

PRAKTIKUM SISTEM OPERASI
MODUL 11: PENJADWALAN PROSES DAN
MANAJEMEN MEMORI (OSSim)



Penyusun:

Nama : Della Fitria Lestari
NIM : L200219268
Kelas : E
Mata Kuliah : Praktikum Sistem Operasi
Program Studi : Informatika

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA 2022

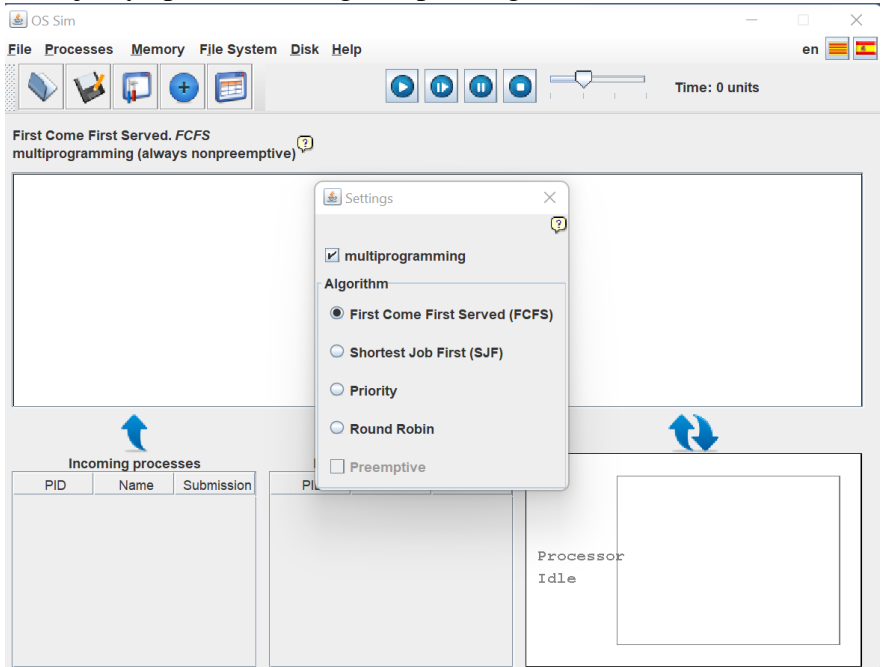
Kegiatan 1. Penjadwalan Proses

1.1 First-Come, First-Served (FCFS)

- a. Membuka program OSSim, selanjutnya pilih menu processes -> process scheduling

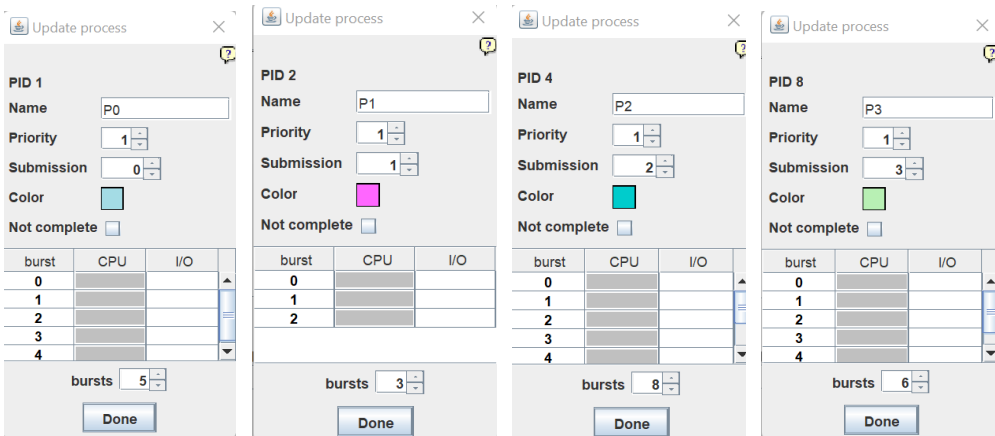


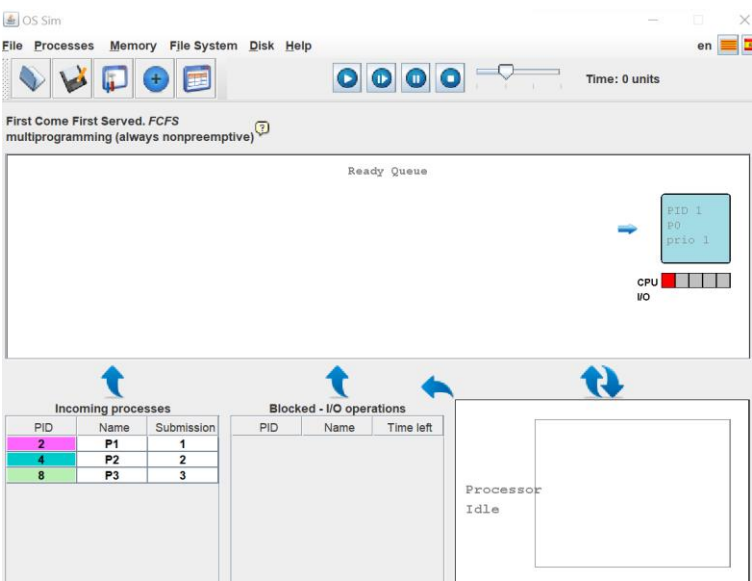
- b. Selanjutnya pilihlah setting dan pilih algoritma First-Come, First-Served (FCFS)



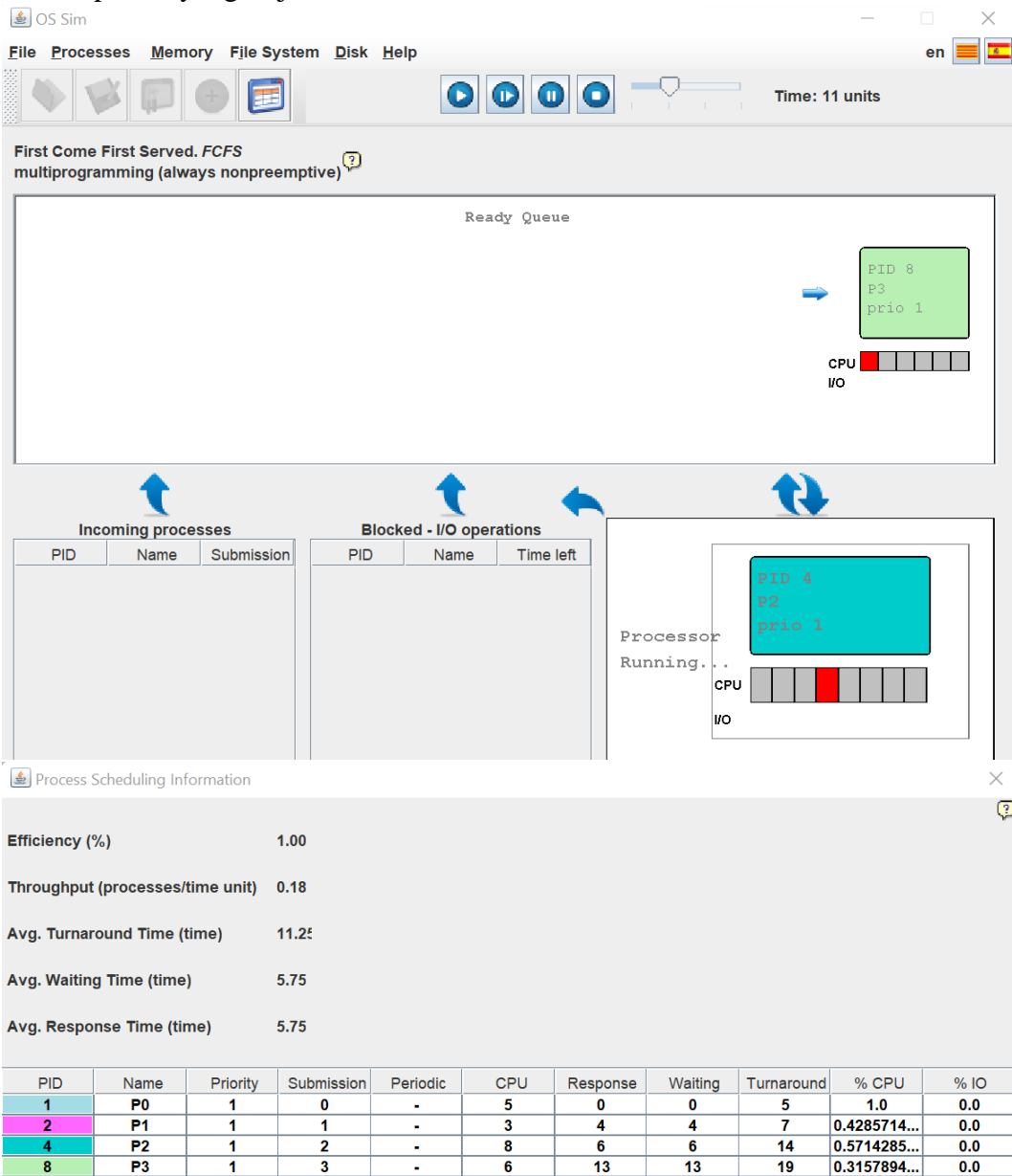
- c. Melakukan input proses sesuai dengan tabel berikut dengan memulai dengan P0 sebagai input proses yang pertama

Proses	Arrival Time	Burst Time	Service Time
P0	0	5	0
P1	1	3	5
P2	2	8	8
P3	3	6	16





d. Jika input sudah selesai dilakukan. Pilih tombol start pada bagian atas. Amati dan analisa proses yang terjadi.



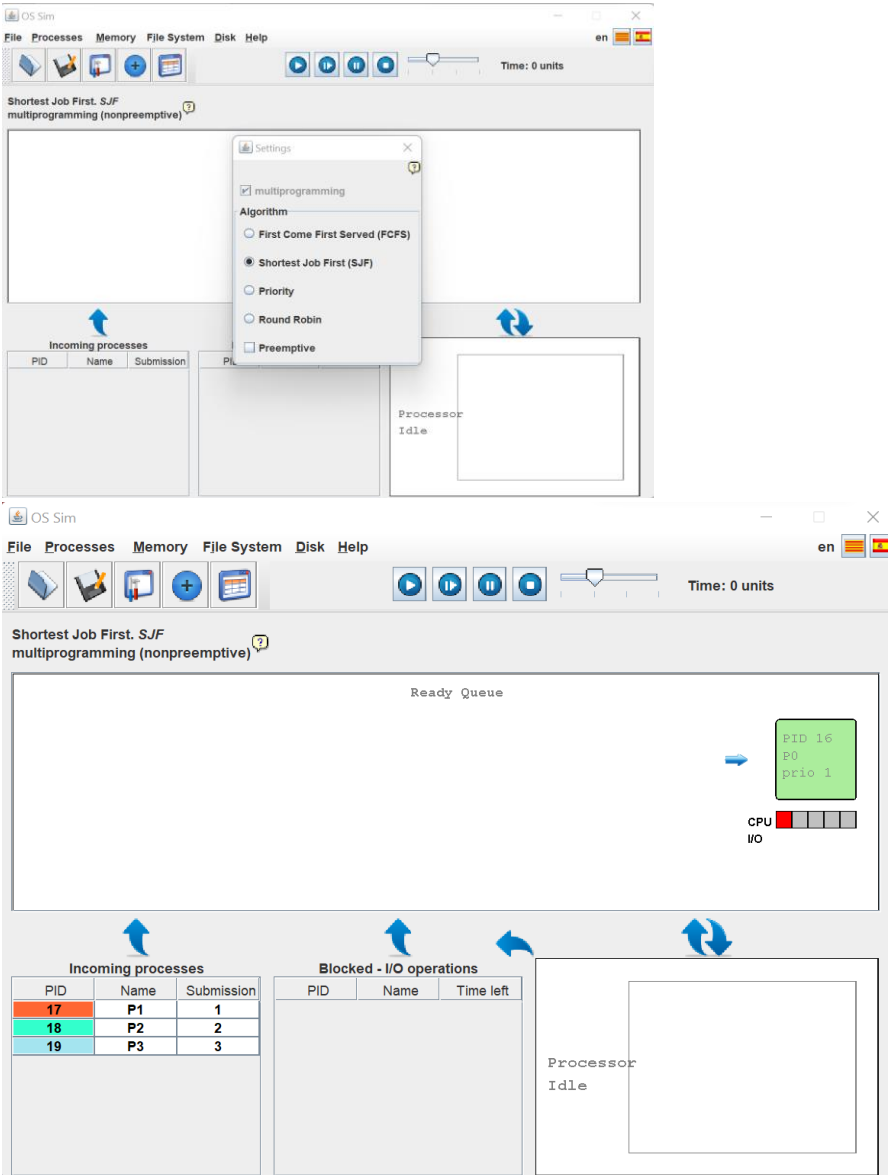
e. Mengisi tabel

Process	Wait time : Service Time – Arrival Time
P0	0-0 = 0
P1	5-1 = 4
P2	8-2 = 6
P3	16-3 = 13
Av wait time	23 / 4 = 5.75

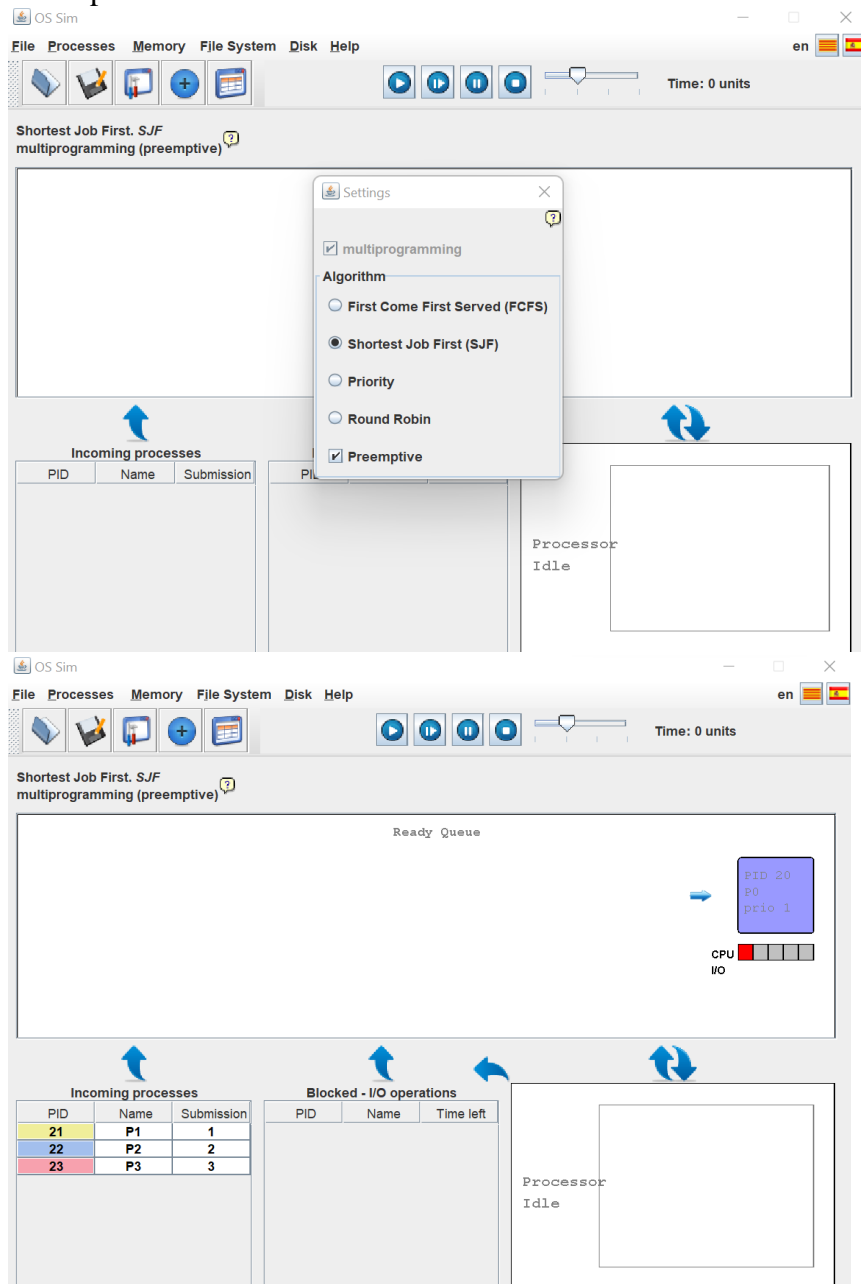
Kesimpulan : setiap proses yang berada pada status ready dimasukkan kedalam FIFO queue atau antrian dengan prinsip first in first out, sesuai dengan waktu kedatangannya. Proses yang tiba terlebih dahulu yang akan dieksekusi.

1.2 Shortest Job First (SJF)

- a. Bukalah program OS Sim, selanjutnya pilih menu processes -> process scheduling
 - b. Selanjutnya pilihlah setting dan pilih algoritma Shortest Job First (SJF). algoritma ini terdiri dari 2 jenis yaitu non-premptive dan preemptive. Untuk mengaktifkan preemptive dengan mencentang menu tersebut. Sebaliknya jika menonaktifkan maka hanya cukup menghilangkan centanganya saja
- 1) Non-Preemptive

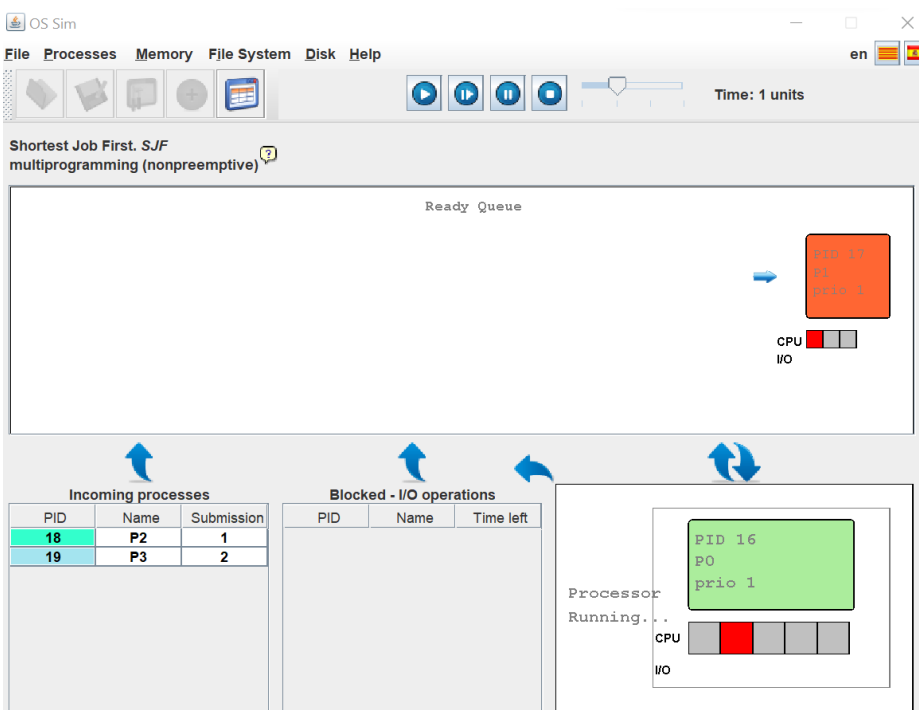


2) Preemperive



c. Selanjutnya klik tombol start. Amati dan analisa proses yang terjadi. Lakukan perbandingan dari hasil keduanya

1) Non-Preemitive



Process Scheduling Information

Efficiency (%)1.00

Throughput (processes/time unit)0.18

Avg. Turnaround Time (time)10.75

Avg. Waiting Time (time)5.25

Avg. Response Time (time)5.25

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
16	P0	1	0	-	5	0	0	5	1.0	0.0
17	P1	1	1	-	3	4	4	7	0.4285714...	0.0
19	P3	1	3	-	6	5	5	11	0.5454545...	0.0
18	P2	1	2	-	8	12	12	20	0.4	0.0

2) Preemtive

OS Sim

FileProcessesMemoryFile SystemDiskHelp

Time: 1 units

Shortest Job First. SJF
multiprogramming (preemptive)

Ready Queue

PID 20
P0
prio 1

CPU I/O

Incoming processes

PID	Name	Submission
22	P2	1
23	P3	2

Blocked - I/O operations

PID	Name	Time left
-----	------	-----------

Processor Running...

PID 21
P1
prio 1

CPU I/O

Process Scheduling Information

Efficiency (%)1.00

Throughput (processes/time unit)0.18

Avg. Turnaround Time (time)10.50

Avg. Waiting Time (time)5.00

Avg. Response Time (time)4.25

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
21	P1	1	1	-	3	0	0	3	1.0	0.0
20	P0	1	0	-	5	0	3	8	0.625	0.0
23	P3	1	3	-	6	5	5	11	0.5454545...	0.0
22	P2	1	2	-	8	12	12	20	0.4	0.0

d. Mengisi tabel

Non-Preemtive

Process	Wait time : Service Time – Arrival Time
P0	0
P1	4
P2	12
P3	5
Av wait time	21 / 4 = 5.25

Kesimpulan : Tidak dapat di interupsi, dimana setiap proses yang ada di ready queue akan dieksekusi berdasarkan burst time. Pemilihan proses adalah proses dalam antrian yang memiliki waktu eksekusi tercepat. CPU tidak memperbolehkan proses yang ada di ready queue untuk menggeser proses yang sedang dieksekusi oleh CPU meskipun proses yang baru tersebut mempunyai burst time yang lebih kecil.

Preemptive

Process	Wait time : Service Time – Arrival Time
P0	3
P1	0
P2	12
P3	5
Av wait time	20 / 4 = 5

Kesimpulan : Dapat di interupsi, proses yang dipilih adalah proses yang memiliki waktu sisa eksekusi terkecil. Teknik ini juga dikenal dengan nama Shortest Remaining Time First. Setiap proses yang ada di ready queue akan dieksekusi berdasarkan burst time.

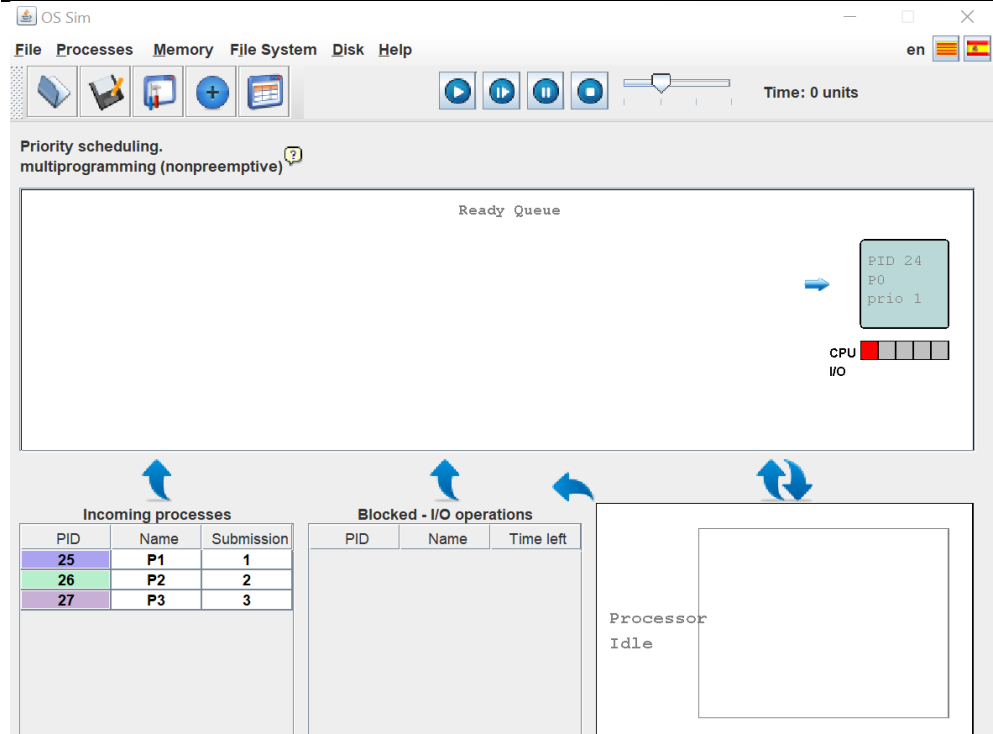
Perbedaan SJF non-preemptive dengan preemptive :

- SJF preemptive yang dieksekusi pertama kali adalah proses yang memiliki burst time paling pendek, sedangkan SJF non-preemptive yang dieksekusi pertama adalah proses yang pertama kali datang kemudian setelahnya yang di eksekusi adalah yang memiliki burst time paling pendek dalam atrian.
- Dalam penjadwalan SJF non-preemptive, proses menahannya hingga mencapai status menunggu atau dihentikan Sedangkan dalam penjadwalan SJF preemptive, proses dimasukkan dalam antrian setiap saat.

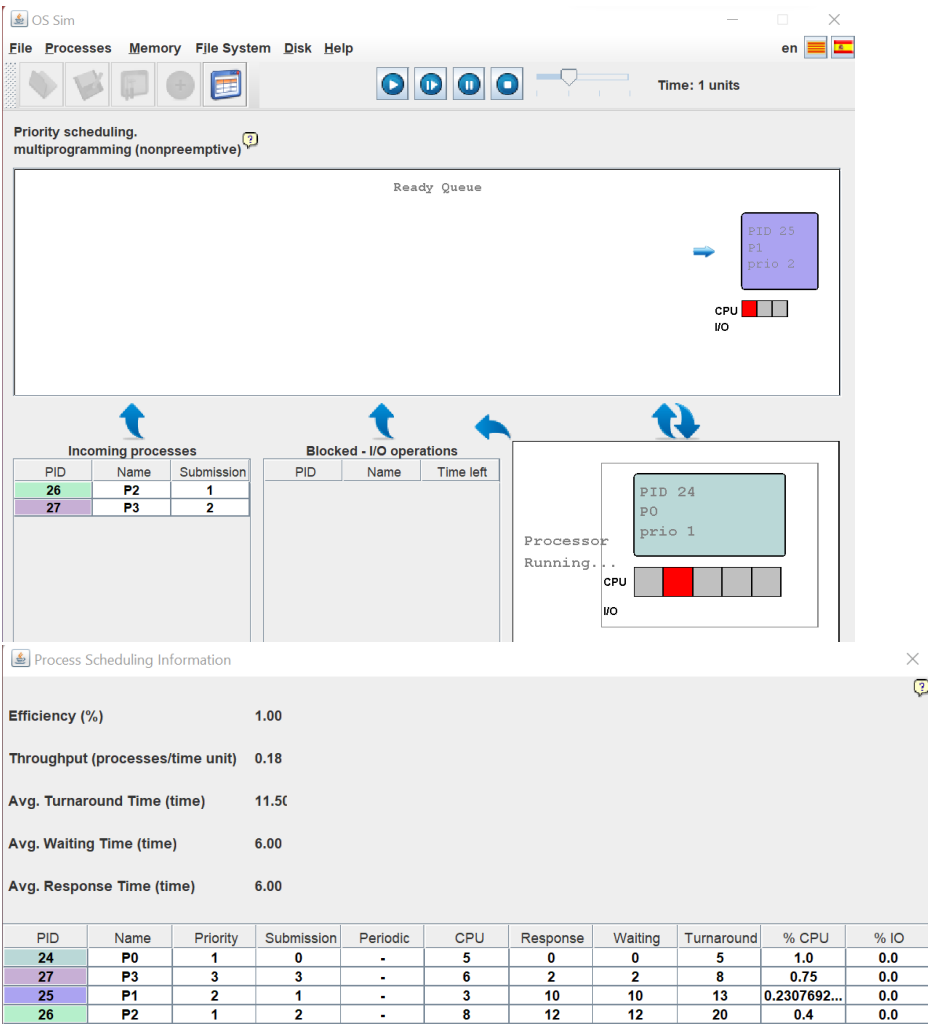
1.3 Priority

- a. Pilihlah menu setting dan pilih algoritma Priority. Selanjutnya tambahkan priority pada setiap proses

Proses	Arrival Time	Burst Time	Priority	Service Time
P0	0	5	1	0
P1	1	3	2	11
P2	2	8	1	14
P3	3	6	3	5



- b. Selanjutnya klik tombol start. Lakukan pengamatan dan analisa proses yang terjadi. Lengkapilah tabel berikut!

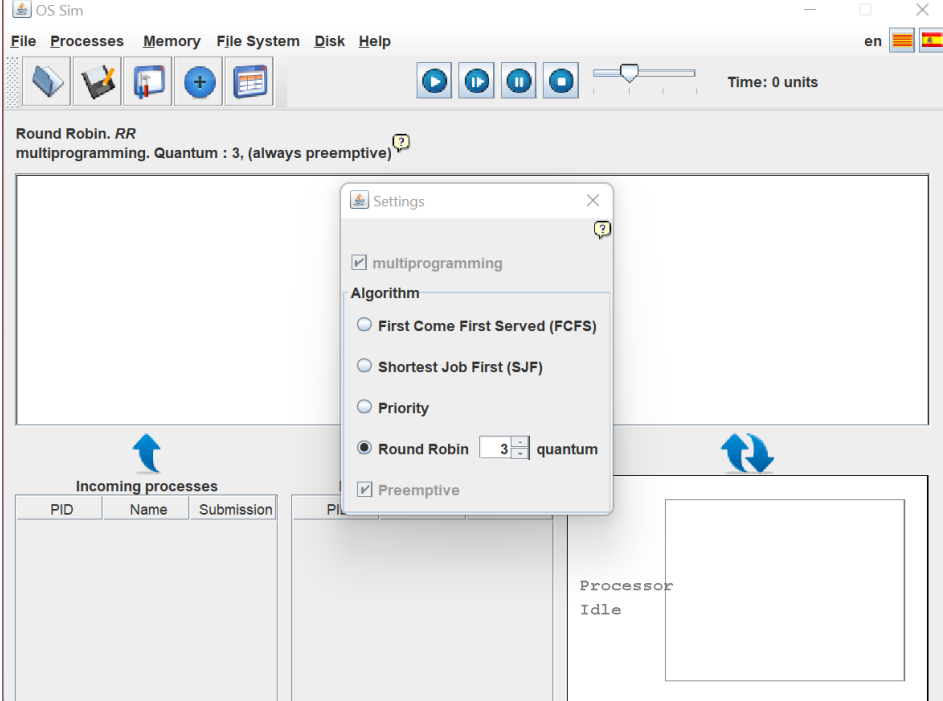


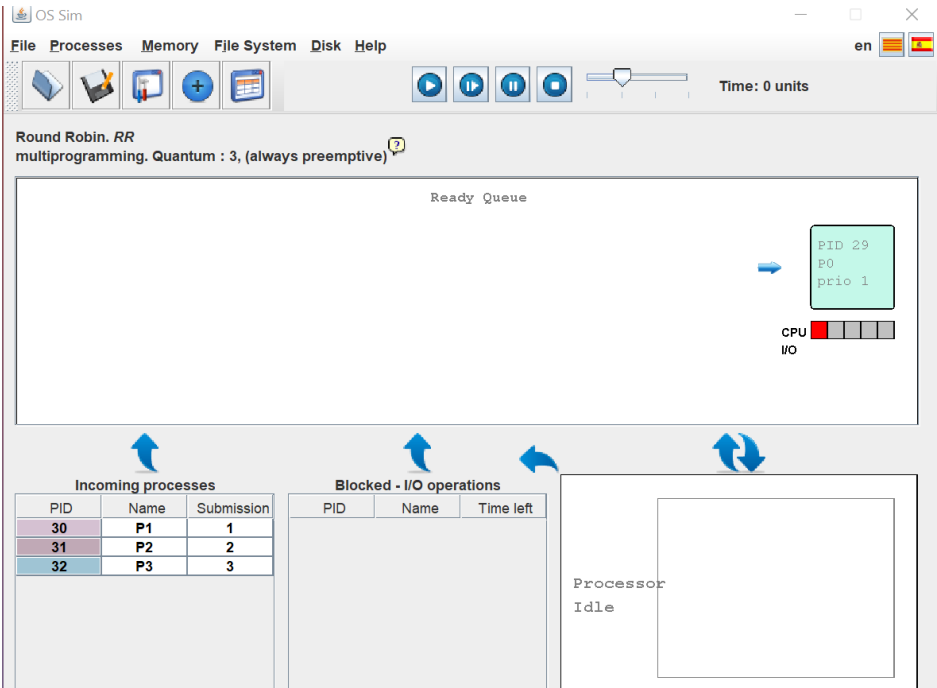
Process	Wait time : Service Time – Arrival Time
P0	0-0 = 0
P1	11-1 = 10
P2	14-2 = 12
P3	5-3 = 2
Av wait time	24 / 4 = 6

Kesimpulan : Penjadwalan yang mendahulukan proses yang memiliki prioritas tertinggi. Setiap proses memiliki prioritasnya masing-masing.

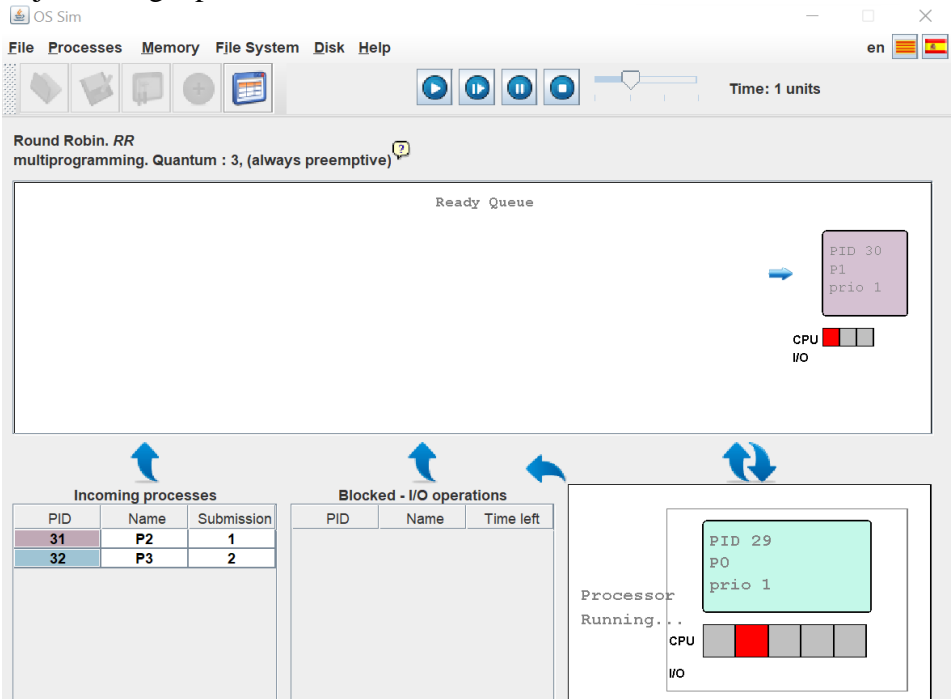
1.4 Round Robin

- a. Pilihlah menu setting dan pilih algoritma Round Robin. Selanjutnya tambahkan quantum time sebesar 3





b. Selanjutnya klik tombol start. Lakukan pengamatan dan analisa proses yang terjadi. Lengkapilah tabel berikut



Process Scheduling Information											
Efficiency (%)		1.00									
Throughput (processes/time unit)		0.18									
Avg. Turnaround Time (time)		14.00									
Avg. Waiting Time (time)		8.50									
Avg. Response Time (time)		3.00									
PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO	
30	P1	1	1	-	3	2	2	5	0.6	0.0	
29	P0	1	0	-	5	0	9	14	0.3571428...	0.0	
32	P3	1	3	-	6	6	11	17	0.3529411...	0.0	
31	P2	1	2	-	8	4	12	20	0.4	0.0	

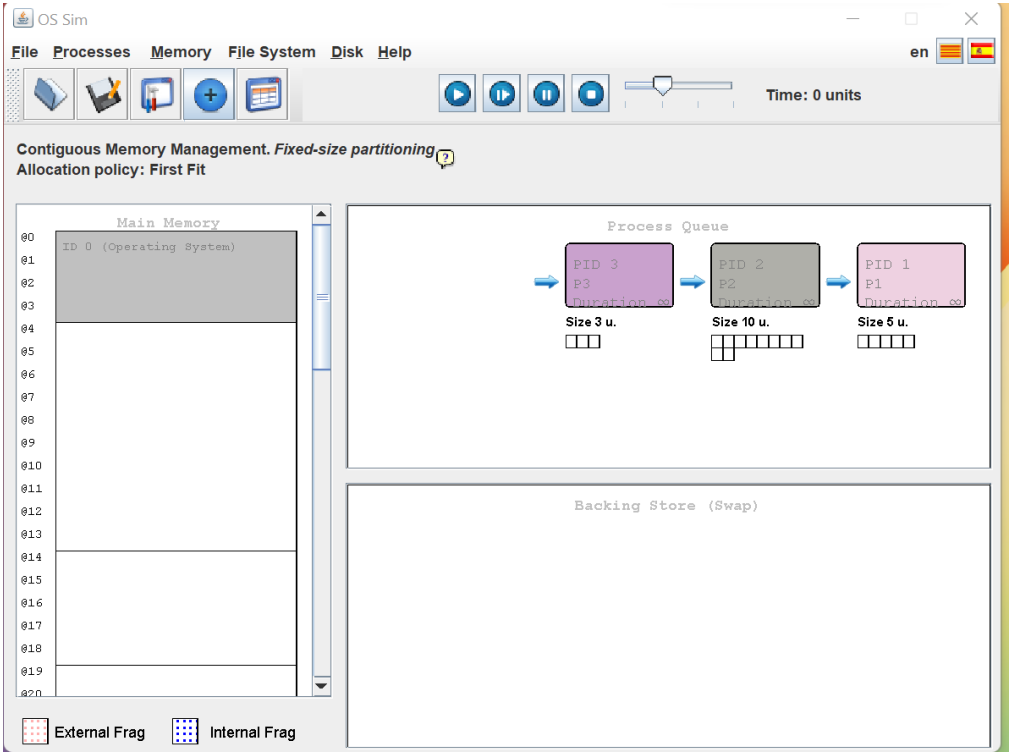
Process	Wait time : Service Time – Arrival Time
P0	9
P1	2
P2	12
P3	11
Av wait time	34 / 4 = 8.5

Kesimpulan : Proses akan mendapat bagian sebesar time quantum. Jika time quantum-nya habis atau proses sudah selesai, CPU akan dialokasikan ke proses berikutnya.

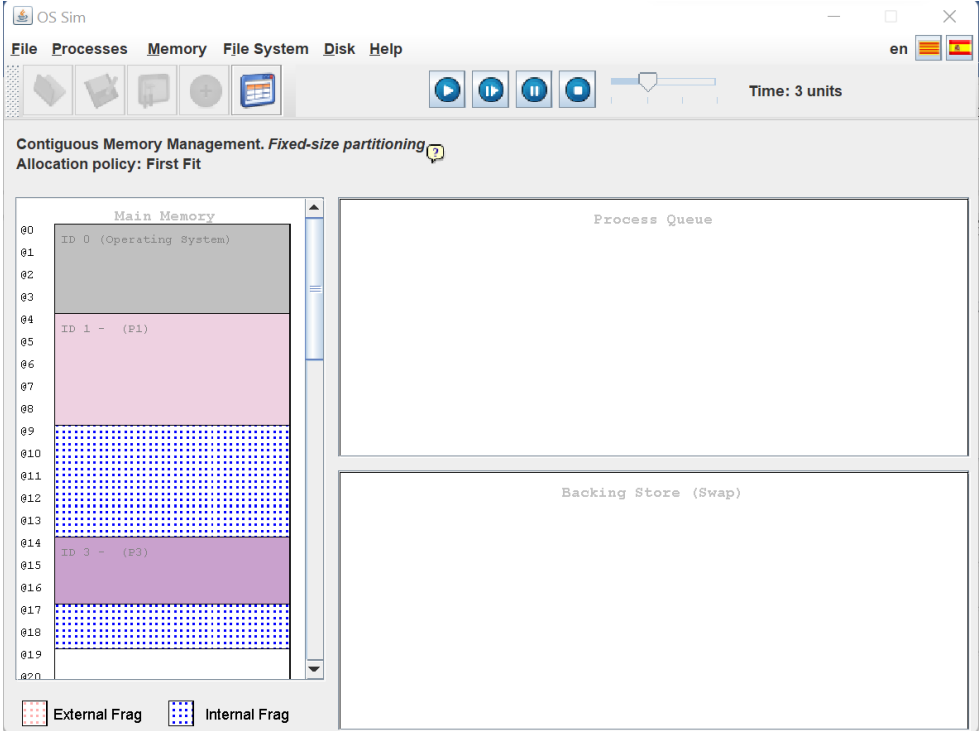
Kegiatan 2: Manajemen Memori

2.1 Contiguous memory management dengan menggunakan partisi berukuran tetap (fixed-size partition) dan aturan first fit

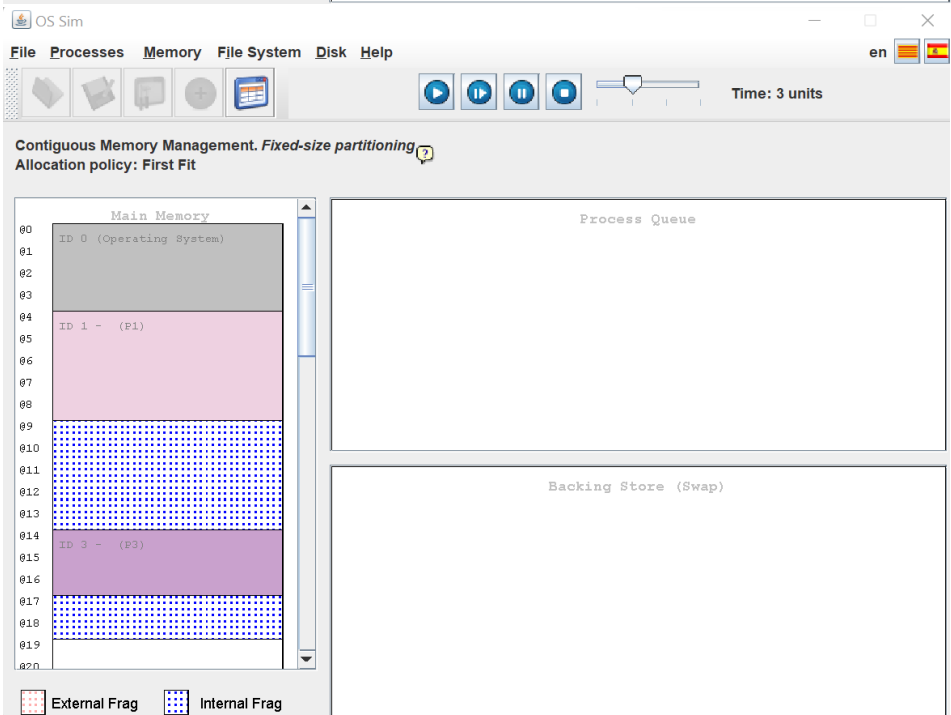
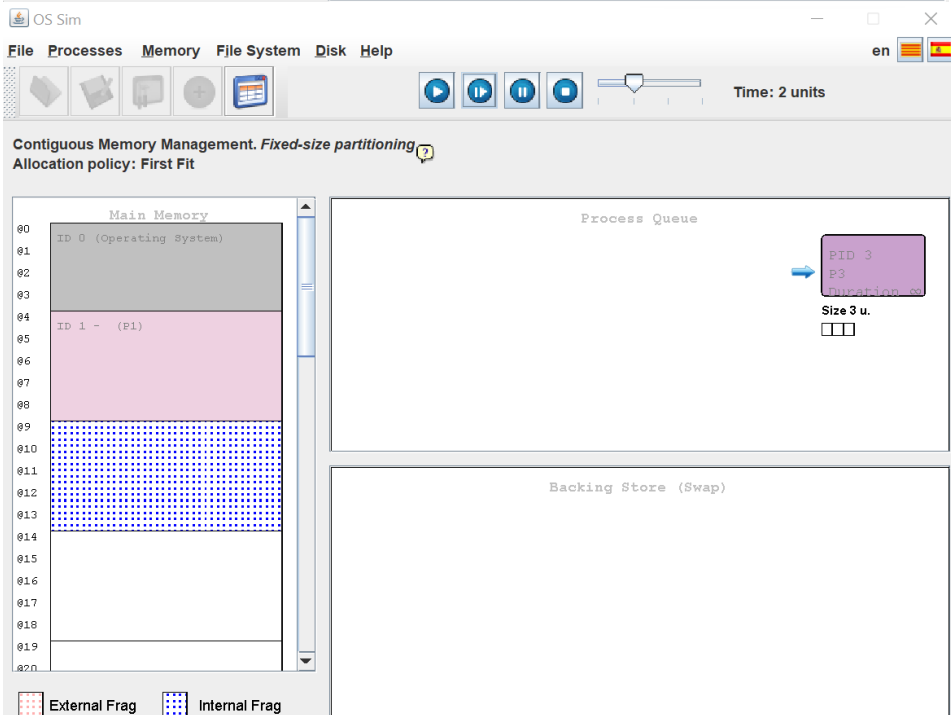
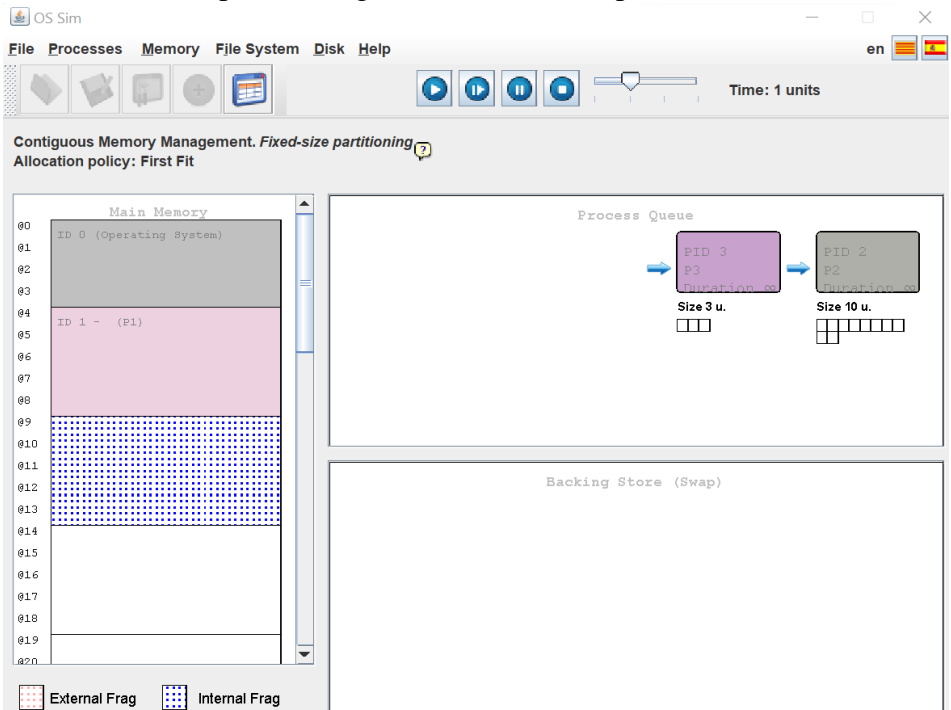
- 1. Pilih menu help >> examples...>> memory management >> Contiguous memory management with fixed-size partitions (first adjustment). Pilihlah folder hingga menampilkan tampilan berikut.



- 2. Selanjutnya klik tombol play untuk melihat proses yang terjadi dan klik tombol stop untuk berhenti

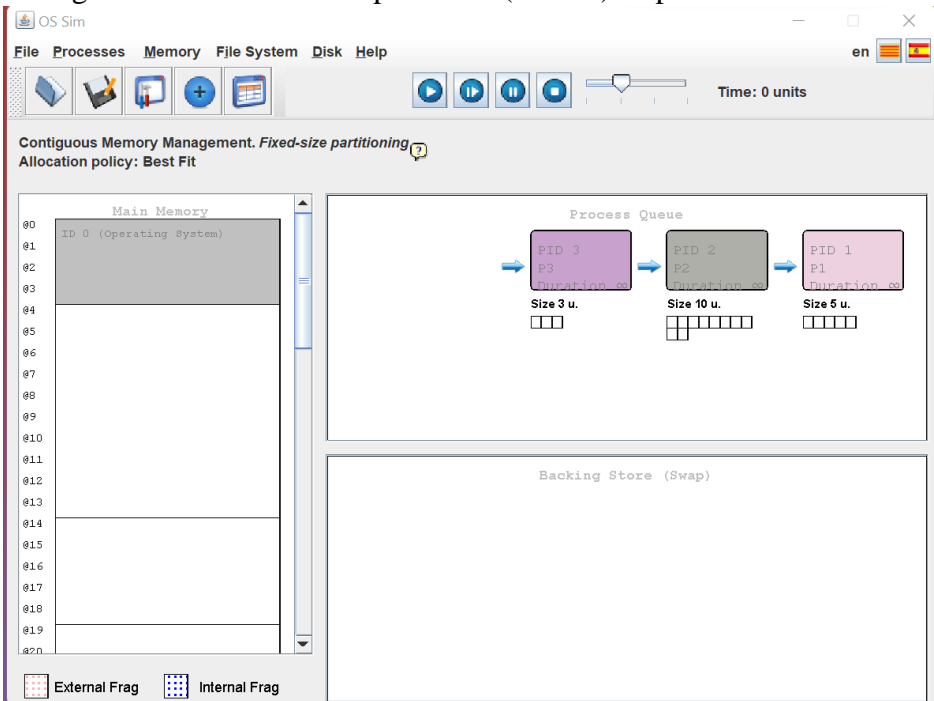


3. Lakukan analisa proses dengan melakukan klik pada tombol next

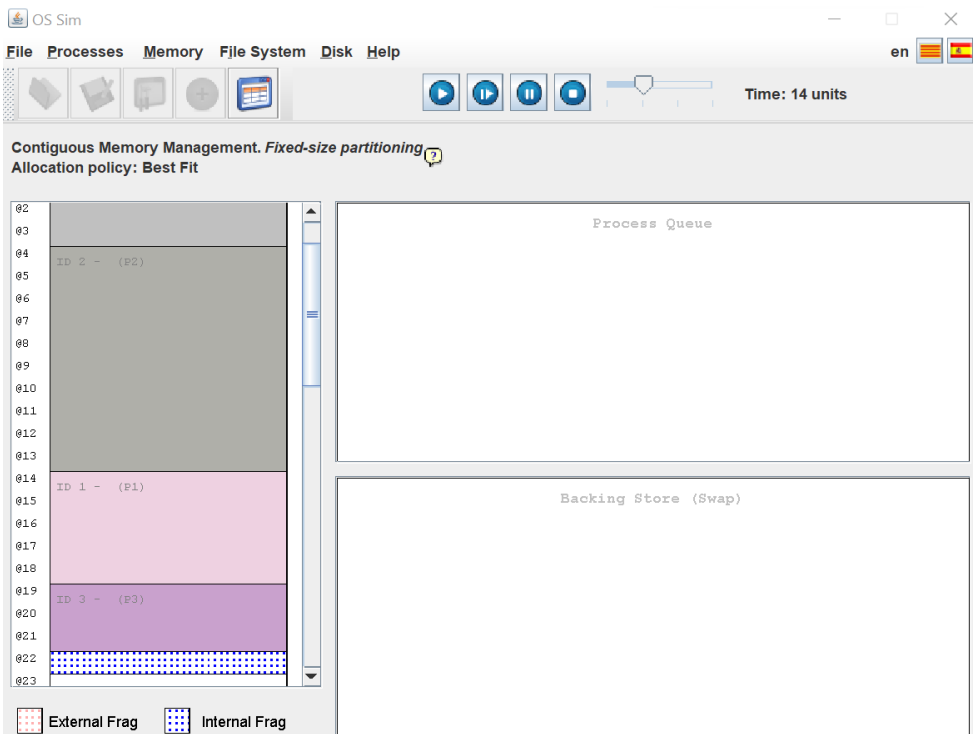


2.2 Contiguous memory management dengan menggunakan partisi berukuran tetap (fixed-size partition) dan aturan best fit

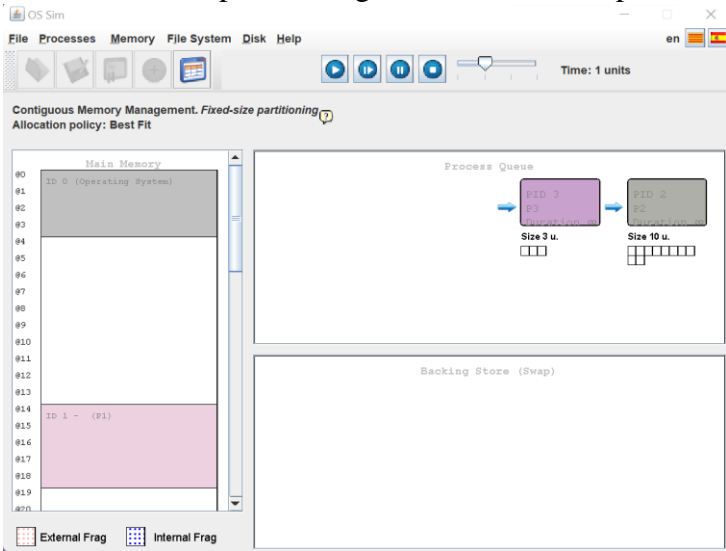
- 1. Pilih menu help >> examples...>> memory management >> Contiguous memory management with fixed-size partitions (best fit)>> pilih folder

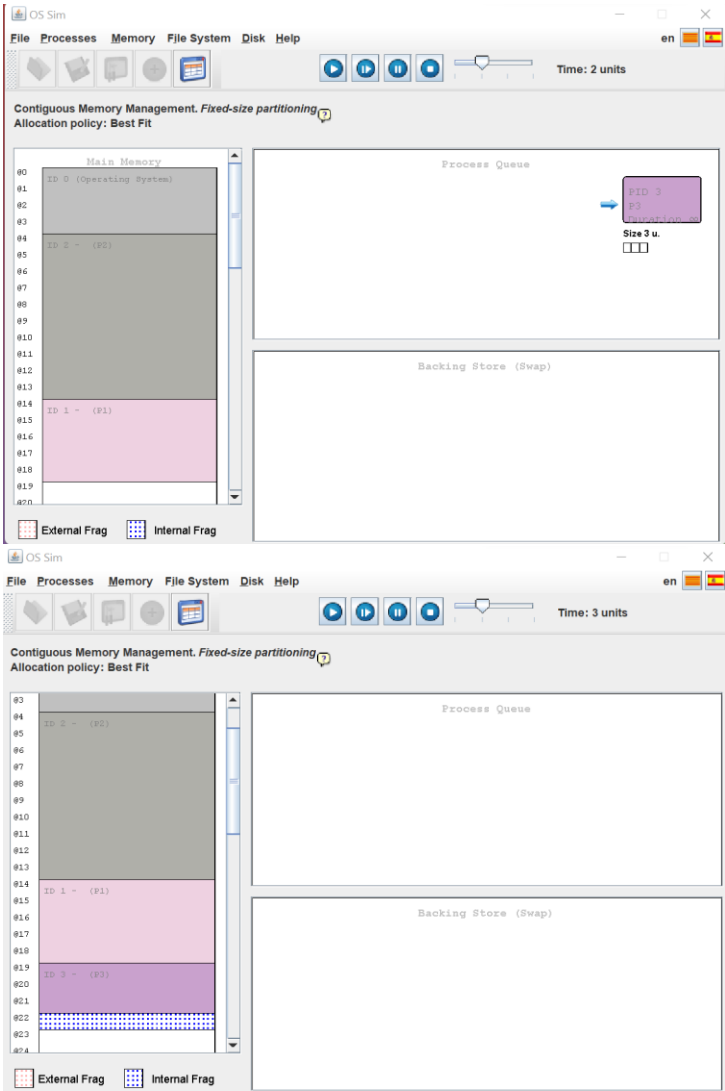


- 2. Selanjutnya klik tombol play untuk melihat proses yang terjadi dan klik tombol stop untuk berhenti



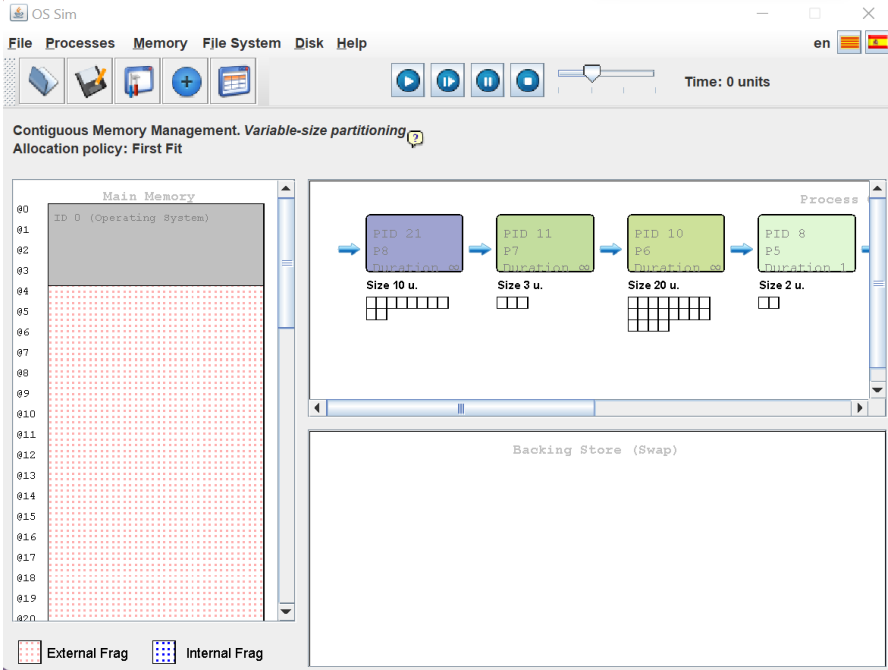
- 3. Lakukan analisa proses dengan melakukan klik pada tombol next



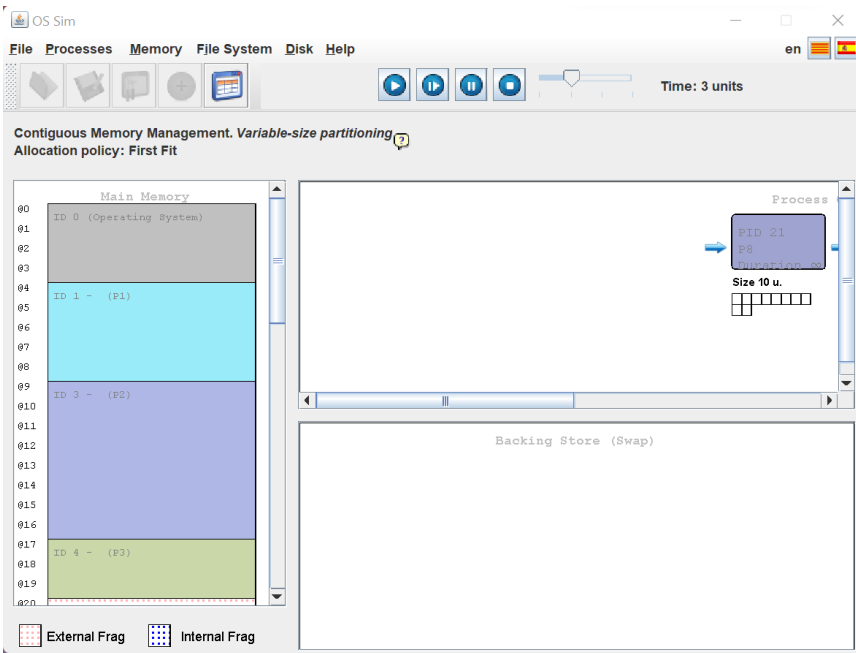


2.3 Contiguous memory management dengan menggunakan partisi berukuran tidak tetap (variable-size partition) >> defragmentas

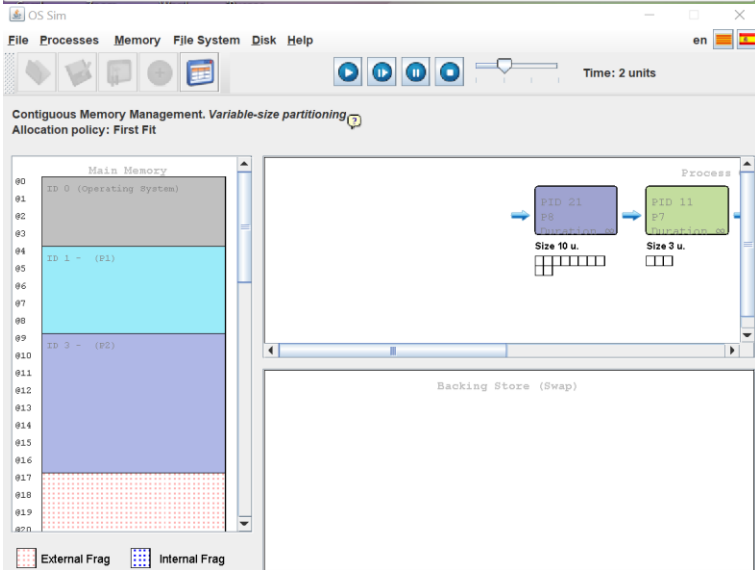
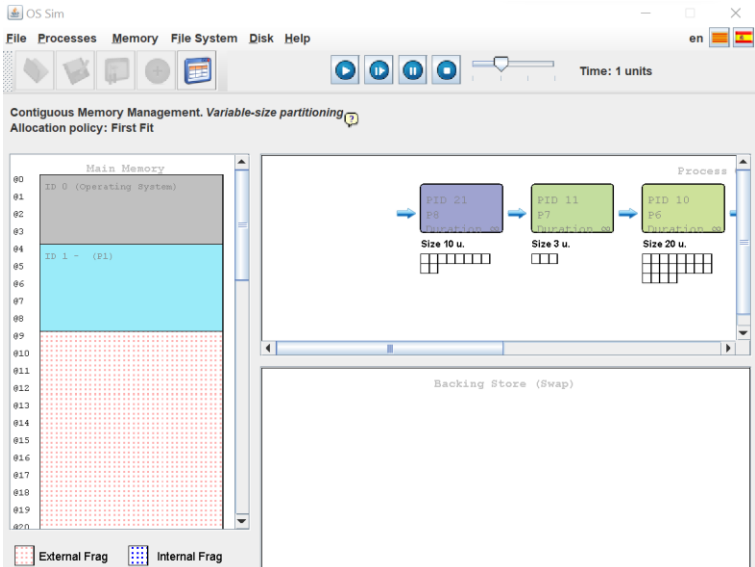
1. Pilih menu help >> examples...>> memory management >>Contiguous memory management with variable-sized partitions (Defragmentation).

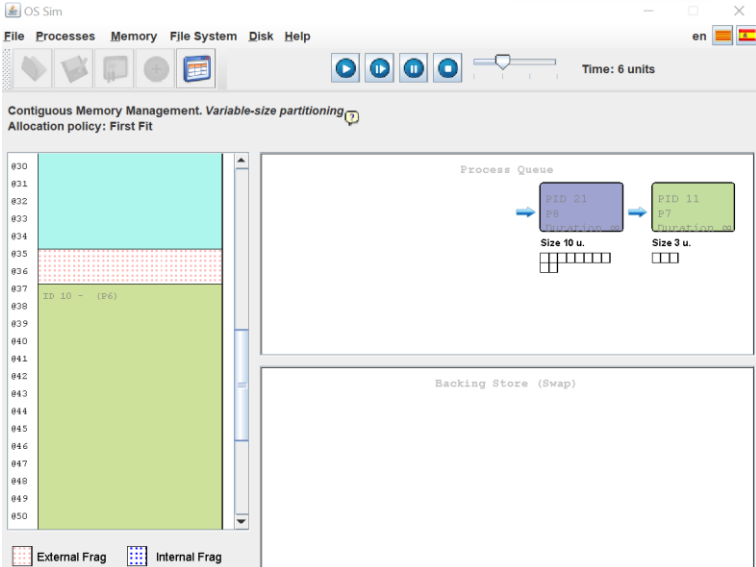
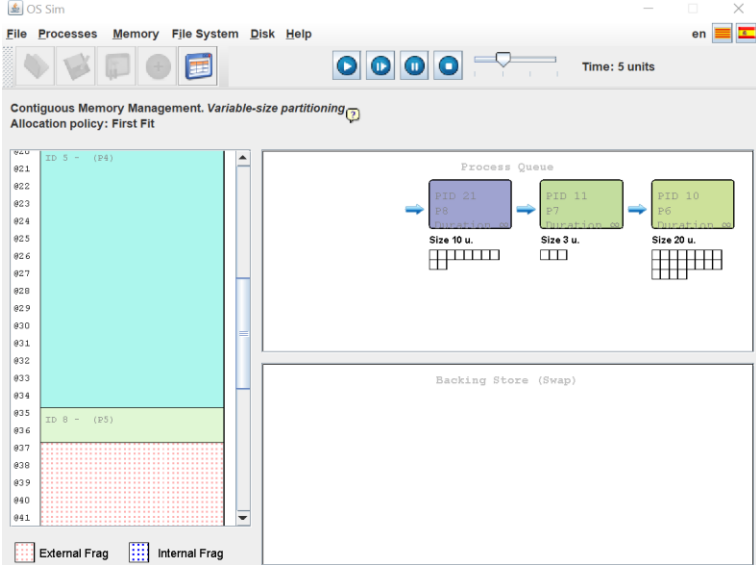
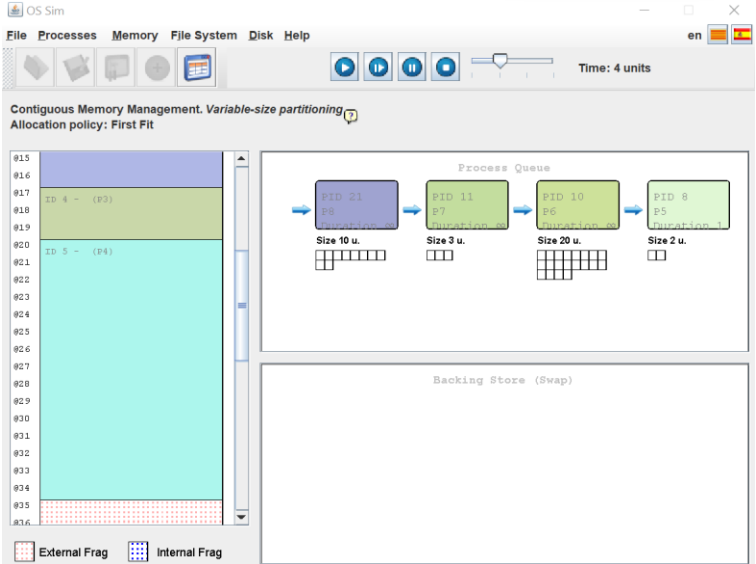
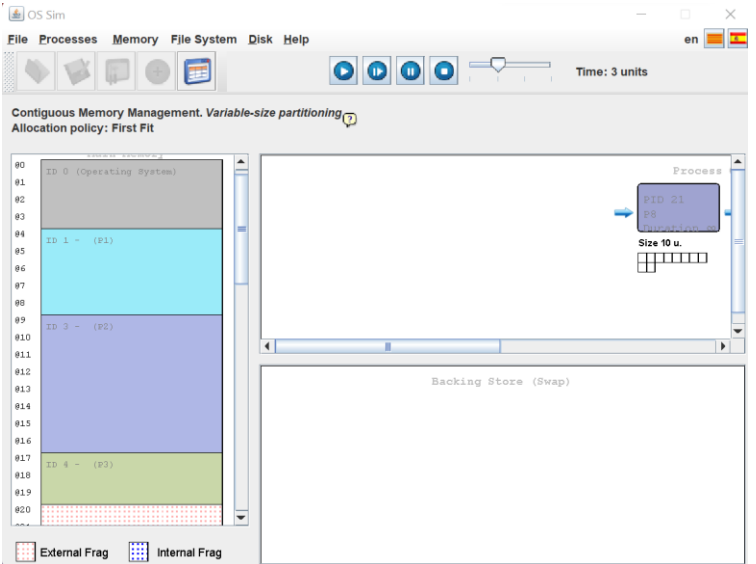


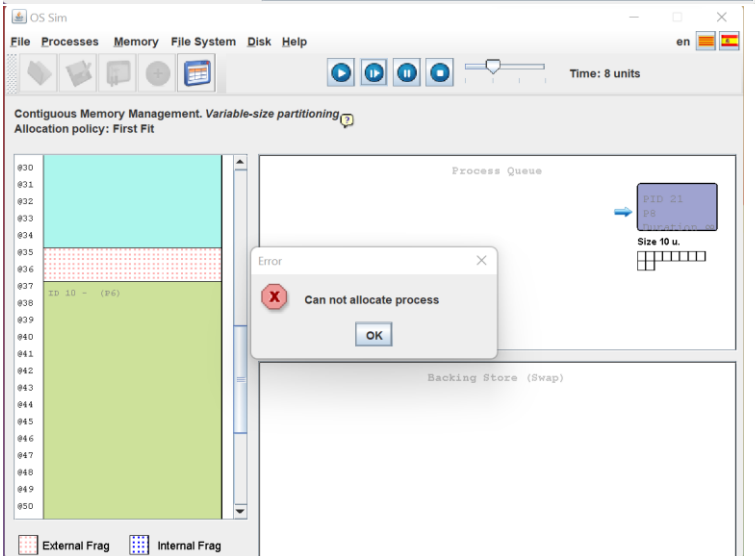
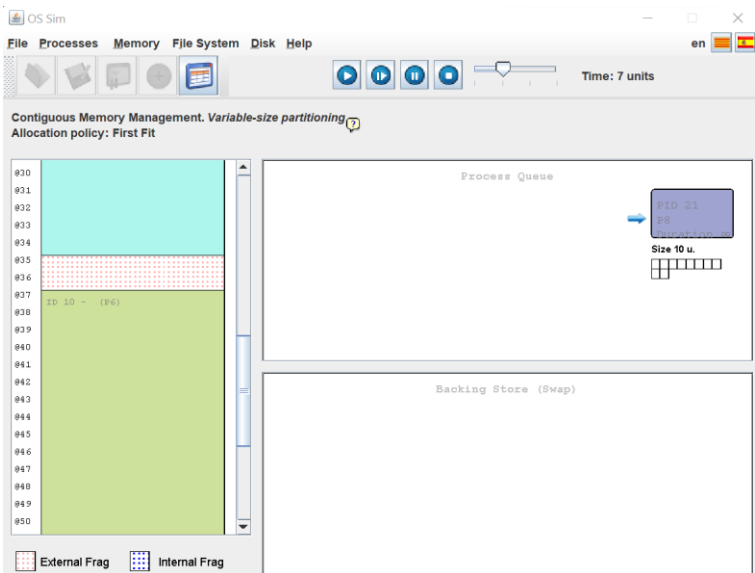
2. Selanjutnya klik tombol play untuk melihat proses yang terjadi dan klik tombol stop untuk berhenti



3. Lakukan analisa proses dengan melakukan klik pada tombol next

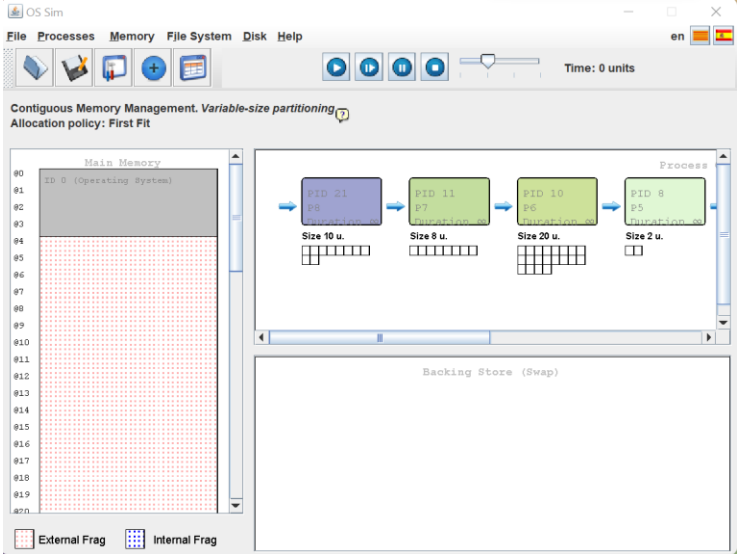




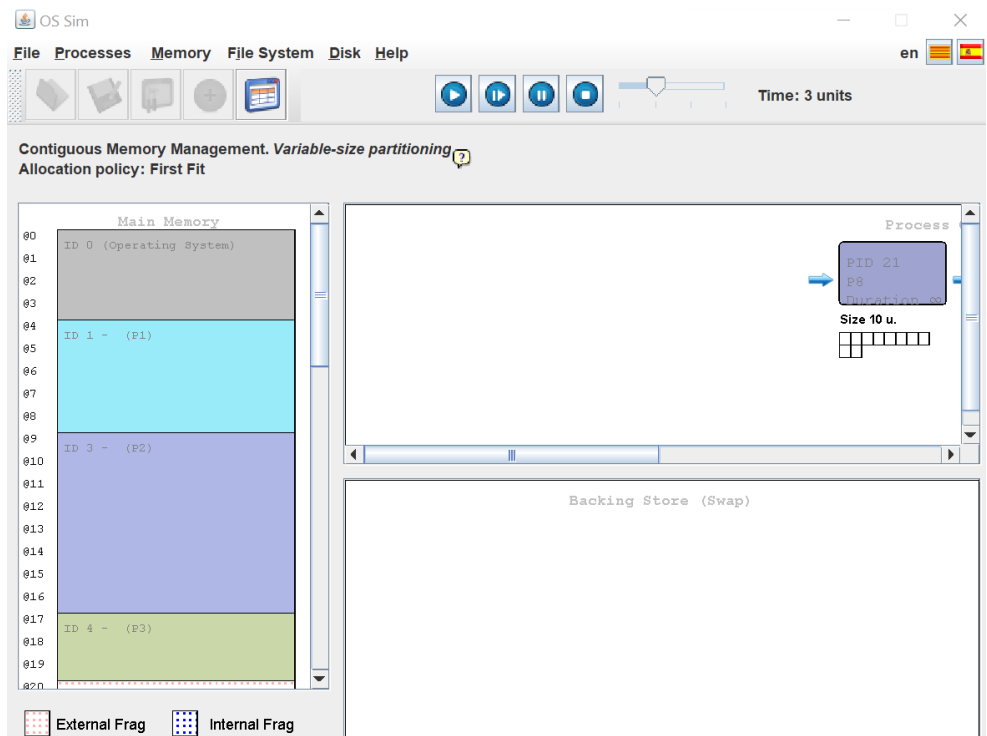


2.4 Contiguous memory management dengan menggunakan partisi berukuran tidak tetap (variable-size partition) >> swap

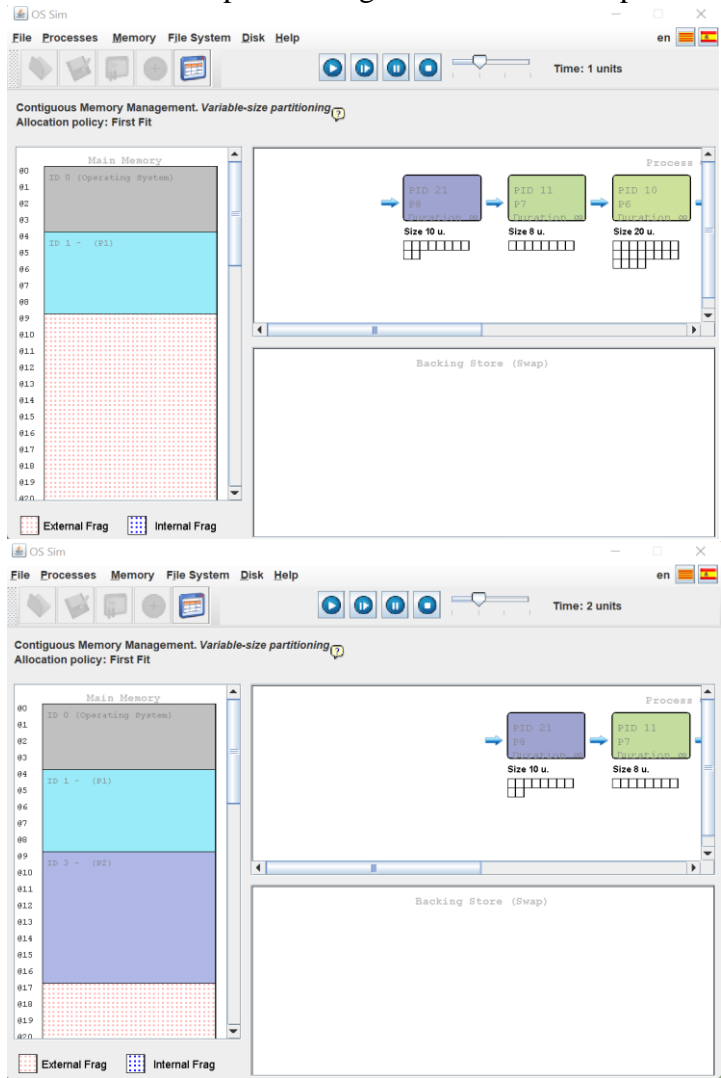
1. Pilih menu help >> examples...>> memory management >>Contiguous memory management with variable-sized partitions (Swap) >>pilih folder.

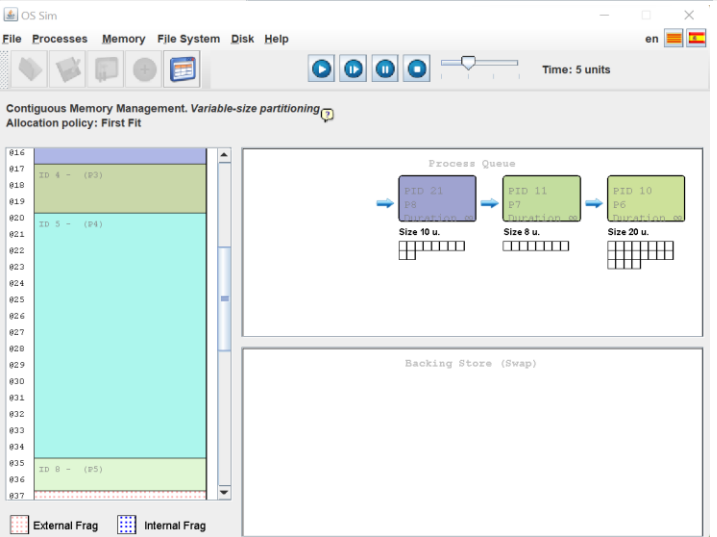
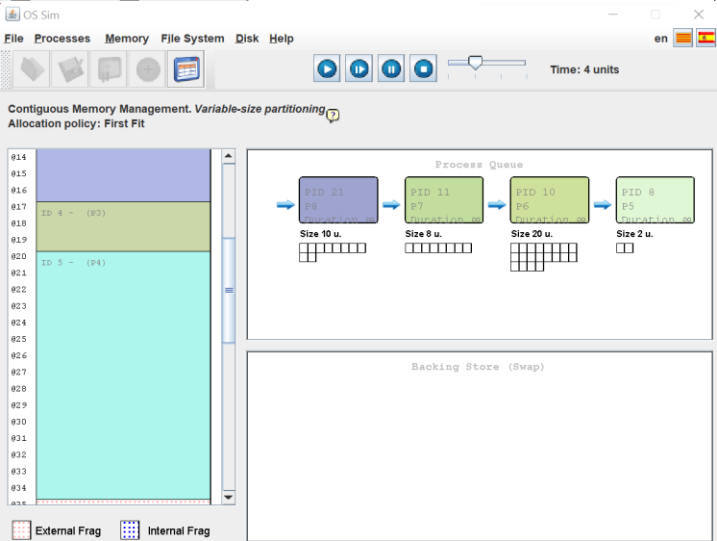
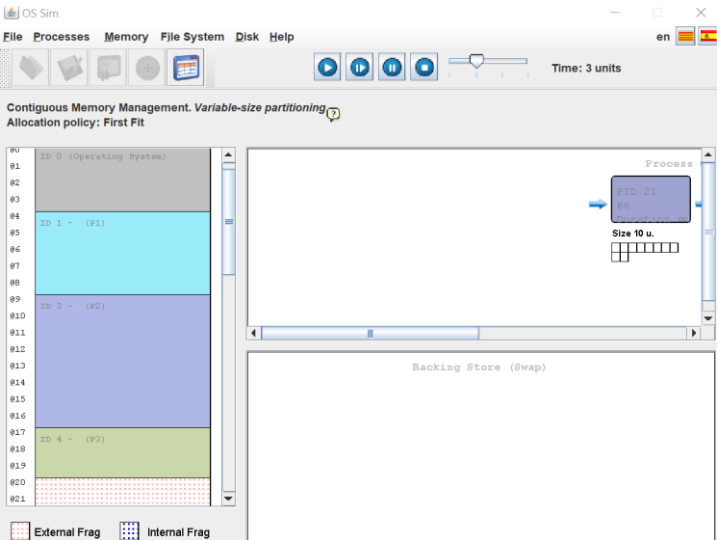


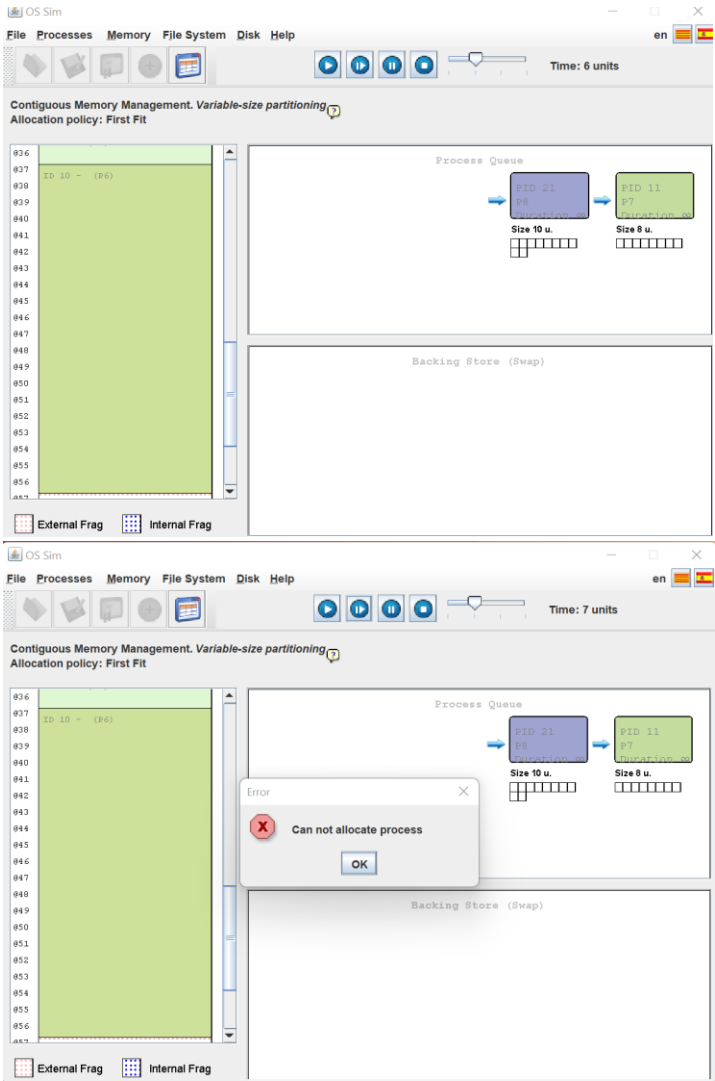
2. Selanjutnya klik tombol play untuk melihat proses yang terjadi dan klik tombol stop untuk berhenti



3. Lakukan analisa proses dengan melakukan klik pada tombol next

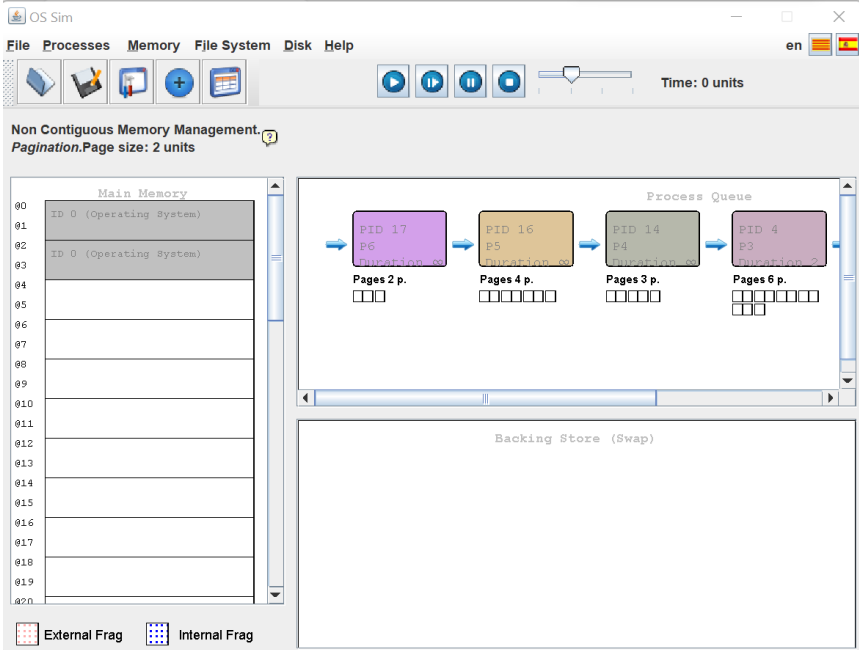




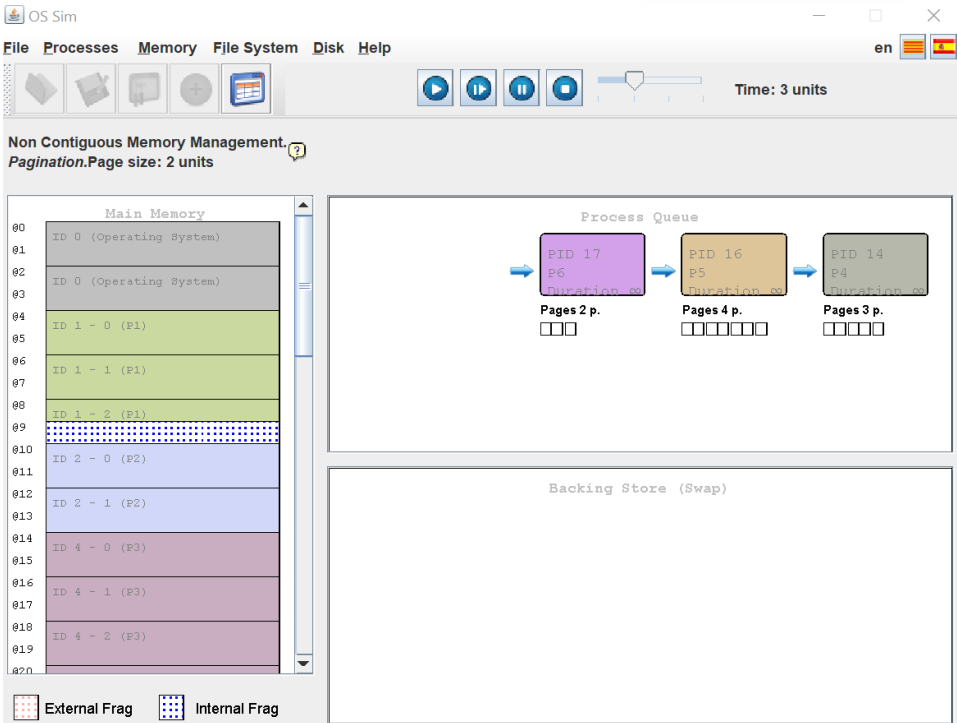


2.5 Pagation (ukuran page 2 unit)

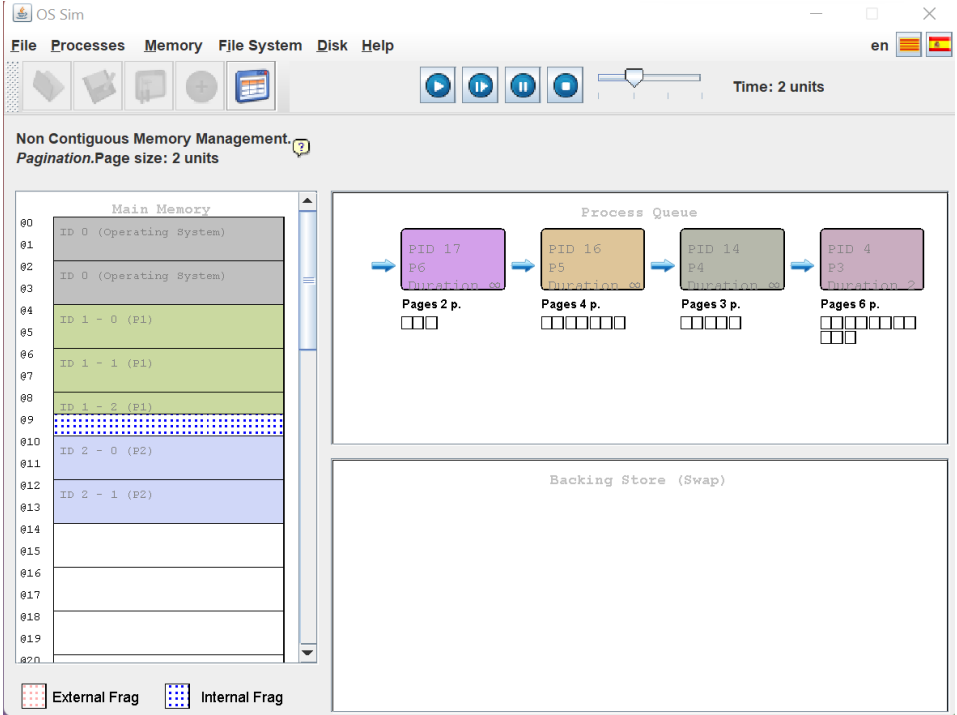
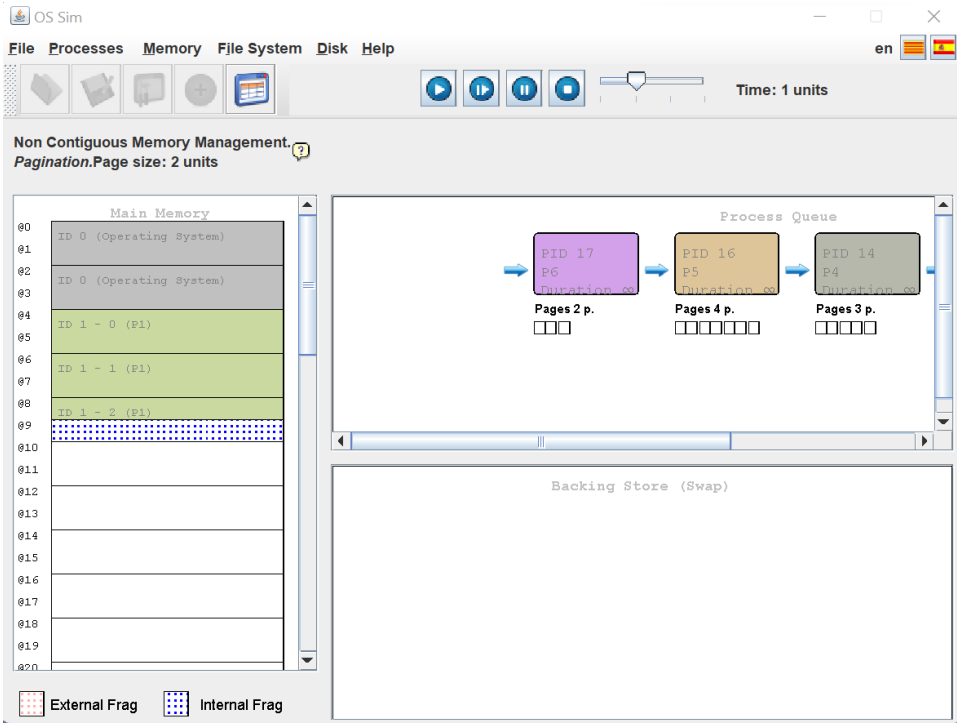
- 1. Pilih menu help >> examples...>> memory management >>Pagation) >> pilih folder

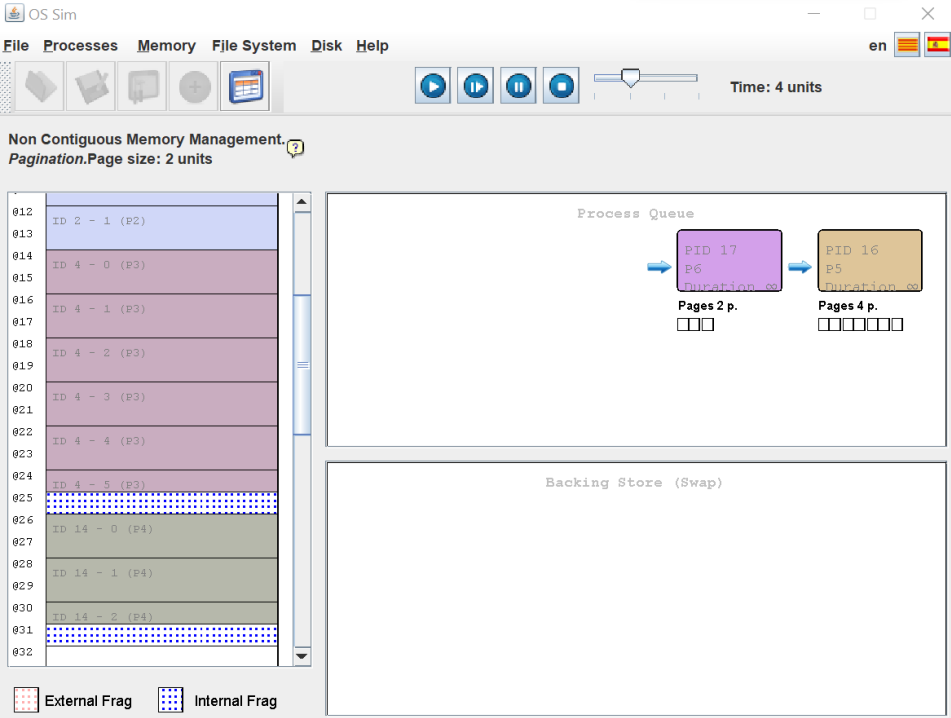
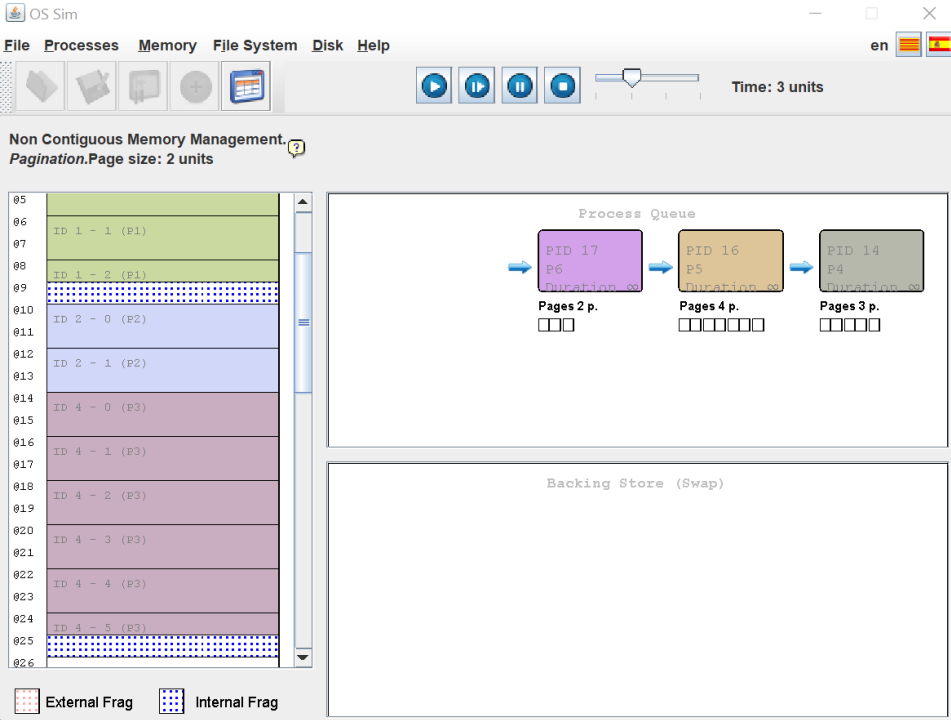


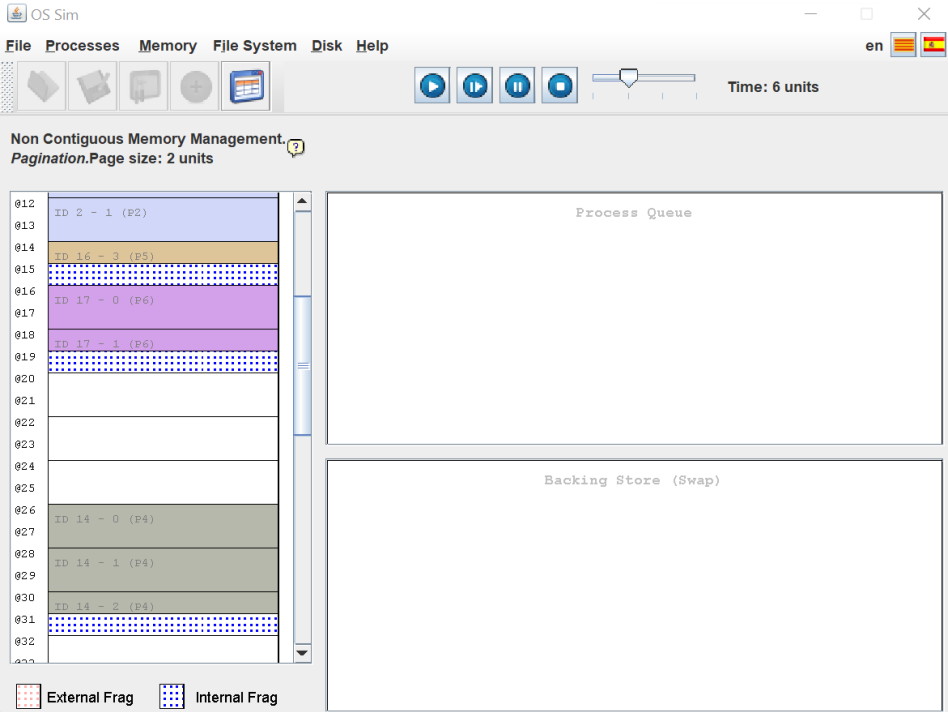
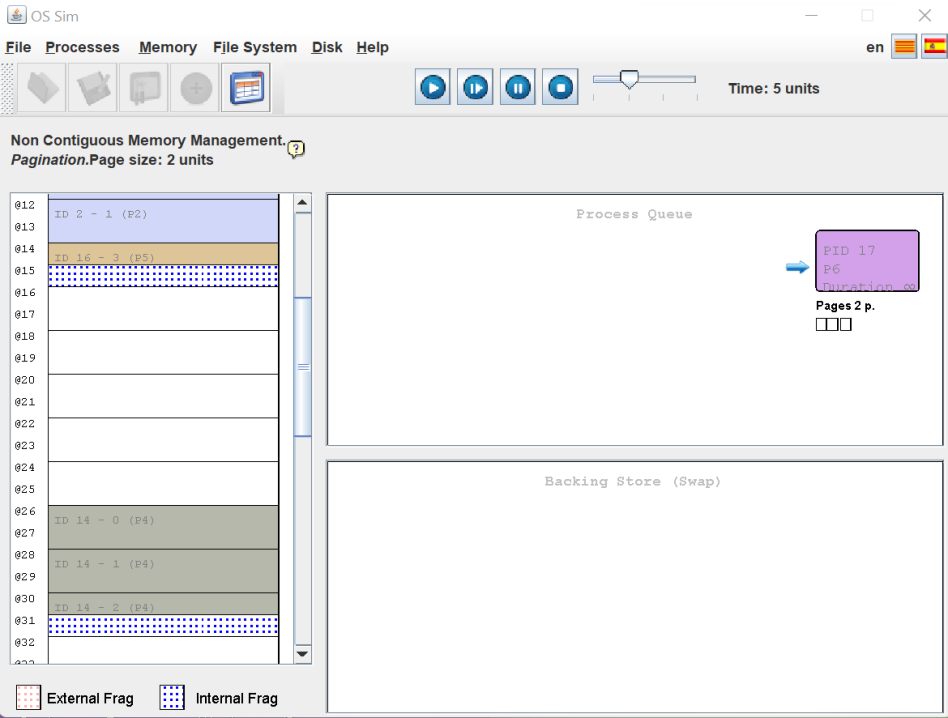
- 2. Selanjutnya klik tombol play untuk melihat proses yang terjadi dan klik tombol stop untuk berhenti



3. Lakukan analisa proses dengan melakukan klik pada tombol next

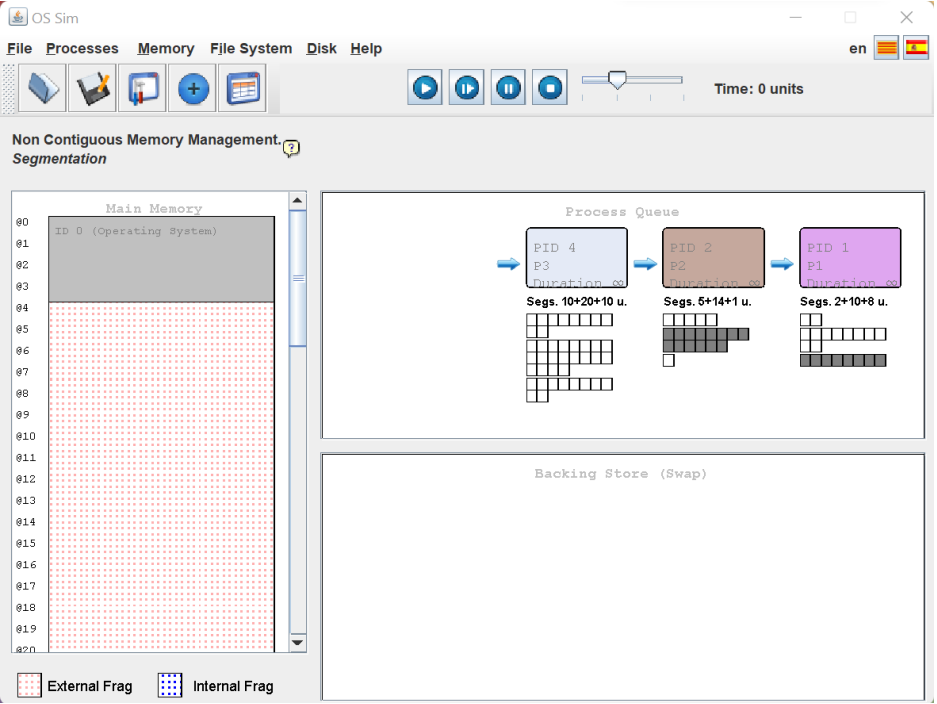




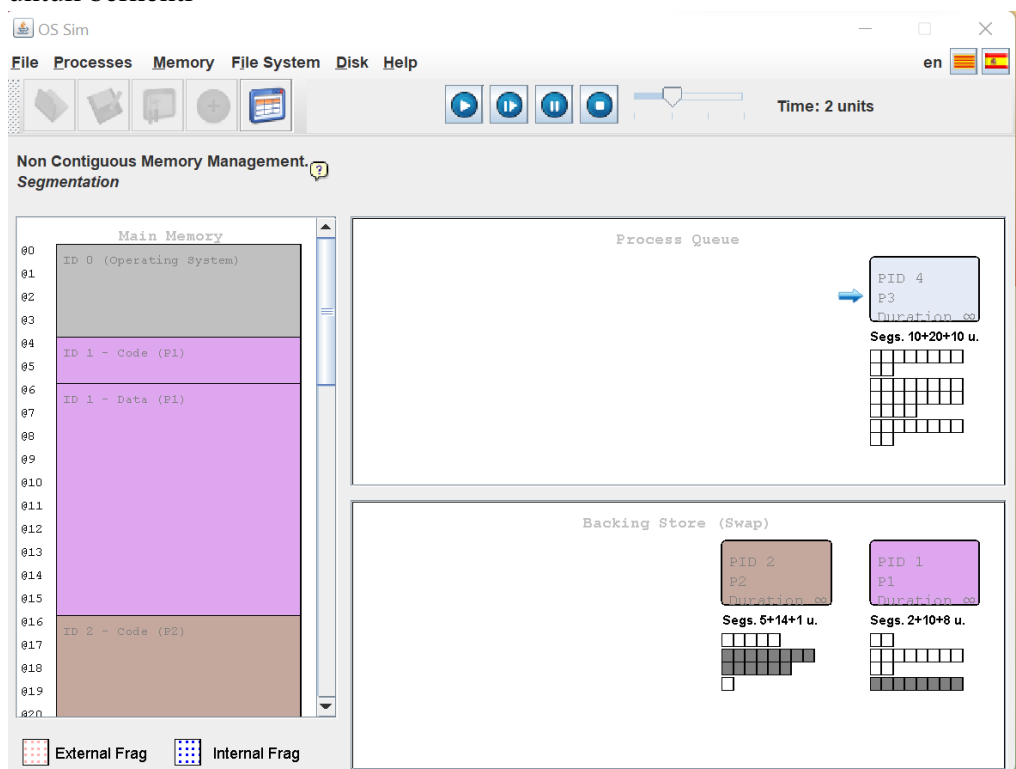


2.6 Pagation (ukuran page 2 unit)

1. Pilih menu help >> examples...>> memory management >>Segmentasi >> pilih folder



2. Selanjutnya klik tombol play untuk melihat proses yang terjadi dan klik tombol stop untuk berhenti



3. Lakukan analisa proses dengan melakukan klik pada tombol next

