Made with Goodnotes

Example 1

First- Come, First-Served (FCFS)

الـ Process الاول ← يدخل أولاً

Process	Burst Time
P1	24
P2	3
P3	3

← الوقت الي يحتاجه الـ Process

① Gantt chart اسوي ←



ابدا ← 0

الوقت الي يحتاجه ← 24

الوقت الي يحتاجه + وقت الي قبلها ← 24+3 ← 27

الوقت الي يحتاجه + وقت الي قبلها ← 27 +3 ← 30

② احسب الأوقات

- waiting time
- Response time
- Turn around time

Response time ← تكون دائماً حساوية في أنواع Non Preemtive

Turn around time ← هو الوقت الي استغرقته الـ process حتى تنفذ ← كل القيم الموجودة في Gant chart ماعدا كل قيم الـ Gantt chart ماعدا الأخيرة

① waiting time = Response time = 0 + 24 + 27 = 51

عشان اطلع الـ Averge احسم القيمة على عدد الـ process

Averge waiting time ✓ Averge Response time = $\frac{0+24+27}{3} = \frac{51}{3} = 17$

② Turn around time = 24 + 27 + 30 = 81

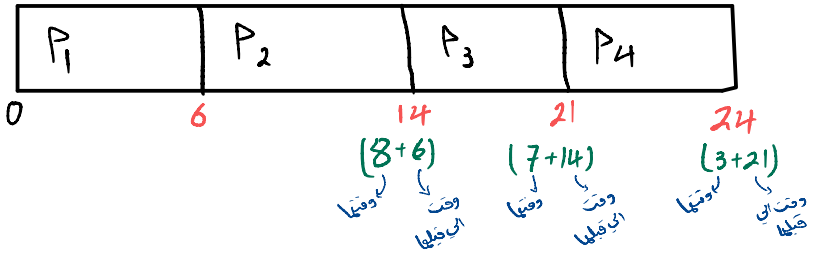
Averge turnaround tim = $\frac{81}{3} = 27$

Process	Burst Time
P1	6
P2	8
P3	7
P4	3

P_1 ← يعني هي اول process جات

P_2 ← تاني process جات

① Gantt chart ارسم →



② احسب الأوقات

gantt chart كل الأرقام في

ماعدا الأخير

Averge waiting & Respons tim $= \frac{0+6+14+21}{4} = \frac{41}{4} = 10.25$

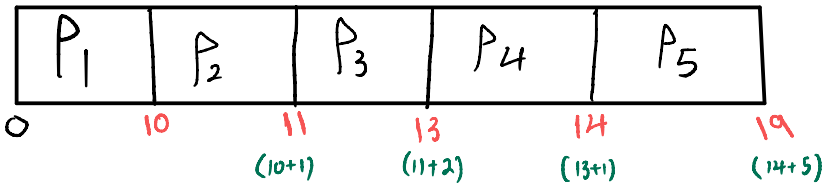
Averge turn around time $= \frac{6+14+21+24}{4} = \frac{65}{4} = 16.25$

كل الأرقام في gantt chart

ماعدا الأول

Process	Burst Time
P1	10
P2	1
P3	2
P4	1
P5	5

① Gantt chart ارسم ←



② احسب الأوقات

Average waiting & Response time = $\frac{0 + 10 + 11 + 13 + 14}{5} = \frac{48}{5} = 9.6$

وقت انتظار P_1 حتى تنفذ — وقت انتظار P_2 حتى تنفذ — وقت انتظار P_3 حتى تنفذ

Averge Turn around time = $\frac{10 + 11 + 13 + 14 + 19}{5} = \frac{67}{5} = 13.4$

وقت انتهاء P_1 من التنفيذ — وقت انتهاء P_2 من التنفيذ

استنتاج محتمله :

- ماهو وقت انتظار P_1 ← 0
- ماهو وقت انتظار P_4 ← 13

↳ waiting time

- ماهو الوقت الذي استغرقته P_5 للتنفيذ ← 19
- ماهو الوقت الذي استغرقته P_1 للتنفيذ ← 10

↳ Turn around time

★ تطبق الاسئله على جميع الأمثله السابقه في هذا النوع

Example 1

Shortest-Job-First (SJF)

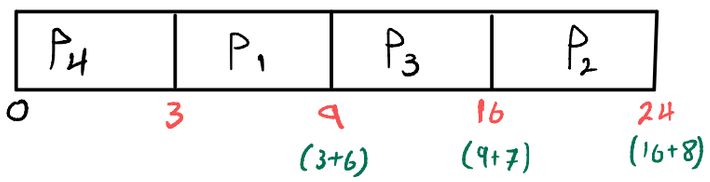
Non preemptive

Process	Burst Time
P1	6
P2	8
P3	7
P4	3

الـ Process الأقصر ← تدخل أولاً

تدخل أولا لأنها أقصر وقت ←

① Gantt Chart نرسم →



② نحسب الأوقات → بما أنها Non Premtive راح تتبع نفس الطريقة السابقة

$$\text{Averge waiting \& Response time} = \frac{0+3+9+16}{4} = \frac{28}{4} = 7$$

$$\text{Averge turnaround time} = \frac{3+9+16+24}{4} = \frac{52}{4} = 13$$

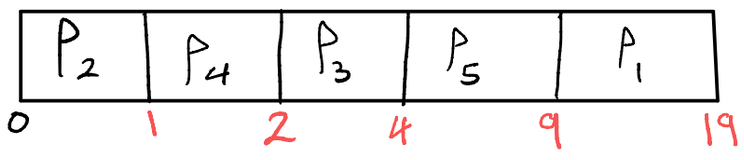
Example 2

Process	Burst Time
P1	10
P2	1
P3	2
P4	1
P5	5

ملاحظة: P_2 و P_4 عندهم نفس الوقت مين بدخل أول؟ الأقصر

الي جاء أول ← P

① gantt chart →



② حساب الأوقات

$$\text{Averge waiting \& Response time} = \frac{0+1+2+4+9}{5} = 3.2$$

$$\text{Averge turnaround time} = \frac{1+2+4+9+19}{5} = \frac{35}{5} = 7$$

Example 1

Shortest-Job-First (SJF)

Shortest-remaining-time-first

preemptive

كم طلبت الـ process حتى تنفذ؟ — متى وصلت؟

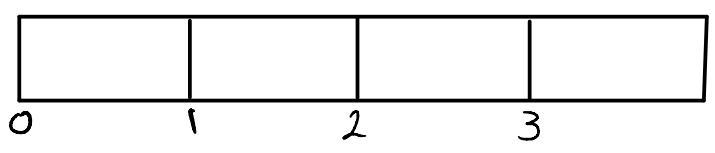
Process	Burst Time	Arrival Time
P1	8	0
P2	4	1
P3	9	2
P4	5	3

اذا كانت الـ process اللي وصلت وقتها اقل من المتبقي ← تدخل

① Gantt chart

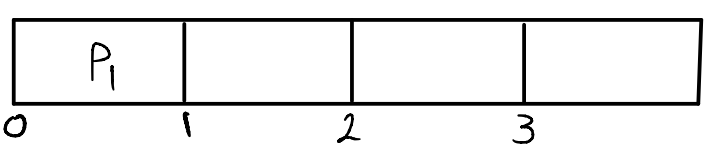
خطوات رسم الـ Gantt chart

① اربط كل قيم عمود الـ Arrival time على الـ chart



خطوات ثابته في كل مسائل هذا النوع

② اول process تدخل هي الـ process اللي جت عند الثانيه 0 ←

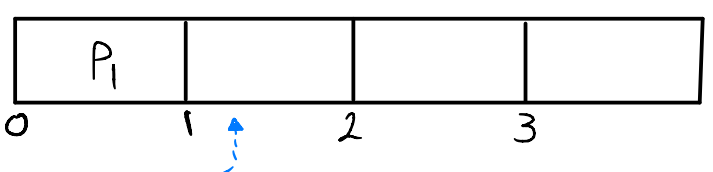


← اسأل كم طلب P_1 حتى ينفذ

P_1 طلبت من cpu 8 ثواني لكن في الرسمه عندها فقط ثانيه وحده وتخرج

يعني المتبقي من P_1 ← 8-1 = 7

③ P_1 دخلت وخرجت ولازالت تحتاج 7 ثواني



④ عشان اعرف مين الـ processes اللي بتدخل بعد P_1 ← اسأل ① مين اللي وصل عند الثانيه 1؟

P2	4	1

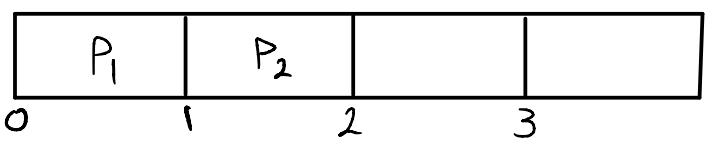
② هل P_2 لـ Burst time اقل من الوقت المتبقي لـ P_1؟

$P_1 = 7$ $P_2 = 4$

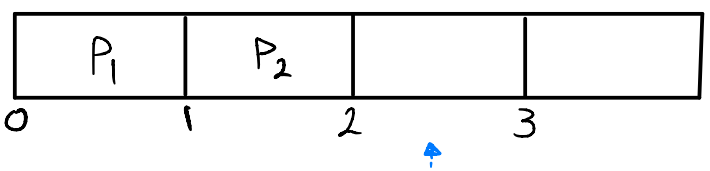
نعم ← اذا P_2 تدخل

P_2 طلبت من cpu ← 4 ثواني لكن في الرسمه عندها فقط ثانيه وحده وتخرج

يعني المتبقي من P_2 ← 4 - 1 = 3



⑤ P_2 دخلت وخرجت ولازالت تحتاج 3 ثواني



⑥ عشان اعرف مين الـ process اللي بتدخل بعد P_2 ← اسأل ① مين اللي وصل عند الثانيه 2 ؟

P3	9	2

② هل P_3 لـ Burst time اقل من الوقت المتبقي لـ P_2؟

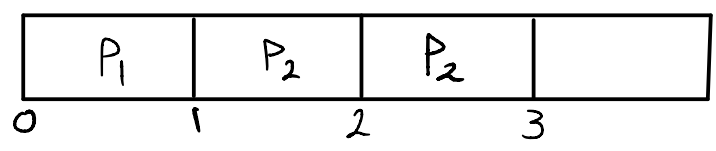
$P_3 = 9$ $P_2 = 3$

لا ← P_2 أقل من P_3 ← P_2 تدخل

⑦ P_3 لا تدخل لأن وقتها ليس اقل من P_2 بالتالي P_2 تكمل عليها ولا تخرج

الآن اصبح:

المتبقي من P_2 ← 3 - 1 = 2



⑧ عشان اعرف مين الـ process اللي بتدخل بعد P_2 ← اسأل ① مين اللي وصل عند الثانيه 3 ؟

P4	5	3

② هل P_4 لـ Burst time اقل من الوقت المتبقي لـ P_2؟

$P_4 = 5$ $P_2 = 2$

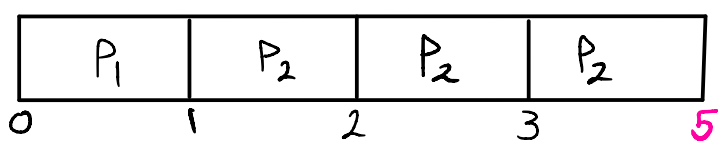
لا ← P_2 أقل من P_4 ← P_2 تدخل

⑦ P_4 لا تدخل لأن وقتها ليس اقل من P_2 بالتالي P_2 تكمل عليها ولا تخرج

الآن اصبح:

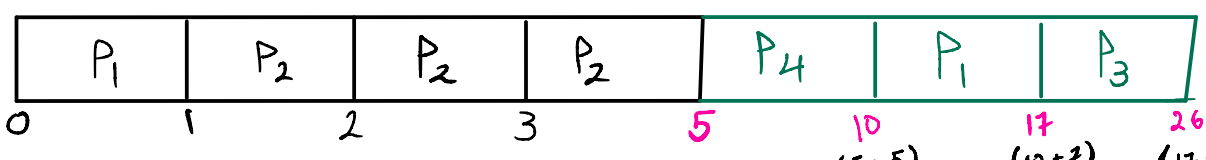
المتبقي من P_2 ← Zero

لو عدنا نفس الخطوات اللي قبل راح نخلص ان P_2 هي ايضا الاقل واح تدخل من جديد بالتالي راح تنتهي لان كان متبقي لها ثانيتين فقط

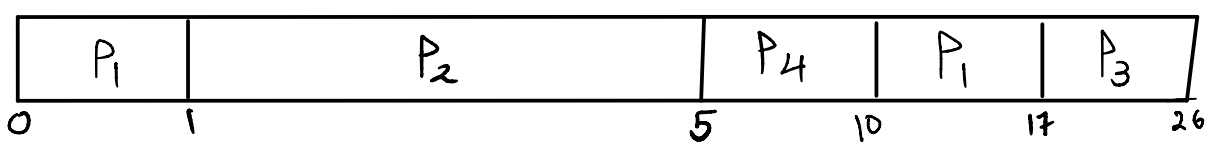


⑧ بعد ما جميع الـ Gantt chart امتلئ اسوي جدول للأوقات المتبقية لكل process وادخل الأقصر فالأقصر

Process	الوقت المتبقي
P_1	7
P_2	Zero ← انتهت
P_3	9
P_4	5



بعد الترتيب



② حساب الأوقات →

في هذا النوع فقط حيطلب
① Averge waiting time
② Averge Response time

Response time

نقرأ القيم من الـ Gantt chart

ونسأل كم هو اول زمن استجابة لهذه الـ process ؟

waiting time

أسئلة

- هل الـ process دخلت في وقتها Arrival time؟
  - هل لما دخلت انتهت فاما بدون ما تخرج وتدخل من جديد؟
    - اذا: waiting time = 0
    - P_2
    - P_2 اكلت عليها تماماً عندما دخلت في الثانية 1 وهو Burst time الخاص بها
    - $P_2 = 0$
  - هل لما دخلت لم تنهي عليها ثم خرجت ودخلت مره اخرى؟
    - waiting time = الوقت اللي دخل فيه مره ثانيه − الوقت اللي خلص فيه اول مره
    - P_1
    - P_1 دخلت في وقتها 0 لكن لم تنهي عليها تماماً وخرجت ثم عادت
    - $P_1 = 10 - 1 = 9$
- هل الـ process دخلت في وقتها Arrival time؟
  - لكن لما دخلت انتهت تماماً؟
    - waiting time = وقت انتهاء اللي قبله في الرسمه − وقت وصوله Arrival time من الجدول
    - P_3 , P_4
    - P_3 و P_4 لم يدخلو في وقتهم 2 و 3 ولكن عندما دخلو في النهاية انتهو عليهم
    - $P_3 = 17 - 2 = 15$
    - $P_4 = 5 - 3 = 2$

وقت انتظار P_2 (لما تدخل) — وقت انتظار P_1 (لما انتظرت 9 ثواني حتى تدخل من جديد) — وقت انتظار P_3 (انتظرت 15 ثانيه حتى تدخل) — وقت انتظار P_2 (انتظرت)

$$\text{Averge waiting time} = \frac{0 + 9 + 15 + 2}{4} = \frac{26}{4} = 6.5$$

Example 2

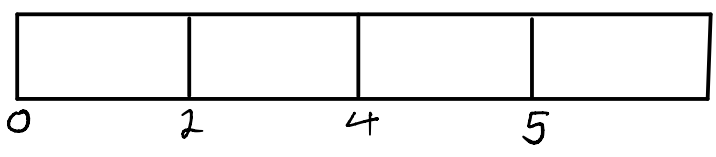
Process	Burst Time	Arrival Time
P1	7	0
P2	4	2
P3	1	4
P4	4	5

خطوات رسم الـ Gantt chart

① Gantt chart

① ارتب كل قيم عمود الـ Arrival time على الـ chart

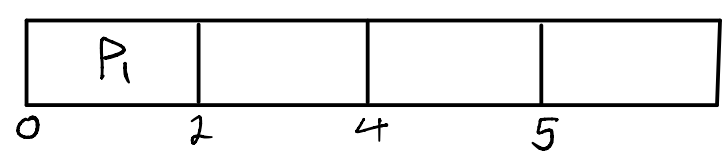
Arrival Time
0
2
4
5



خطوات ثابته في كل مسائل هذا النوع

② اول process تدخل هي الـ process اللي جت عند الثانيه 0 ←

Process	Burst Time	Arrival Time
P1	7	0



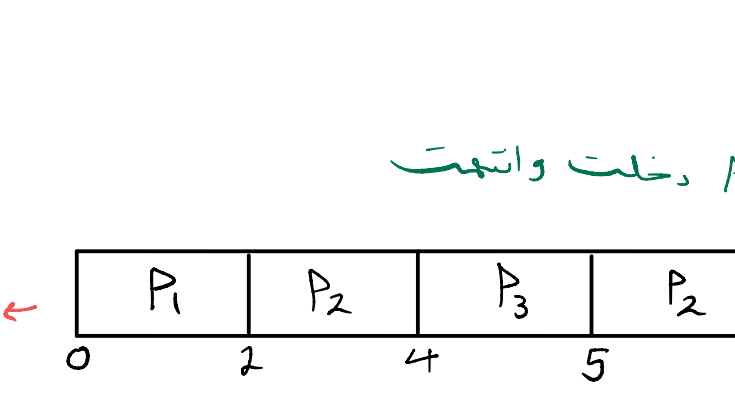
← اسأل كم طلب P_1 حتى ينفذ

Process	Burst Time	Arrival Time
P1	7	0

P_1 طلبت من cpu 8 ثواني لكن في الرسمه عندها فقط 2 ثانيتين وتخرج

يعني المتبقي من P_1 ← $7 - 2 = 5$

③ P_1 دخلت وخرجت ولازالت تحتاج {5} ثواني



④ حسنان اعرف من الـ process اللي يدخل بعد P_1 ← أسأل ① مين اللي وصل عند الثانية 2 ؟

P2	4	2

② هل Burst time لـ P_2 اقل من الوقت المتبقي لـ P_1 ؟

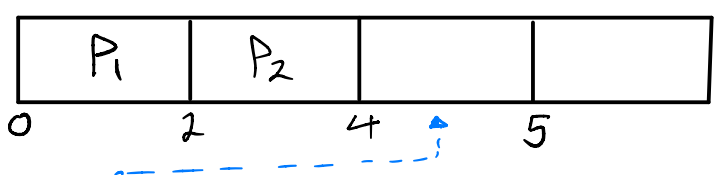
$P_1 = 5$ $P_2 = 4$

نعم ← اذا P_2 تدخل

P_2 طلبت من cpu 4 ثواني لكن في الرسمه عندها فقط ثانيتين 2 وتخرج

يعني المتبقي من P_2 ← $4 - 2 = 2$

⑤ P_2 دخلت وخرجت ولازالت تحتاج {2} ثواني



⑥ حسنان اعرف من الـ process اللي يدخل بعد P_2 ← أسأل ① مين اللي وصل عند الثانيه 4 ؟

P3	1	4

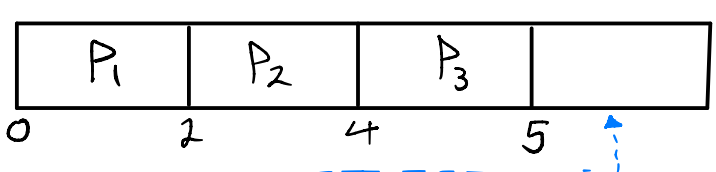
② هل Burst time لـ P_3 اقل من الوقت المتبقي لـ P_2 ؟

$P_3 = 1$ $P_2 = 2$

نعم ← اذا P_3 تدخل

⑦ P_3 دخلت و انتهت

← لأنها تحتاج فقط ثانيه واحده



⑧ حسنان اعرف من الـ process اللي يدخل بعد P_3 ← أسأل ① مين اللي وصل عند الثانيه 5 ؟

P4	4	5

② هل Burst time لـ P_4 أقل من الوقت المتبقي لـ P_2 ؟

لأن P_3 انتهت

$P_4 = 5$ $P_2 = 2$

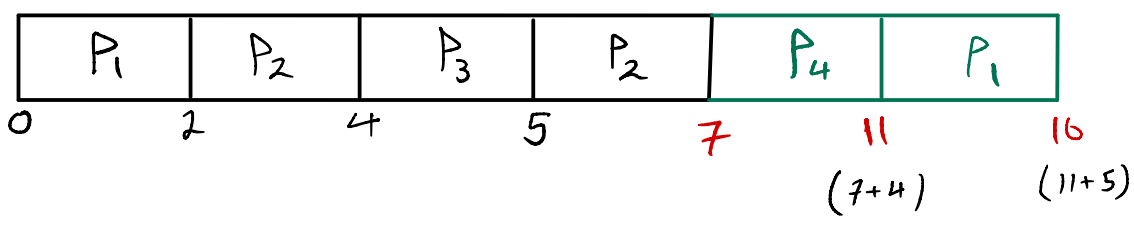
لا ← اذا P_2 تدخل

⑨ P_2 دخلت وانتهت

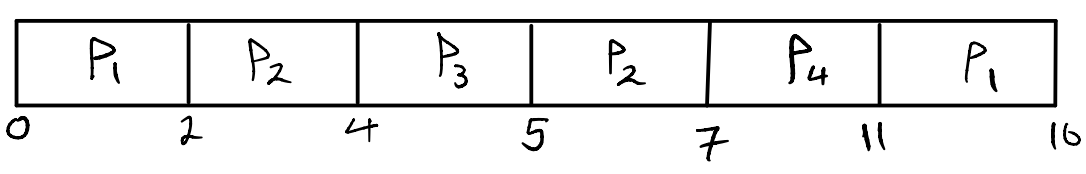
← لأنها تحتاج فقط ثانيتين

⑩ بعد ما جمع الـ Gantt chart امشي اسوي جدول للأوقات المتبقية للـ process وادخل الأقصر فالأقصر

Process	الوقت المتبقي
P_1	5
P_2	Zero → انتهت
P_3	Zero → انتهت
P_4	4



Gantt chart →



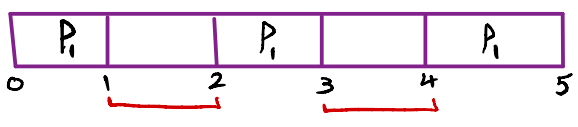
② حساب الأوقات

- دخلت في وقتها وانتهت ← Zero ← P_3
- دخلت في وقتها ولم تنتهي ← الوقت اللي خلص فيه اول مره − الوقت اللي دخل فيه ثاني مره = P_1 , P_2
  - $P_1 = 11 - 2 = 9$
  - $P_2 = 5 - 4 = 1$
- لم تدخل في وقتها ولكن لما دخلت انتهت ← وقت وصوله − وقت انتهاء اللي قبله = P_4
  - $P_4 = 7 - 5 = 2$

$$\text{Averge waiting time} = \frac{0 + 9 + 1 + 2}{4} = \frac{12}{4} = 3$$

ملاحظه

لو كان الـ gantt chart على هذا الشكل



P_1 دخلت وخرجت أكثر من مره ← كيف احسب الـ waiting time لها ؟

$P_1 = (2 - 1) + (4 - 3) = 2$

هو جميع الأوقات اللي ما شغلت الـ P_1 فيها داخل gantt chart

Example 1

Priority

Non preemptive

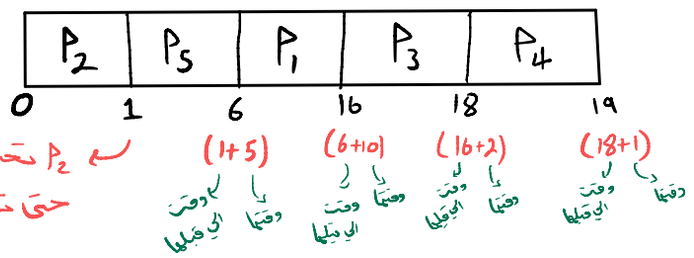
اولوية كل process

Process	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	4
P4	1	5
P5	5	2

الـProcess ذات الأولوية الأعلى (الرقم الأقل) ← تدخل أولا

① طريقة دخول الـProcess الى Gantt chart حسب اولويتها

① Gantt chart ارسم



P2 تحتاج إلى 1 ثانية حتى تنتهي

P2	1	1

وقت الدخول = وقت الخروج السابق

وقت الخروج = وقت الدخول + وقت التنفيذ

② حساب الأوقات ← كل الأوقات في الرسم ماعدا الأخير ← waiting time يتم حساب Non preemtive بما أنها مثل SJF Non Preemtive و FCFS

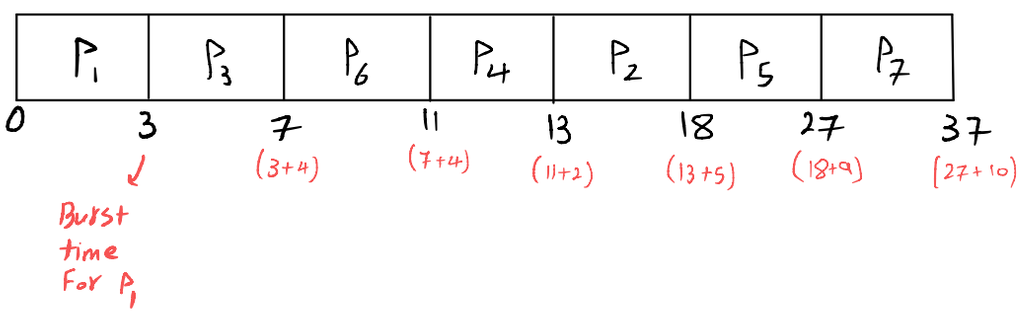
$$\text{Averge waiting time} = \frac{0+1+6+16+18}{5} = 8.2$$

Example 2

Process	Burst Time	Priority
P1	3	2
P2	5	6
P3	4	3
P4	2	5
P5	9	7
P6	4	4
P7	10	10

① طريقة دخول الـProcess الى Gantt chart حسب اولويتها

① gantt chart ارسم



② حساب الأوقات

$$\text{Averge waiting time} = \frac{0+3+7+11+13+18+27}{7} = \frac{79}{7} = 11.2$$

Made with Goodnotes

Round Robin (RR)

preemptive

Process	Burst Time
P1	24
P2	3
P3	3
Time Quantum(q) = 4	

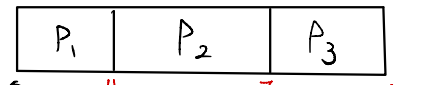
q

① تدخل الـ Process بالترتيب عادي ولكن كل Process لها حد 4 ثواني فقط

★ اذا احتاجت الـ Process اقل من 4 ثواني تخرج وتنتهي

★ اذا احتاجت الـ Process أكثر من 4 ثواني تنتظر حتى تنتهي جميع الـ Process الاخرى ثم تدخل

① ارسم gantt chart ⟵



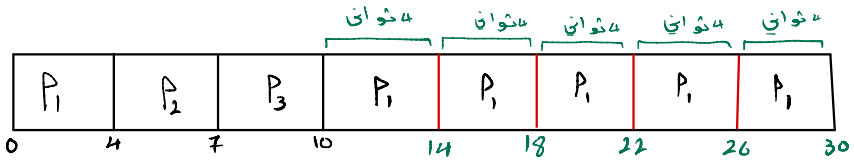
P_1 احتاجت 24 ولكنها مجبرة على الخروج بعد مرور 4 ثواني

P_2 احتاجت 3 ثواني فقط ثم خرجت

P_3 احتاجت 3 وخرجت

① المتبقي الآن هي P_1 ← وتحتاج إلى 20 ثانية

اقسم الـ gantt chart الى 4 ثواني حتى اكمل 20



② احسب الأوقات

وقت انتهاء P_3 من التنفيذ

وقت انتهاء P_2 من التنفيذ

وقت انتهاء P_1 من التنفيذ

$$\text{Averg Turn around time} = \frac{30 + 7 + 10}{3} = \frac{47}{3} = 15.67$$

ملاحظة

★ higher average turnaround time than SJF, but better response

RR يعتبر اكبر من SJF في turn around

لكن

RR يعتبر اقل وافضل من SJF في Response