

Exercice 1 (11 pts)**Partie I****- FIFO (1 pt)**

	0	4	3	7	0	3	1	3	0	7	8	0	7	8	0	2
Frame1	0	0	0	7	7	7	7	3	3	3	3	0	0	0	0	0
Frame 2		4	4	4	0	0	0	0	0	7	7	7	7	7	7	2
Frame 3			3	3	3	3	1	1	1	1	8	8	8	8	8	8
Défauts de pages	X	X	X	X	X		X	X		X	X	X				X
Taux de défauts de pages = $11/16 * 100 = 68,75\%$																

- Optimal (1 pt)

	0	4	3	7	0	3	1	3	0	7	8	0	7	8	0	2
Frame1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
Frame 2		4	4	7	7	7	1	1	1	1	8	8	8	8	8	8
Frame 3			3	3	3	3	3	3	3	7	7	7	7	7	7	7
Défauts de pages	X	X	X	X			X			X	X					X
Taux de défauts de pages = $8/16 * 100 = 50\%$																

Partie II

1. Le nombre de pages occupées par la matrice B est : $24*24/120 = 4,8 = 5$ pages. **(0,5 pt)**

Les pages de la matrice B sont notées :

Processus 1 : $P_{(1,1)}, P_{(2,1)}, P_{(3,1)}, P_{(4,1)}, P_{(5,1)}$

Processus 2 : $P_{(1,2)}, P_{(2,2)}, P_{(3,2)}, P_{(4,2)}, P_{(5,2)}$

2. Le nombre de lignes dans chaque page est :

Pages de 1-4 : 5 lignes ; Page 5 : 4 lignes **(0,5 pt)**

3. Les chaînes de références qui correspondent à l'exécution des deux processus:

Comme la matrice B est la même pour les deux processus (variable globale) et pour simplifier l'écriture des chaînes de références, les pages de B sont notées : P_1, P_2, P_3, P_4, P_5

Processus 1**Partie I : (1 pt)**

$$\underbrace{P_{(0,1)} P_1 \dots P_{(0,1)} P_1}_{110 \text{ fois}} \underbrace{P_{(0,1)} P_2 \dots P_{(0,1)} P_2}_{85 \text{ fois}} \underbrace{P_{(0,1)} P_3 \dots P_{(0,1)} P_3}_{60 \text{ fois}} \underbrace{P_{(0,1)} P_4 \dots P_{(0,1)} P_4}_{35 \text{ fois}} \underbrace{P_{(0,1)} P_5 \dots P_{(0,1)} P_5}_{10 \text{ fois}} P_{(0,1)}$$

La chaîne réduite est : $P_{(0,1)} P_1 P_{(0,1)} P_2 P_{(0,1)} P_3 P_{(0,1)} P_4 P_{(0,1)} P_5 P_{(0,1)}$

Partie II : (1 pt)

$$\underbrace{P_{(0,1)} P_1 \dots P_{(0,1)} P_1}_{100 \text{ fois}} \underbrace{P_{(0,1)} P_2 \dots P_{(0,1)} P_2}_{50 \text{ fois}} \underbrace{P_{(0,1)} P_3 \dots P_{(0,1)} P_3}_{6 \text{ fois}} P_{(0,1)}$$

La chaîne réduite est : $P_{(0,1)} P_1 P_{(0,1)} P_2 P_{(0,1)} P_3 P_{(0,1)}$

Processus 2**Partie I : (1 pt)**

$\underbrace{P_{(0,2)} P_1 \dots P_{(0,2)} P_1}_{10 \text{ fois}} \underbrace{P_{(0,2)} P_2 \dots P_{(0,2)} P_2}_{35 \text{ fois}} \underbrace{P_{(0,2)} P_3 \dots P_{(0,2)} P_3}_{60 \text{ fois}} \underbrace{P_{(0,2)} P_4 \dots P_{(0,2)} P_4}_{85 \text{ fois}} \underbrace{P_{(0,2)} P_5 \dots P_{(0,2)} P_5}_{86 \text{ fois}} P_{(0,2)}$

La chaîne réduite est : $P_{(0,2)} P_1 P_{(0,2)} P_2 P_{(0,2)} P_3 P_{(0,2)} P_4 P_{(0,2)} P_5 P_{(0,2)}$

Partie II : (1 pt)

$\underbrace{P_{(0,2)} P_3 \dots P_{(0,2)} P_3}_{12 \text{ fois}} \underbrace{P_{(0,2)} P_4 \dots P_{(0,2)} P_4}_{60 \text{ fois}} \underbrace{P_{(0,2)} P_5 \dots P_{(0,2)} P_5}_{84 \text{ fois}} P_{(0,1)}$

La chaîne réduite est : $P_{(0,1)} P_3 P_{(0,1)} P_4 P_{(0,1)} P_5 P_{(0,1)}$

4. Programme qui calcul le nombre de 0 (1pt)**Processus 1**

```
// Partie III
int NB1 = 0;
for (int i = 0; i < 24; i+=2)
    for (int j = 0; j < 24; j++)
    {
        if (B[i][j]==0) { NB1 ++; }
    }
```

Processus 2

```
// Partie III
int NB2 = 0;
for (int i = 1; i < 24; i+=2)
    for (int j = 0; j < 24; j++)
    {
        if (B[i][j]==0) { NB2 ++; }
    }
```

5. La chaîne de référence correspondant à l'exécution des deux processus en temps partagé : (2 pts)

$\underbrace{P_{(0,1)} P_1 \dots P_{(0,1)} P_1}_{72 \text{ fois}} \underbrace{P_{(0,2)} P_1 \dots P_{(0,2)} P_1}_{48 \text{ fois}} \underbrace{P_{(0,2)} P_2 \dots P_{(0,2)} P_2}_{24 \text{ fois}} \underbrace{P_{(0,1)} P_2 \dots P_{(0,1)} P_2}_{48 \text{ fois}} \underbrace{P_{(0,1)} P_3 \dots P_{(0,1)} P_3}_{24 \text{ fois}}$
 $\underbrace{P_{(0,2)} P_2 \dots P_{(0,2)} P_2}_{48 \text{ fois}} \underbrace{P_{(0,2)} P_3 \dots P_{(0,2)} P_3}_{24 \text{ fois}} \underbrace{P_{(0,1)} P_3 \dots P_{(0,1)} P_3}_{48 \text{ fois}} \underbrace{P_{(0,1)} P_4 \dots P_{(0,1)} P_4}_{24 \text{ fois}} \underbrace{P_{(0,2)} P_3 \dots P_{(0,2)} P_3}_{24 \text{ fois}}$
 $\underbrace{P_{(0,2)} P_4 \dots P_{(0,2)} P_4}_{48 \text{ fois}} \underbrace{P_{(0,1)} P_4 \dots P_{(0,1)} P_4}_{24 \text{ fois}} \underbrace{P_{(0,1)} P_5 \dots P_{(0,1)} P_5}_{48 \text{ fois}} \underbrace{P_{(0,2)} P_4 \dots P_{(0,2)} P_4}_{24 \text{ fois}} \underbrace{P_{(0,2)} P_5 \dots P_{(0,2)} P_5}_{48 \text{ fois}} P_{(0,1)} P_{(0,2)}$

Donc, la chaîne réduite est :

$P_{(0,1)} P_1 P_{(0,2)} P_1 P_{(0,2)} P_2 P_{(0,1)} P_2 P_{(0,1)} P_3 P_{(0,2)} P_2 P_{(0,2)} P_3 P_{(0,1)} P_3 P_{(0,1)} P_4 P_{(0,2)} P_3 P_{(0,2)} P_4 P_{(0,1)} P_4 P_{(0,1)} P_5$
 $P_{(0,2)} P_4 P_{(0,2)} P_5 P_{(0,1)} P_{(0,2)}$

6. Application de LRU (1pt)

	$P_{(0,1)}$	P_1	$P_{(0,2)}$	P_1	$P_{(0,2)}$	P_2	$P_{(0,1)}$	P_2	$P_{(0,1)}$	P_3
F1	$P_{(0,1)}$	$P_{(0,1)}$	$P_{(0,1)}$	$P_{(0,1)}$	$P_{(0,1)}$	P_2	P_2	P_2	P_2	P_2
F2		P_1	P_1	P_1	P_1	P_1	$P_{(0,1)}$	$P_{(0,1)}$	$P_{(0,1)}$	$P_{(0,1)}$
F3			$P_{(0,2)}$	$P_{(0,2)}$	$P_{(0,2)}$	$P_{(0,2)}$	$P_{(0,2)}$	$P_{(0,2)}$	$P_{(0,2)}$	P_3
DP	X	X	X			X	X			X
Nombre de défauts de pages = 6										

Exercice 2 (9 pts)1. Diagramme d'exécution : **(2,5 pts)** = 1ms

E/S											P1	P1					P3			
CPU	P1	P1	P1	P2	P2	P2	P3	P3	P3	P1	P2	P2	P1	P1	P4	P3	P2	P3	P3	P2
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

2. Etat des processus **(0,25 * 4 = 1pt)**

	Instant	8 ms	11 ms	12,5 ms	16,5 ms
Processus	P1	Prêt	Bloqué	Actif	Terminé
	P2	Prêt	Actif	Prêt	Actif
	P3	Actif	Prêt	Prêt	Bloqué
	P4	Prêt	Prêt	Prêt	Terminé

3. Les programmes système

Le PCB de chaque processus contient les champs suivants (en plus du champ état) :

- File : F1 ou F2
- Durée : la durée d'exécution restante, initialement égale à la durée d'exécution du processus
- Attente : la durée d'attente d'un processus dans la file F1.

Scheduler() (1 pt)**Début**

```

| etiq : Si vide(F2) Alors Défiler (F2, P-actif) ;
|           P-actif.etat := "Actif" ;
|           Lpsw(P-actif) ;
|           Sinon Si vide(F1) Alors Défiler (F1, P-actif) ;
|               P-actif.etat := "Actif" ;
|               P-actif.Attente := 0 ;
|               H := 3ms/5µs = 600;
|               Lpsw(P-actif) ;
|               Sinon goto etiq;
|           Fsi ;
| Fsi ;
| Fin.
```

R_it Horloge() // Lancée chaque 5µs (3pts)

Début

```

| <S.G.CXT>
| H--;
| Si (P-actif ≠ NULL) Alors P-actif.Durée := P-actif.Durée - 5µs ; Fsi ; (0,5 pt)
|
| Pour i:= 1 à NB_Périph
|   Faire
|     Si (Tete_file(FBi) ≠ NULL) Alors Tete_file(FBi).Durée := Tete_file(FBi).Durée - 5µs ; Fsi ; (0,5 pt)
|   Fait;
|
| Pt := Tete(F1);
| TQ (Pt ≠ NULL)
|   Faire
|     Pt.Attente := Pt.Attente + 5µs;
|     Si (Pt.Durée > T2) Alors Trouver := vrai;
|     |
|     | Pt.Attente := 0;
|     | Defiler_element( F1, Pt, P) ; (1pt)
|     | Enfiler_Tete (F2, P) ;
|     Fsi;
|     Pt := Pt.svt;
|   Fait;
|   Si (Trouver = vrai) Alors Si (P-actif ≠ NULL) et (P-actif.File = F1) Alors P-actif.etat := "Prêt";
|   |
|   | Enfiler (F1, P-actif) ; (0,5 pt)
|   | Lpsw(scheduler);
|   Fsi;
| Fsi;
|
| Si (H=0) et P-actif ≠ NULL) Alors P-actif.etat := "Prêt";
| |
| | Enfiler (F1, P-actif) ; (0,5 pt)
| | Lpsw(scheduler);
| Fsi;
| <R.CXT>
Fin.

```

N.B : Enfiler_Tete (F2, P) permet d'enfiler le processus P en tête de la file F2 tout en respectant l'ordre FIFO entre les processus retirer de la file F1.

R_it fin d'E/S() (1,5 pts)

Début

```

| <S.G.CXT>
| .....
| Si (N ≠ 0) Alors .....
| | Sinon Si  $\neg$  vide ( $F_E/S_i$ ) Alors .....
| | | Fsi ;
| | | Défiler ( $FB_i$ , p) ;
| | | p.etat := "Prêt";
| | | Si (p.File = F2)
| | | | Alors Enfiler (F2, p);
| | | | Sinon Si (p.Durée ≤ T1)
| | | | | Alors p.File := F2;
| | | | | Enfiler (F2, p) ;
| | | | | Si (P-actif ≠ NULL) et (P-actif.File = F1)
| | | | | | Alors P-actif.etat := "Prêt";
| | | | | | Enfiler (F1, P-actif) ;
| | | | | | Lpsw(scheduler);
| | | | | Fsi;
| | | | | Sinon Enfiler (F1, p);
| | | | Fsi;
| | | Fsi ;
| | Fsi;
| <R.CXT>
Fin.

```