1.

stack	high
↓	
↑	
heap	
data	
text	low

stack → temporary data storage when invoking function (such as function parameters, return address/ and local variables)

heap ⇒ Memory that is dynamically allocated during program run time.

data → Global value

text → executable code.

2. (i) Switching the CPU core to another process require performing a state save of the current process and a state restore of a different process.

(ii) When system call or interrupt occur.

3. Process Control Block (PCB)
 ① Process state
 ② Program counter
 ③ CPU register
 ④ CPU - scheduling information
 ⑤ Memory - management information
 ⑥ Accounting information
 ⑦ I/O status information

4. (i) Zombie

Definition : a process that has terminated, but whose parent has not yet to called wait().

Handled : Only need to wait a moment. The process identifier of the zombie process and its entry in the process table are released.

(ii) Orphan

Definition : a parent didn't invoke wait() and instead terminated.

Handled : Assign a init process to orphan to be its new parent ⇒ only in UNIX
Assign a systemd or other process in Linux systems.

5. Direct

→ process must name each other explicitly

→ Links are established automatically between every pair of process that want to communication

→ A link is associated with exactly two processes

→ Between each pair of processes, there exists exactly one link.

→ Symmetry in addressing – sender process and receiver process must name the other.

Indirect

→ Send to and received from mailbox or port

→ Mailbox has a unique identification.

→ A process can communicate only if they share a mailbox

→ A link may be associated with more than two processes

→ A number of different links may exist, with each link corrsponding to one mailbox

→ Link may be unidirectional or bi-directional.

6. Socket

→ an endpoint for communication

→ communication of IP address and port

→ Server wait clients by listing to a specificed port

→ All ports below 1024 are well known, used for standard services for server.

→ clients are arbitrary port number greater than 1024

→ Common and efficient

→ low level communication

RPC

→ Abstracts procedure calls between processes on networked systems

→ high level communication

case 1:

當 client 傳送 request 給 server 端，但沒有傳送成功，
則 client 一直傳送 request 直到傳送成功，且回傳給 client.

case 2:

當 client 傳送 request 給 server 端是成功的，但 server 端回傳 ACK 給 client 失敗
則 client 繼續傳 request 給 server 端，只是此時 server 端的 time stamp 有記錄了，
所以就直接回傳相同的 ACK 直到成功。

8. OS typically provides a redezvous (or matchmaker) service to connect client and server.