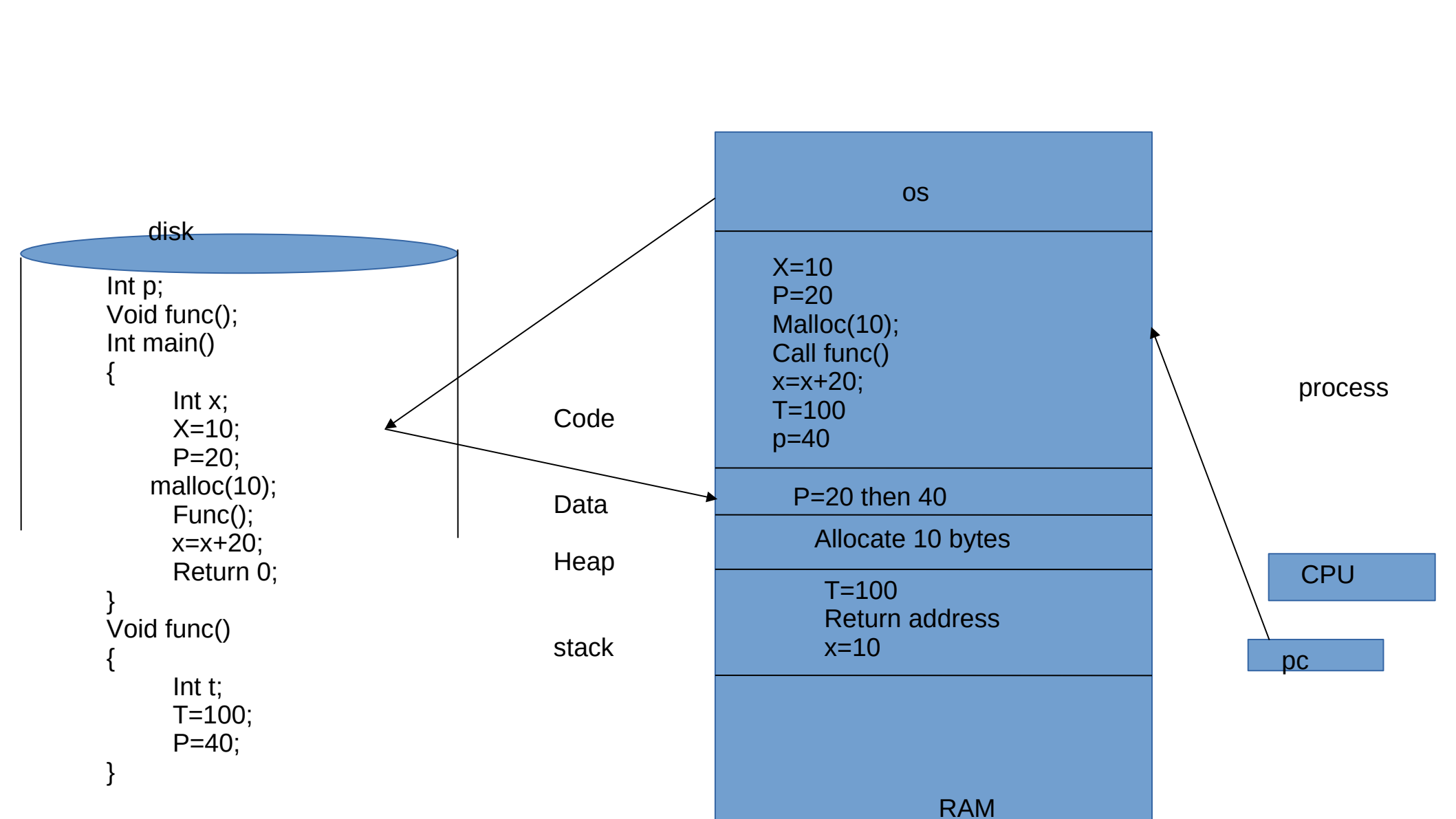

disk
Int p;
Void func();
Int main()
{
Int x;
X=10;
P=20;
malloc(10);
Func();
x=x+20;
Return 0;
}
Void func()
{
Int t;
T=100;
P=40;
}
Code
Data
Heap
stack
os
X=10
P=20
Malloc(10);
Call func()
x=x+20;
T=100
p=40
P=20 then 40
Allocate 10 bytes
T=100
Return address
x=10
RAM
process
CPU
pc

- Os processes execute system code and user processes execute user code.
- The term job and process are used interchangeably.
- Two processes may be associated with the same program. But they are two separate execution sequence.

PROCESS STATE

- New:

A process is being created.

- Running:

Instructions are being executed.

- Waiting:

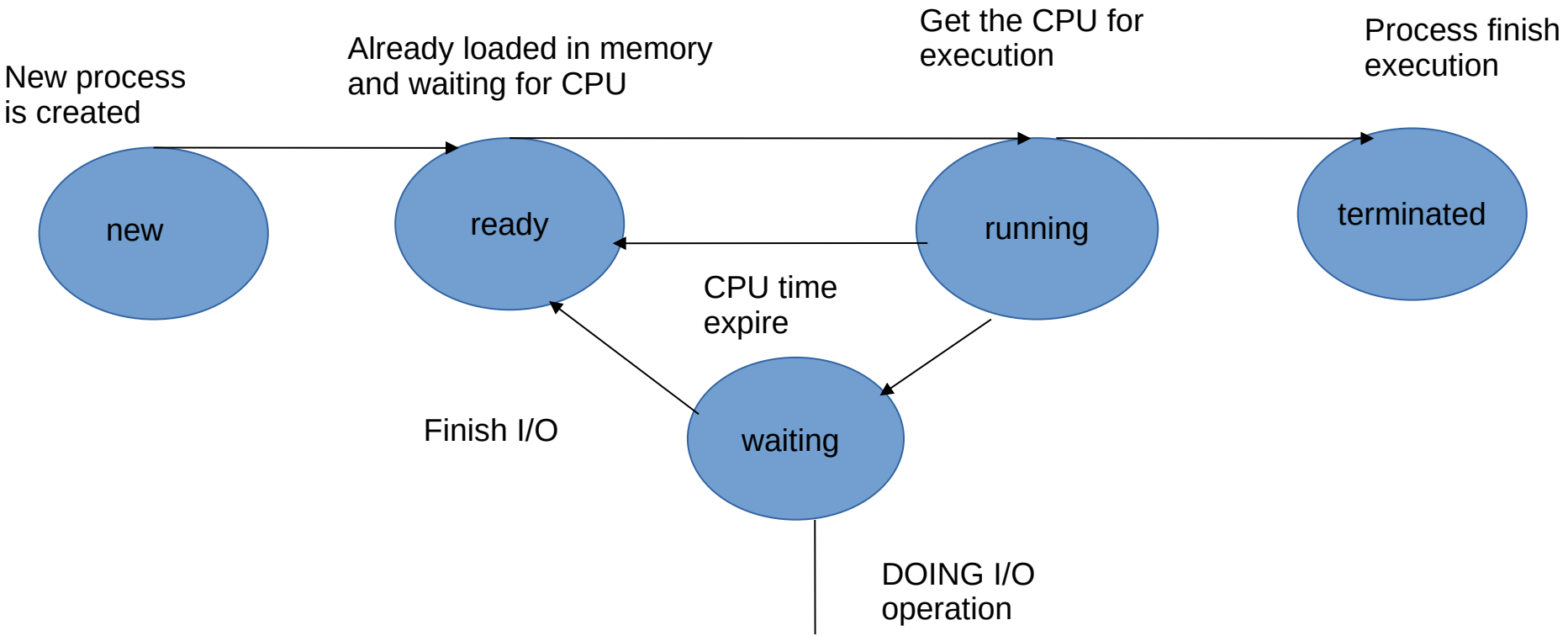
The process is waiting for some event to occur (such as an I/O completion).

- Ready:

The process is waiting to be assigned to a processor .

- Terminated:

The process has finished execution.


New process
is created
Already loaded in memory
and waiting for CPU
Get the CPU for
execution
Process finish
execution
new
ready
running
terminated
CPU time
expire
Finish I/O
waiting
DOING I/O
operation

Process Control Block

Each process is represented in the OS by a process control block(PCB).

It contains many peace of information:

- Process state: new/ready/running/.....
- Program counter information: Address of the next instruction to be execute.
- CPU registers: Registers may be very in numbers.
- CPU scheduling information: Process priority.
- Memory management information: This may include the value of the base and limit address and other informations: page table, segment table etc.
- Account information: This includes the amount of CPU and real time used, process number etc..
- I/O status information: This includes the list of I/O devices allocated to this process.