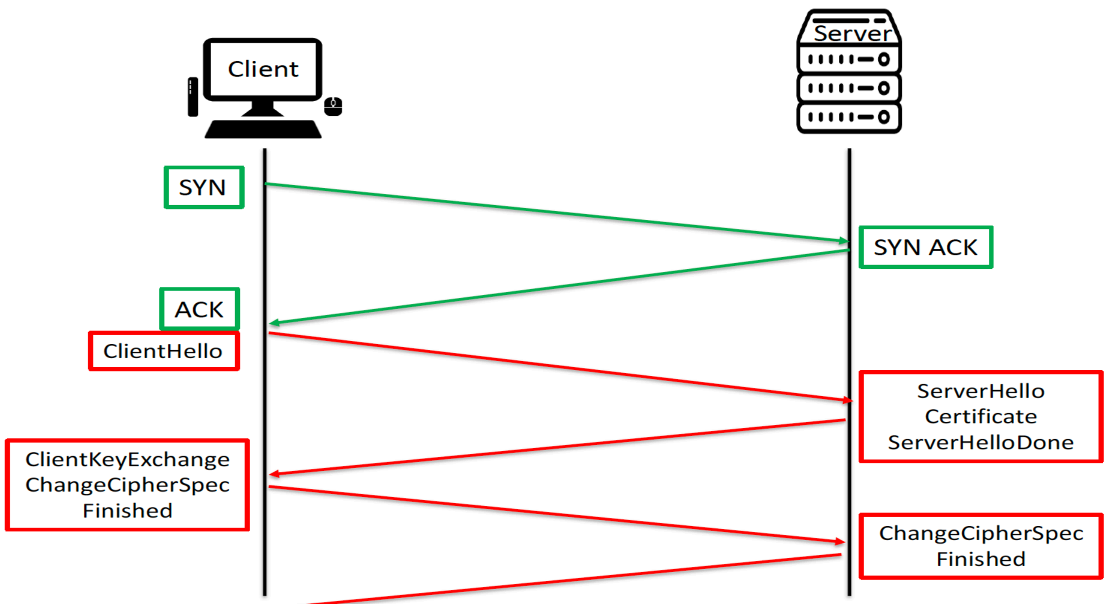
SECURITY & PRIVACY

Key concepts

- CIA Triad
  - Confidentiality: who is allowed to access what
  - Integrity: data to not be tampered by unauthorized party
  - Availability: data protected but available when needed
- Symmetric encryption
  - How does it work?
  - Key distribution problem.
- Asymmetric encryption
  - Public Key Cryptography – how it works?
  - Key splitting
  - Very slow to encrypt/decrypt

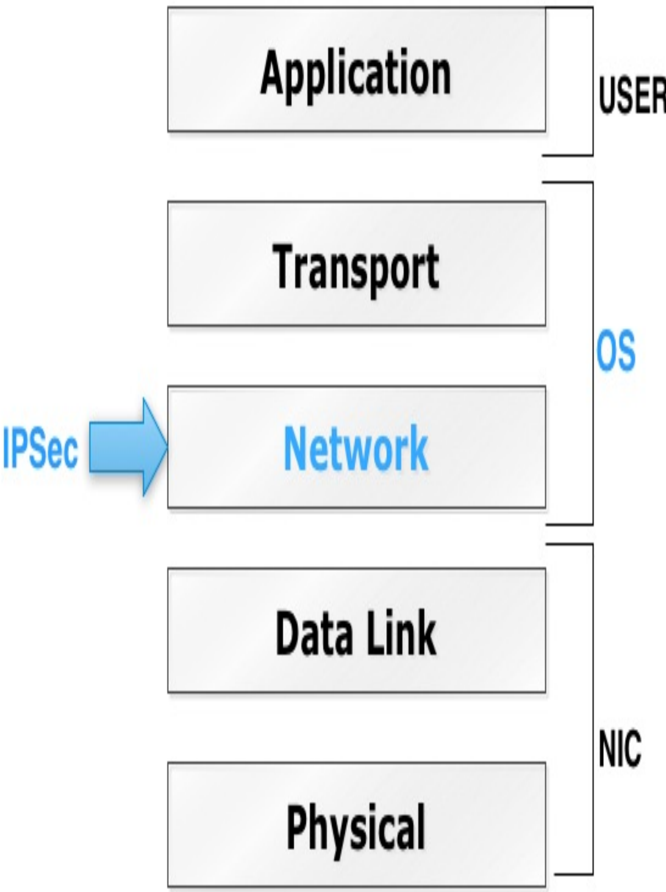
Key concepts

- Transport Layer Security (TLS)
  - Reliable in terms of CIA
- TLS handshake



Key concepts

- IPSec
  - A set of protocols
  - Resides on network layer (not transport at TLS)!
  - Can encrypt from network layer and above



Key concepts

- ISAKMP
  - Part of IPSec
  - Establishing Security Associations (SAs)
  - Procedures for
    - Authentication
    - Creation/management of SAs
    - Key generation/key transport techniques
    - Threat mitigation

Key concepts

- IPSec Modes
  - Gateway-to-gateway
  - End device – to – gateway
  - Gateway – to – server
- IKE, ESP & AH
  - IKE: Internet Key Exchange
    - Provides perfect forward secrecy
  - ESP: payload encryption
  - AH: packet header encryption

Key concepts

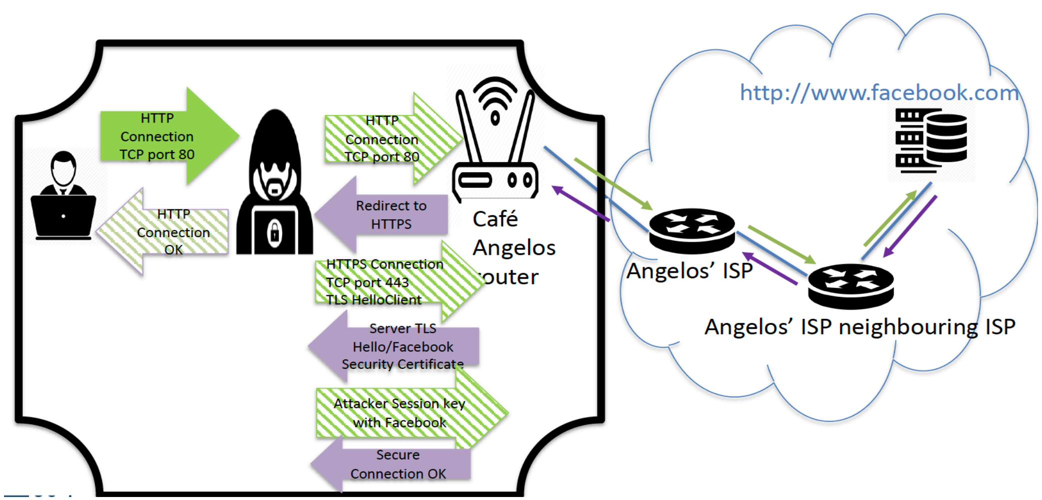
- Privacy & GDPR:
  - What are sensitive personal data?
  - Which are the principles?
    - Lawfulness, fairness & transparency
    - Purpose limitation
    - Data minimization
    - Accuracy
    - Storage limitation
    - Integrity and confidentiality
    - Accountability

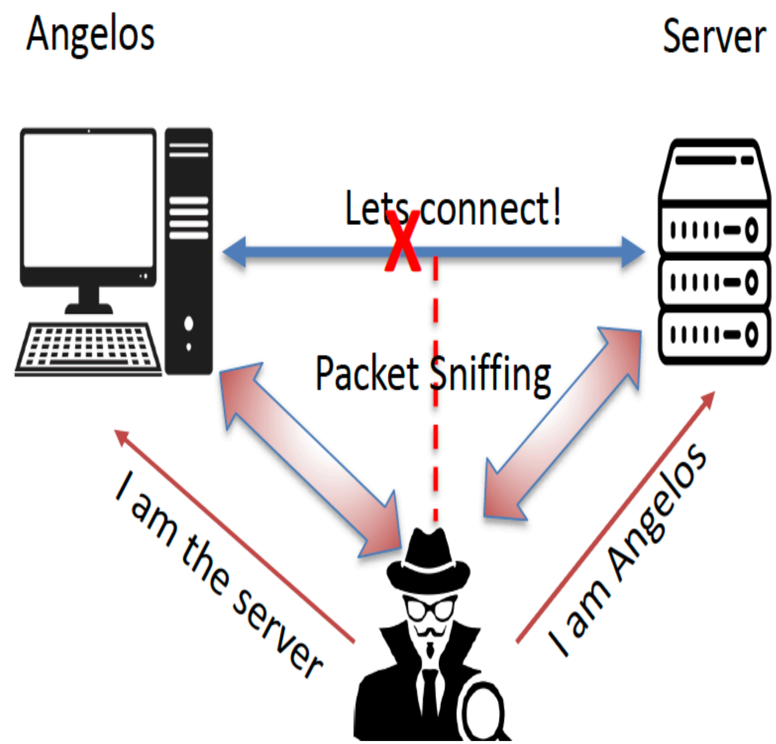
Key concepts

- Privacy criteria:
  - Anonymity
  - Pseudonymity
  - Unlinkability
  - Unobservability

Key concepts

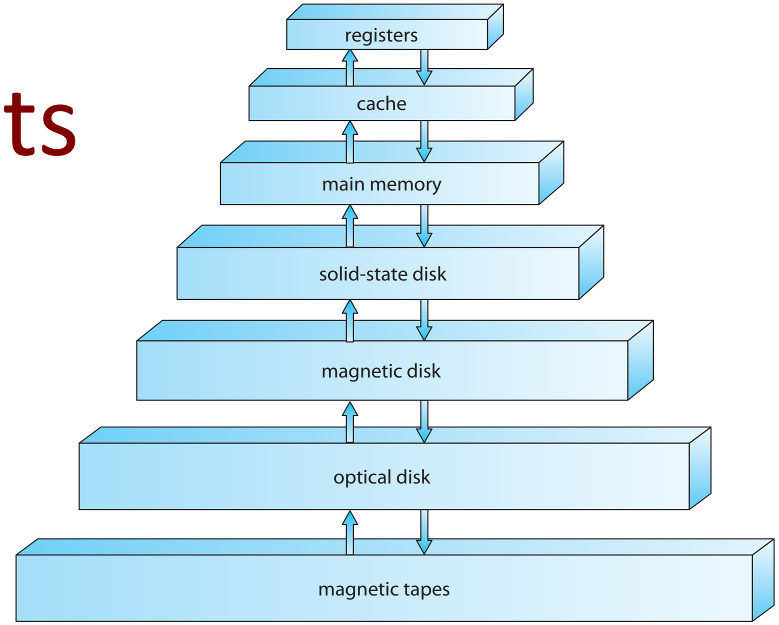
- Threat analysis
  – Man-in-The-Middle (MiTM)
  – Example: SSL Stripping



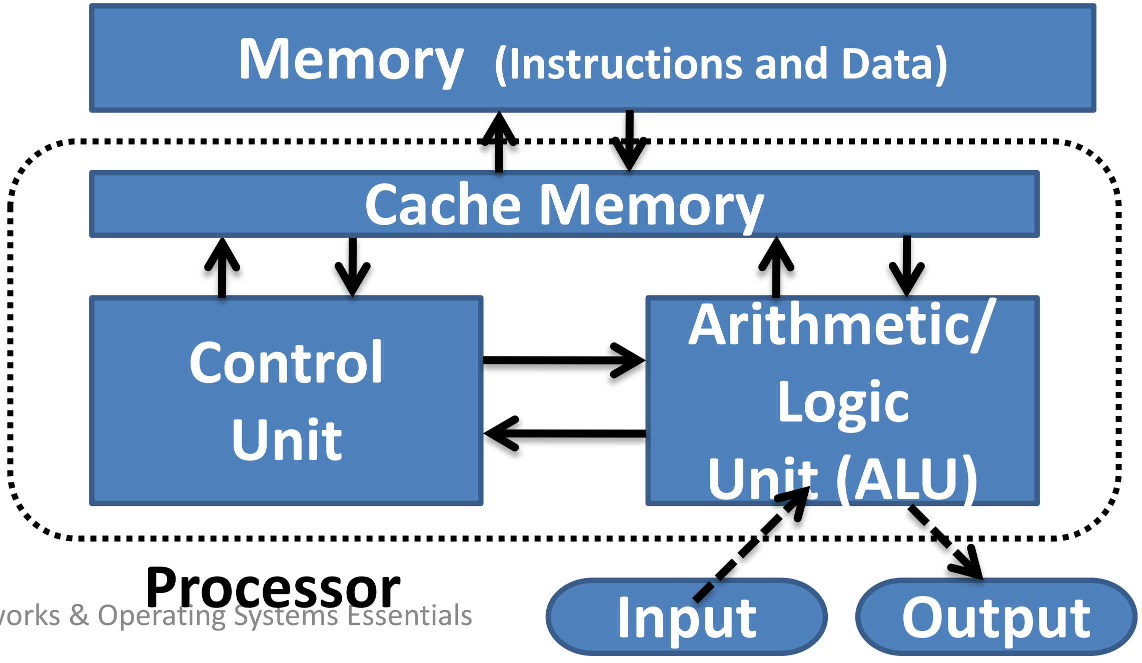


OPERATING SYSTEMS

Key concepts



- Storage/Memory hierarchy
- Von Neumann architecture
- The processor….
- ISA: what is it?



Key concepts

- General
  - Operating system roles (resource allocator, control program)
  - Director Memory Access (DMA)
  - System Calls: What are they? Methods to pass parameters?
- Processes
  - What is a process? What is a thread? How do they differ?
  - Multiprogramming/timesharing. Parallelism vs concurrency.
  - Process Control Block. Process creation. Process states and transitions.
  - IPC:
    - Message Passing (direct/indirect communication, synchronous/asynchronous communication)
    - The Critical Section Problem
    - Starvation, livelock, priority inversion, deadlock
    - Peterson's solution, TSL/CAS, mutexes/semaphores, condition variables (definitions)

Key concepts

- Scheduling
  - CPU/IO bursts. Typical lifecycle of a process.
  - Types of schedulers (long/medium/short-term, (non-)preemptive)
  - Scheduler tasks. Dispatcher tasks.
  - CPU utilisation, throughout, waiting time, turnaround time, response time
  - FCFS, priority-based, SJF, SRTF, RR (with examples)
- Memory management
  - Memory hierarchy (registers → cache → RAM → …)
  - Caching (cache lines, write-through vs write-back, direct-mapped vs fully-associative vs n-way set-associative)
  - Memory address space. Internal fragmentation. External fragmentation.
  - BR/LR. Variable-sized partitions.
  - Paging: frames vs pages, contents and operation of a page table and a TLB, hierarchical/hashed/inverted page tables, demand paging, page fault, thrashing
  - Page replacement algorithms: FIFO, LRU, LFU, aging (with examples); OPT, Random, Clock, NRU (definitions)

Key concepts

- File systems
  - Disk anatomy: seek delay, rotation delay
  - Sectors, fragments, blocks
  - Disk schedulers: FCFS, SSTF, SCAN/LOOK, C-SCAN/C-LOOK
  - Block allocation: contiguous, linked, indexed; comparison among them
  - Disk free space management
  - Filesystem tricks to alleviate disk I/O
  - Logical file system structures (FCB, open file tables, file pointer, file-open count)
  - Directories (what are they? what are their entries? hard vs soft links)

Cache replacement algorithms

- Assume cache with 3 slots
- Consider the access string:

  A, B, C, A, B, B, B, A, C, D, B

- Compute contents of the cache after each access, indicating cache misses, for:
  - LRU
  - LFU
  - Aging with bits shifted after every access

LRU

- Access string: A, B, C, A, B, B, B, A, C, D, B
- Maintaining last access time for every page

A	B	C	A	B	B	B	A	C	D	B
A (0)	A (0)	A (0)	A (3)	A (3)	A (3)	A (3)	A (7)	A (7)	A (7)	B (10)
	B (1)	B (1)	B (1)	B (4)	B (5)	B (6)	B (6)	B (6)	D (9)	D (9)
		C (2)	C (2)	C (2)	C (2)	C (2)	C (2)	C (8)	C (8)	C (8)
*	*	*							*	*

LFU

- Access string: A, B, C, A, B, B, B, A, C, D, B
- Maintaining access counts for every page

A	B	C	A	B	B	B	A	C	D	B
A (1)	A (1)	A (1)	A (2)	A (2)	A (2)	A (2)	A (3)	A (3)	A (3)	A (3)
	B (1)	B (1)	B (1)	B (2)	B (3)	B (4)	B (4)	B (4)	B (4)	B (5)
		C (1)	C (1)	C (1)	C (1)	C (1)	C (1)	C (2)	D (1)	D (1)
*	*	*							*	*

Aging

- Access string: A, B, C, A, B, B, B, A, C, D, B
- Maintaining 3 bits for every page, shifting on every access

A	B	C	A	B	B	B	A	C	D	B
A (100)	A (010)	A (001)	A (100)	A (010)	A (001)	A (000)	A (100)	A (010)	A (001)	B (100)
	B (100)	B (010)	B (001)	B (100)	B (110)	B (111)	B (011)	B (001)	D (100)	D (010)
		C (100)	C (010)	C (001)	C (000)	C (000)	C (000)	C (100)	C (010)	C (001)
*	*	*							*	

Process scheduling

- Consider the below processes
- Show scheduling order and execution times of individual processes (optionally, GANTT)
- Compute turnaround times and average waiting time
- Schedulers:
  - FCFS
  - Non-preemptive priority scheduling
  - SJF
  - RR (quantum: 2)
  - SRTF

Process ID	Arrival Time	CPU Burst Time	Priority
P1	0	4	1
P2	3	4	3
P3	3	2	2
P4	6	7	1
P5	7	2	4

Process scheduling

- How to run by hand
  1. Set time to arrival of first process
  2. While there are more processes to run
     1. Write down the processes in the READY queue (it helps if you use the format PID(remaining time))
     2. Use scheduling algorithm to choose next process to execute
     3. "Execute" process for some time
        - Burst time, for non-preemptive algorithms
        - Quantum, for RR
        - Till arrival of next process, for SRTF
     4. Advance time by the same amount

FCFS

PID	Arrival	Burst
P1	0	4
P2	3	4
P3	3	2
P4	6	7
P5	7	2

- Scheduling order/execution times:
  - Time: 0; Ready queue: P1; next process: P1
    - P1 executes: 0 - 4
  - Time: 4; Ready queue: P2, P3; next process: P2
    - P2 executes: 4 - 8
  - Time: 8; Ready queue: P3, P4, P5; next process: P3
    - P3 executes: 8 - 10
  - Time: 10; Ready queue: P4, P5; next process: P4
    - P4 executes: 10 - 17
  - Time: 17; Ready queue: P5; next process: P5
    - P5 executes: 17 - 19
- Waiting times:
  - P1: 0 (0-0), P2: 1 (4-3), P3: 5 (8-3), P4: 4 (10-6), P5: 10 (17-7)
  - Average: (0 + 1 + 5 + 4 + 10)/5 = 4
- Turnaround times:
  - P1: 4 (4-0), P2: 5 (8-3), P3: 7 (10-3), P4: 11 (17-6), P5: 12 (19-7)

Non-preemptive priority

PID	Arrival	Burst	Priority
P1	0	4	1
P2	3	4	3
P3	3	2	2
P4	6	7	1
P5	7	2	4

- Scheduling order/execution times:
  - Time: 0; Ready queue: P1; next process: P1
    - P1 executes: 0 - 4
  - Time: 4; Ready queue: P2, P3; next process: P3
    - P3 executes: 4 - 6
  - Time: 6; Ready queue: P2, P4; next process: P4
    - P4 executes: 6 - 13
  - Time: 13; Ready queue: P2, P5; next process: P2
    - P2 executes: 13 - 17
  - Time: 17; Ready queue: P5; next process: P5
    - P5 executes: 17 - 19
- Waiting times:
  - P1: 0 (0-0), P2: 10 (13-3), P3: 1 (4-3), P4: 0 (6-6), P5: 10 (17-7)
  - Average: (0 + 3 + 10 + 0 + 10)/5 = 4.6
- Turnaround times:
  - P1: 4 (4-0), P2: 14 (17-3), P3: 3 (6-3), P4: 7 (13-6), P5: 12 (19-7)

SJF

PID	Arrival	Burst
P1	0	4
P2	3	4
P3	3	2
P4	6	7
P5	7	2

- Scheduling order/execution times:
  - Time: 0; Ready queue: P1; next process: P1
    - P1 executes: 0 - 4
  - Time: 4; Ready queue: P2, P3; next process: P3
    - P3 executes: 4 - 6
  - Time: 6; Ready queue: P2, P4; next process: P2
    - P2 executes: 6 - 10
  - Time: 10; Ready queue: P4, P5; next process: P5
    - P5 executes: 10 - 12
  - Time: 12; Ready queue: P4; next process: P4
    - P4 executes: 12 - 19
- Waiting times:
  - P1: 0 (0-0), P2: 3 (6-3), P3: 1 (4-3), P4: 6 (12-6), P5: 3 (10-7)
  - Average: (0 + 3 + 1 + 6 + 3)/5 = 2.6
- Turnaround times:
  - P1: 4 (4-0), P2: 7 (10-3), P3: 3 (6-3), P4: 13 (19-6), P5: 5 (12-7)

RR (quantum: 2)

PID	Arrival	Burst
P1	0	4
P2	3	4
P3	3	2
P4	6	7
P5	7	2

- Scheduling order/execution times:
  - Time: 0; Ready queue: P1 (4); next process: P1
    - P1 executes: 0 - 2
  - Time: 2; Ready queue: P1 (2); next process: P1
    - P1 executes: 2 - 4
  - Time: 4; Ready queue: P2 (4), P3 (2); next process: P2
    - P2 executes: 4 - 6
  - Time: 6; Ready queue: P3 (2), P4 (7), P2 (2); next process: P3
    - P3 executes: 6 - 8
  - Time: 8; Ready queue: P4 (7), P2 (2), P5 (2); next process: P4
    - P4 executes: 8 - 10
  - Time: 10; Ready queue: P2 (2), P5 (2), P4 (5); next process: P5
    - P2 executes: 10 - 12
  - Time: 12; Ready queue: P5(2), P4 (5); next process: P5
    - P5 executes: 12 - 14
  - Time: 14; Ready queue: P4 (5); next process: P4
    - P4 executes: 14 - 16
  - Time: 16; Ready queue: P4 (3); next process: P4
    - P4 executes: 16 - 18
  - Time: 18; Ready queue: P4 (1); next process: P4
    - P4 executes: 18 - 19

- Waiting times:
  - P1: (0-0) + (2-2) = 0
  - P2: (4-3) + (10-6) = 5
  - P3: (6-3) = 3
  - P4: (8-6) + (14-10) + (16-16) + (18-18) = 6
  - P5: (12-7) = 5
  - Average: (0+5+3+6+5)/5 = 3.8
- Turnaround times:
  - P1: 4-0 = 4
  - P2: 12-3 = 9
  - P3: 8-3 = 5
  - P4: 19-6 = 13
  - P5: 14-7 = 5

SRTF

PID	Arrival	Burst
P1	0	4
P2	3	4
P3	3	2
P4	6	7
P5	7	2

- Scheduling order/execution times:
  - Time: 0; Ready queue: P1 (4); next process: P1
    - P1 executes: 0 - 3
  - Time: 3; Ready queue: P1 (1), P2 (4), P3 (2); next process: P1
    - P1 executes: 3 - 4
  - Time: 4; Ready queue: P2 (4), P3 (2); next process: P3
    - P3 executes: 4 - 6
  - Time: 6; Ready queue: P2 (4), P4 (7); next process: P2
    - P2 executes: 6 - 7
  - Time: 7; Ready queue: P2 (3), P4 (7), P5 (2); next process: P5
    - P5 executes: 7 - 9
  - Time: 9; Ready queue: P2 (3), P4 (7); next process: P2
    - P2 executes: 9 - 12
  - Time: 12; Ready queue: P4 (7); next process: P4
    - P4 executes: 12 - 19

- Waiting times:
  - P1: (0-0) + (0-0) = 0
  - P2: (6-3) + (9-7) = 5
  - P3: (4-3) = 1
  - P4: (12-6) = 6
  - P5: (7-7) = 0
  - Average: (0+5+1+6+0)/5 = 2.4
- Turnaround times:
  - P1: 4-0 = 4
  - P2: 12-3 = 9
  - P3: 6-3 = 3
  - P4: 19-6 = 13
  - P5: 9-7 = 2