

Exercice 1 : Choisir la ou les bonnes réponses et argumenter.

1. A la première génération d'ordinateurs, le système d'exploitation était :

- (a) un robot (b) un bonhomme (c) il n'y avait pas de SE.

Rep : c. rappeler le principe de porte ouverte

2. A la 2nd génération, sont apparues les cartes perforées. Elles servaient :

- (a) A la lecture des programmes et des données, donc en entrée de l'ordinateur
(b) A l'écriture des résultats, donc à la sortie de l'ordinateur
(c) Entre les deux.

Rep : a. rappel

3. La configuration "Lecteur de carte perforées → ordinateur → imprimante" a posé un problème :

- (a) Pendant la lecture des données, le processeur restait inoccupé.
(b) Pendant l'impression des résultats, l'unité centrale restait inoccupée.
(c) L'ordinateur était trop lent par rapport au lecteur et à l'imprimante.

Rep : a et b

4. Pour résoudre ce problème :

- (a) On remplace les cartes perforées par les bandes magnétiques.
(b) On remplace l'imprimante par les bandes magnétiques.
(c) On interpose des lecteurs de bandes magnétiques entre le lecteur de cartes et l'ordinateur et entre celui-ci et l'imprimante.

Rep : c

5. Le problème qui persistait était :

- (a) La rapidité des E/S par rapport au processeur.
(b) la lenteur des périphériques d'E/S par rapport au processeur.
(c) Aucun. Tout était parfait.

Rep : b

6. Le premier SE est réellement apparu avec le traitement par lots (Batch processing). A cette époque, un programme s'appelait :

- (a) un travail (b) un lot (c) un job

Rep : c

7. Pourquoi a-t-on eu recours au batch processing ?

- (a) Pour maximiser le rendement du processeur
(b) Pour minimiser le temps d'attente du processeur entre la fin d'un job et le début du suivant
(c) Pour décharger l'opérateur qui était responsable du chargement manuel des jobs

Rep : a, b

8. A cette époque, régnait toujours la monoprogrammation qui veut dire :

- (a) que le processeur exécute un seul programme à la fois
(b) qu'à un moment donné, il y avait un seul programme en mémoire centrale
(c) que le processeur exécute une seule instruction à la fois

Rep : b

9. A la 3^{ème} génération est apparue la multiprogrammation qui veut dire :

- (a) que le processeur exécute plusieurs programmes à la fois
(b) qu'à un moment donné, il y a plusieurs programmes en mémoire centrale
(c) que le processeur exécute plusieurs instructions à la fois

Rep : b

11. Un système temps partagé suppose que :

- (a) il y a plusieurs programmes en mémoire centrale
(b) il y a plusieurs utilisateurs connectés au même ordinateur
(c) il y a plusieurs ordinateurs.

Rep : a, b

12. Un système temps partagé est caractérisé par:

- (a) L'allocation de ressources selon un quantum de temps
(b) L'allocation du CPU selon un quantum de temps.
(c) Un quantum de temps est alloué à chaque usager.

Rep : b

13. Quelle est la séquence de boot d'un PC

- (a) BIOS, chargement du SE, Post
(b) Post, BIOS, chargement du SE
(c) BIOS, Post, chargement du SE

Rep : c

14. Un système multiprogrammé est :

- (a) un système temps partagé (b) un système monotâche
(c) un système dont l'exécution des programmes et les opérations d'E/S se font en parallèle

Rep : c

15. Les OS actuels sont :

- (a) monoprogrammés (b) Multiprogrammés (c) les deux ?

Rep : cela dépend de l'usage auquel ils sont destinés : usage courant : multiprogrammé

16. Windows est-il un OS ? Si oui, est-il : monoprogrammé ou multiprogrammé ? monotâches ou multitâches ? Monopostes ou multipostes ?

Rep : Multiprogrammé, multitâche, mais monoposte ?

17. Même question pour Unix.

Rep : Multiprogrammé, multitâche, multiposte.

18. Le lancement du SE est initié par un programme appelé :

- (a) I/O.sys (b) BIOS (c) Bootstrap

Rep : c

19. Qu'est-ce que la programmation parallèle ? *Rep : Un ordi + plusieurs processeurs.*

Solution Exercice 2 (TD1):

1°)

$$a) D = \frac{60}{15} = 4 \quad ; \quad r = \frac{1}{15} = 0,066 \approx 0,07 \approx 7\%$$

$$b) \text{ lecture: } \begin{array}{l} 1000 \text{ cartes / mn} \\ 300 \text{ cartes} \rightarrow 0,3 \text{ mn} \end{array}$$

$$\text{écriture: } \begin{array}{l} 1000 \text{ lignes} \rightarrow 1 \text{ mn} \\ 500 \text{ lignes} \rightarrow 0,5 \text{ mn} \end{array}$$

$$r = \frac{1}{0,3 + 1 + 0,5} = \frac{1}{1,8} = 0,55 \approx 55\% \quad ; \quad D = \frac{60}{1,8} = 33$$

$$2^{\circ}) a) r = \frac{50 \times 1}{50 + 5(\text{montage})} = 90\% \quad ;$$

un batch de 50 jobs nécessite 5 mn + 50 mn \rightarrow 55 mn, pendant les 5 mn restantes (60-55) on fait encore exécuter 5 jobs d'où $D = 50 + 5 = 55$

b) Un batch nécessite:

$$5 + 10 + 50 \times 0,3 + 50 + 50 \times 0,5 + 10 = 120 \text{ mn}$$

Le temps d'attente moyen d'un usager est: $120 \text{ mn} = 2 \text{ h}$

3°)

$$\text{Pour le 1}^{\text{er}} \text{ job} \rightarrow r = \frac{1}{0,3 + 1} = 77\%$$

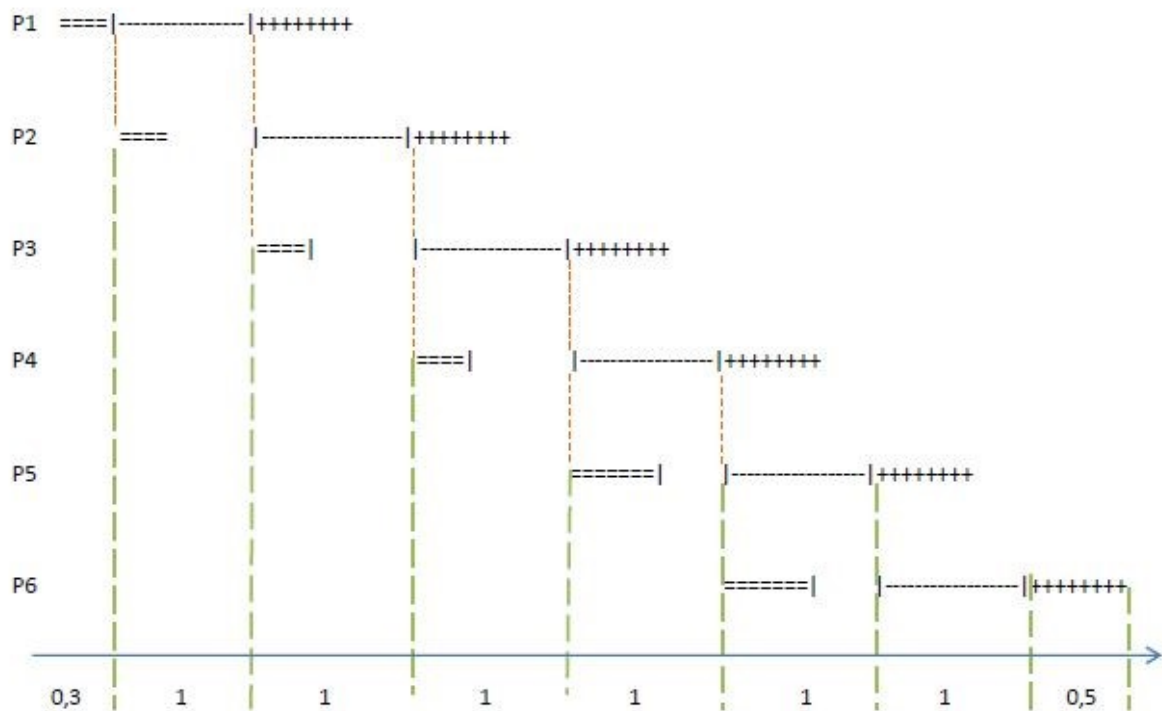
$$\text{Pour les } (n-1) \text{ jobs suivants: } r = 100\%$$

$$\left. \begin{array}{l} \text{le 1}^{\text{er}} \text{ job} \rightarrow 1,3 \text{ mn} \\ \text{le } n^{\text{me}} \text{ job} \rightarrow 1,5 \text{ mn} \end{array} \right\} \begin{array}{l} 2,8 \text{ mn} \rightarrow 2 \text{ jobs} \\ 60 - 2,8 = 57,2 \rightarrow 57 \text{ jobs} \end{array} \rightarrow D = 59 \text{ jobs}$$

Question 3.a

Soit à déterminer le nombre n de jobs que l'on peut faire passer en une heure.

===== signifie lecture, ---- signifie traitement, ++++++++ signifie impression.



Calcul de D :

$0,3 + 1 \cdot n + 0,5 = 60 \text{ mn} \rightarrow n = 60 - 0,5 - 0,3 \approx 59$, donc durant 1 heure on peut faire passer 59 jobs.

Calcul de N :

Pour le 1^{er} job, le rendement du CPU est : $N1 = 1/0,3 = 77\%$.

Pour les $n-1$ jobs suivants, le rendement du CPU est : $N2 = 1/1 = 100\%$.

Question 3.b

Temps de lecture de 1000 cartes = $1200 \times 1 \text{ mn} / 1000 = 1,2 \text{ mn}$

Temps d'impression de 1000 lignes = $1500 \times 1 \text{ mn} / 1000 = 1,5 \text{ mn}$

A l'inverse du cas précédent (question 3.a), on constate que le temps de traitement (CPU) est inférieur au temps de lecture et d'impression.

===== signifie lecture, ---- signifie traitement, ++++++ signifie impression.

$a = 1,2 - 1 = 0,2$ est temps d'attente du CPU.

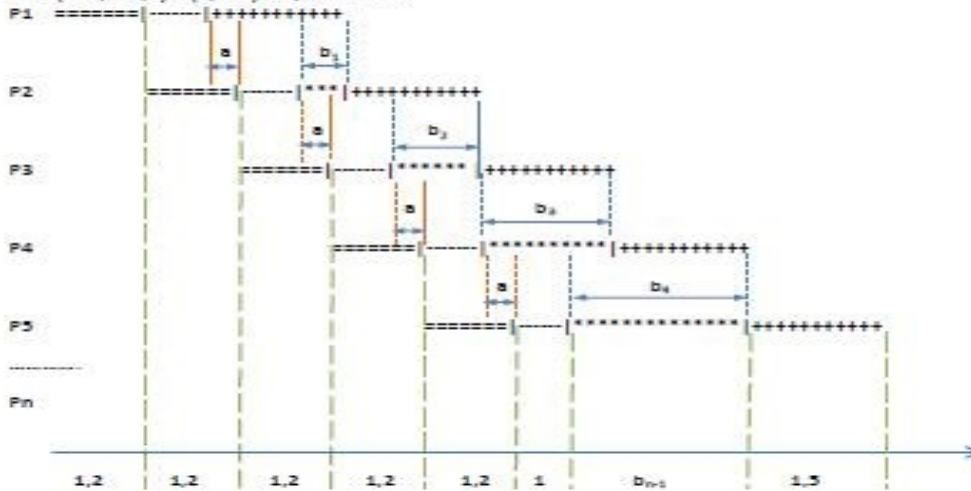
b_1, b_2, b_3, \dots : Temps d'attente du job pour passer à l'impression car l'imprimante est occupée par l'impression du job précédent.

$$b_1 = (1 + 1,5) - (1,2 + 1) = 0,3$$

$$b_2 = (1 + 0,3 + 1,5) - (1,2 + 1) = 0,6$$

$$b_3 = (1 + 0,6 + 1,5) - (1,2 + 1) = 0,9$$

$$b_4 = (1 + 0,9 + 1,5) - (1,2 + 1) = 1,2$$



$$b_1 = 0,3 = 0,3 \times 1; b_2 = 0,6 = 0,3 \times 2; b_3 = 0,9 = 0,3 \times 3; b_4 = 1,2 = 0,3 \times 4; b_5 = 1,5 = 0,3 \times 5$$

$$b_{n-1} = 0,3 \times (n-1).$$

Calcul de D :

$$1,2 \times n + 1 + b_{n-1} + 1,5 = 60 \text{ mn}$$

$$1,2 \times n + 1 + 0,3 \times (n-1) + 1,5 = 60 \text{ mn}$$

$$1,5 \times n + 1,5 = 60 \rightarrow n = (60 - 1,5) / 1,5 = 39, \text{ durant une heure on peut exécuter 39 jobs.}$$

Calcul de N :

$$N = 39 \times 1 \text{ mn} / 60 \text{ mn} = 0,65 = 65\%$$