

3. Les programmes :
Scheduler() (0,75 pt)

Debut

```

Timer1 := -1 ;
etiq : Si [] vide(F1)
Alors Défiler (P-actif, F1) ;
      P-actif.etat := "Actif" ;
      Lpsw(P-actif) ;
Sinon Si [] vide(F2)
Alors Défiler (P-actif, F2) ;
      P-actif.etat := "Actif" ;
      Timer1 := Q/Freq(H) = 2 ms / 1 ms = 2 ;
      Lpsw(P-actif) ;
      Sinon Goto etiq ;
Fsi ;

```

Fin.

Routine d'it Horloge() // Lancée chaque 1 ms (2 pts)

Debut

```

<S.G.CXT>
Si Tete_file(FB) ≠ NULL
Alors Tete_file(FB).Durée := Tete_file(FB).Durée - 1 ; (0,5 pt)
Fsi ;
Si (P-actif ≠ NULL et P-actif.File = F1)
Alors P-actif.Durée := P-actif.Durée - 1 ;
      Si (P-actif.Durée ≤ S_Remain) Alors P-actif.etat := "Prêt" ;
        P-actif.File := F2 ; (0,75 pt)
        Enfiler (P-actif, F2) ;
        Lpsw(scheduler) ;
      Fsi ;
Fsi ;
Si (Timer1 = 0 et P-actif.File = F2) Alors // Fin Q pour un processus de F2
      P-actif.etat := "Prêt" ;
      Enfiler (P-actif, F2) ; (0,75 pt)
      Lpsw(scheduler) ;
Fsi ;
<R.G.CXT>

```

Fin.

SVC (cause,) (2 pts)

Debut

<S.G.CXT>

Case cause of

Creation : Créer son PCB;

PCB.etat := "Prêt";

PCB.Durée := Estimer_Durée();

Si (PCB.Durée > S_Remain)

Alors PCB.File := F1; (0,75 pt)

Enfiler (PCB, F1);

Trier (F1);

Préemptive (PCB); // elle est notée à part

Sinon PCB.File := F2;

Enfiler (PCB, F2);

Fsi;

<R.G.CXT>

Lecture : Vérifier les droits d'accès

P-actif.etat := "Bloqué";

Enfiler (P-actif, FB);

Trier (FB); // la file FB, est triée selon la durée d'exécution restante sauf

// pour le premier processus car son E/S est en cours de traitement.

Entrée_Asynchrone.INIT(N,@);

Lpsw(scheduler);

Ecriture : Vérifier les droits d'accès

P-actif.etat := "Bloqué";

Enfiler (P-actif, FB);

Trier (FB); // la file FB, est triée selon la durée d'exécution restante sauf

// pour le premier processus car son E/S est en cours de traitement.

Sortie_Asynchrone.INIT(N,@);

Lpsw(scheduler);

Fin_processus : Libérer les ressources allouées à P-actif; (0,25 pt)

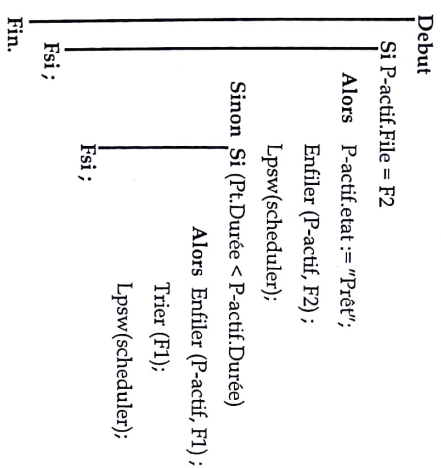
Lpsw(scheduler);

FCase;

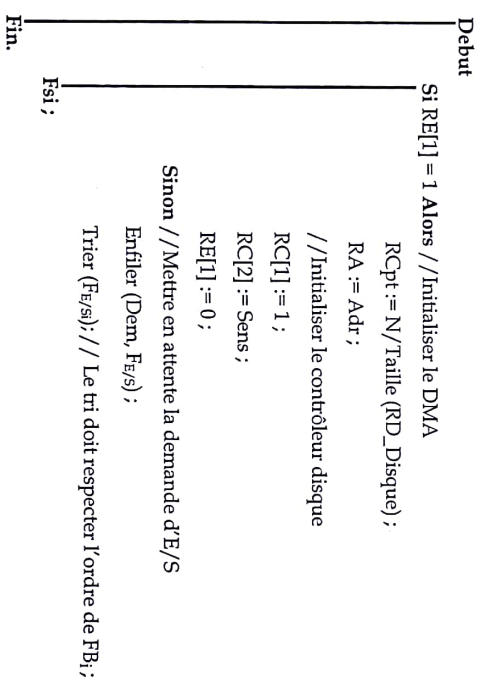
Fin.

(0,25 pt)

Préemptive (P_i) (0,75 pt)



Pilote Asynchrone_DMA.INIT(N, Adr, Sens) (0,5 pt)



Pilote_Asynchrone_DMA.Rit(N, Adr, Sens) (1 pt)

Debut

<S.G.CXT>

Si RE[2] = 1

Alors Traiter_Erreur();

Sinon Si vide ($F_{E/S}$) Alors Défiler (Dem, $F_{E/S}$);

Pilote_Asynchrone_DMA.INIT(Dem.N, Dem.Adr, Dem.Sens);

Sinon RC[1] := 0;

Fsi;

Défiler (p, FB_i);

p.etat := "Prêt";

Si ((p.File = F2) ou (p.Durée ≤ S_Remain)) Alors P-actif.File := F2;

Enfiler (p, F2);

Sinon Enfiler (p, F1);

Trier (F1);

Préemptive (p);

Fsi;

Fsi;

<R.G.CXT>

Fin.

Exercice 2 (10 pts)

1.

- FIFO (0,5 pt)

	0	4	0	2	7	0	8	2	0	5	2	8	9	0	8	5	9	4	0
Frame1	0	0	0	0	7	7	7	2	2	2	2	2	2	0	0	0	9	9	9
Frame 2		4	4	4	4	0	0	0	0	5	5	5	5	5	8	8	8	4	4
Frame 3				2	2	2	8	8	8	8	8	8	9	9	9	5	5	5	0
Défauts de pages	X	X		X	X	X	X	X		X			X	X	X	X	X	X	X
Taux de défauts de pages = 15/19 = 73,68%																			

- LRU (0,5 pt)

	0	4	0	2	7	0	8	2	0	5	2	8	9	0	8	5	9	4	0
Frame1	0	0	0	0	0	0	0	0	0	0	0	8	8	8	8	8	8	4	4
Frame 2		4	4	4	7	7	7	2	2	2	2	2	2	0	0	0	9	9	9
Frame 3				2	2	2	8	8	8	5	5	5	9	9	9	5	5	5	0
Défauts de pages	X	X		X	X		X	X		X		X	X	X		X	X	X	X
Taux de défauts de pages = 14/19 = 73,68%																			

- Seconde chance (1 pt)

	0	4	0	2	7	0	8	2	0	5	2	8	9	0	8	5	9	4	0
Frame 1	0	0	0	0	7	7	7	2	2	2	2	2	9	9	9	5	5	5	0
	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
Frame 2		4	4	4	4	0	0	0	0	0	0	8	8	8	8	8	9	9	9
		1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	1	1	0
Frame 3				2	2	2	8	8	8	5	5	5	5	0	0	0	0	4	4
				1	0	0	1	0	0	1	1	1	0	1	1	0	0	1	0
Défauts de pages	X	X		X	X	X	X	X		X		X	X	X		X	X	X	X
Taux de défauts de pages = $15/19 = 78,94\%$																			

2. Ce programme initialise une matrice carrée A[96, 96] (Partie I), puis il calcule la somme de la diagonale principale (Partie II), ensuite, il réalise la transposée de A par permutation de ses éléments (Partie III). (0,75 pt)
3. Le nombre de pages de A est $(96 \times 96) / 480 = 19,2 \approx 20$ pages. (0,25pt)
4. Le nombre d'éléments dans chaque page est :
 - 480 éléments pour les pages de 1 à 19, et (0,25 pt)
 - 96 éléments pour la page 20. (0,25 pt)
 Le nombre maximal de lignes dans une page est : $480 / 96 = 5$ lignes. (0,25 pt)
5. Les adresses du premier et du dernier élément les 5 premières pages : (1,25 pts = 0,25 pt pour chaque page)

Page	@ premier élément	@ dernier élément
Page 1	480	959
Page 2	960	1439
Page 3	1440	1919
Page 4	1920	2399
Page 5	2400	2879

6. La chaîne de référence qui correspond à l'initialisation de la matrice A (Partie I du programme) est : (1 pt)

$\underbrace{01 \dots 01}_{480 \text{ fois}} \underbrace{02 \dots 02}_{480 \text{ fois}} \underbrace{03 \dots 03}_{480 \text{ fois}} \underbrace{04 \dots 04}_{480 \text{ fois}} \dots \underbrace{019 \dots 019}_{480 \text{ fois}} \underbrace{020 \dots 020}_{96 \text{ fois}}$

Donc, la chaîne réduite est :

0 1 0 2 0 3 0 4 0 5 0 6 0 19 0 20

7. La chaîne de référence qui correspond au calcul de somme des éléments de la diagonale de la matrice A (Partie II du programme) est : (1 pt)

$\underbrace{010 \dots 010}_{5 \text{ fois}} \underbrace{020 \dots 020}_{5 \text{ fois}} \underbrace{030 \dots 030}_{5 \text{ fois}} \underbrace{040 \dots 040}_{5 \text{ fois}} \dots \underbrace{0190 \dots 0190}_{5 \text{ fois}} \underbrace{0200 \dots 0200}_{1 \text{ fois}}$

Donc, la chaîne réduite est :

0102030405060190200

8. La chaîne de référence qui correspond au calcul de la transposée de la matrice A (Partie III du programme) et ce pour $0 \leq i \leq 12$ est : (3 pts)

$\underbrace{0101101 \dots 0101101}_{10 \text{ fois}}$ $\underbrace{0201201 \dots 0201201}_{10 \text{ fois}}$ $\underbrace{0202202}_{1 \text{ fois}}$ $\underbrace{0201201 \dots 0201201}_{5 \text{ fois}}$
 $\underbrace{0202202}_{2 \text{ fois}}$ $\underbrace{0201201 \dots 0201201}_{5 \text{ fois}}$ $\underbrace{0202202 \dots 0202202}_{3 \text{ fois}}$ $\underbrace{0201201 \dots 0201201}_{5 \text{ fois}}$
 $\underbrace{0202202 \dots 0202202}_{4 \text{ fois}}$ $\underbrace{0301301 \dots 0301301}_{5 \text{ fois}}$ $\underbrace{0302302 \dots 0302302}_{5 \text{ fois}}$
 $\underbrace{0301301 \dots 0301301}_{5 \text{ fois}}$ $\underbrace{0302302 \dots 0302302}_{5 \text{ fois}}$ $\underbrace{0303303}_{1 \text{ fois}}$

Donc, la chaîne réduite est :

$01 \underbrace{0201201 \dots 0201201}_{10 \text{ fois}}$ $02 \underbrace{0201201 \dots 0201201}_{5 \text{ fois}}$ $02 \underbrace{0201201 \dots 0201201}_{5 \text{ fois}}$ 02
 $\underbrace{0201201 \dots 0201201}_{5 \text{ fois}}$ 02 $\underbrace{0201201 \dots 0201201}_{5 \text{ fois}}$ 02 $\underbrace{0301301 \dots 0301301}_{5 \text{ fois}}$
 $\underbrace{0302302 \dots 0302302}_{5 \text{ fois}}$ $\underbrace{0301301 \dots 0301301}_{5 \text{ fois}}$ $\underbrace{0302302 \dots 0302302}_{5 \text{ fois}}$ 03