**Exercice 4 : Résolution d’un problème d’ordonnancement**

Un système d’exploitation installé sur une machine disposant d’un seul processus doit gérer 5 processus A, B, C, D et E. Comme illustré dans la figure ci-dessus, chaque processus est décrit par le nombre d’instructions qu’il contient, son temps de lancement, son PID et l’UID de l’utilisateur qui l’a lancé :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Processus | PID | Instructions | Temps de lancement | UID |
| A | 1 | 3 | 1 | 1 |
| B | 2 | 6 | 2 | 2 |
| C | 3 | 4 | 3 | 2 |
| D | 4 | 2 | 4 | 1 |
| E | 5 | 3 | 6 | 3 |

Le processeur de la machine peut traiter l’instruction par cycle, les temps de lancement étant, pour des raisons de simplicité, exprimés eux aussi en cycle de processeurs (un temps de lancement de 2 indique que le programme a été lancé au début du deuxième cycle de processeur).

Une table d’ordonnancement peut être écrite pour un ensemble de processus sous la forme du tableau.

|  |  |  |  |
| --- | --- | --- | --- |
| Cycle | Processus en attente | Etat de fin de cycle | Nombre d’instructions restantes |

Ou la colonne cycle contient le numéro du cycle processeur (le premier étant 1), la colonne Processus en attente contient le processus présent dans le processeur durant le cycle et Etat de fin ce cycle contient l’état du processus à la fin du cycle. Cet état peut être : en attente, en exécution ou arrêté.

1. Ordonnancement Round Robin (Tourniquet)

En utilisant un ordonnanceur de type Round Robin :

* 1. Écrire la table d’ordonnancement du système pour traiter l’ensemble des processus.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cycle | Processus en attente | Etat de fin de cycle | Nombre d’instructions restantes | Liste d’attente durant le cycle |
| 1 | A | En attente | 2 | **B** |
| 2 | B | En attente | 5 | **A** |
| 3 | C | En attente | 3 | **A,B** |
| 4 | D | En attente | 1 | **A,B,C** |
| 5 | A | En attente | 1 | **B,C,D** |
| 6 | E | En attente | 2 | **B,C,D,A** |
| 7 | B | En attente | 4 | **C,D,A,E** |
| 8 | C | En attente | 2 | **D,A,E,B** |
| 9 | D | Arrêté | Terminé | **A,E,B,C** |
| 10 | A | Arrêté | Terminé | **E,B,C** |
| 11 | E | En attente | 1 | **B,C** |
| 12 | B | En attente | 3 | **C,E** |
| 13 | C | En attente | 1 | **E,B** |
| 14 | E | Arrêté | Terminé | **B,C** |
| 15 | B | En attente | 2 | **C** |
| 16 | C | Arrêté | Terminé | **B** |
| 17 | B | En attente | 1 | **B** |
| 18 | B | Arrêté | Terminé |  |

1. Calculer pour chaque processus son temps d’attente (nombre de cycle avant la première exécution du processus) et son temps d’exécution total (nombre de cycles entre l’exécution de sa dernière instruction et son lancement).

|  |  |  |
| --- | --- | --- |
| Processus | Temps d’attente | Temps d’exécution |
| A | 0 | 9 |
| B | 0 | 16 |
| C | 0 | 13 |
| D | 0 | 5 |
| E | 0 | 8 |

1. Ordonnancement avec priorité

Nous utilisons maintenant un ordonnanceur basé sur des priorités qui fonctionne de la manière suivante :

∙ Chaque processus possède une priorité sous forme de nombre entier entre 0 et 10

∙ La priorité maximale d’un processus et 0, la minimale est 10

∙ Au lancement, un processus à une priorité de 0

∙ A chaque cycle d’exécution, la priorité du processus augmente de 1

∙ A chaque cycle, l’ordonnanceur exécute le processus le plus prioritaire

∙ A chaque cycle, la priorité des processus non exécuté diminue de 1 (sans pouvoir passer sous 0)

∙ Chaque fois que 2 processus on même priorité, celui qui au moins d’instructions est choisi en utilisant cet ordonnanceur :

1. Écrire la table d’ordonnancement du système pour traiter l’ensemble des processus (ajouter à la table une colonne priorité qui contient la priorité du processus à la fin du cycle).\*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cycle | Processus en attente | Etat de fin de cycle | Nombre d’instructions restantes | Priorité de fin de cycle | Liste d’attente durant le cycle |
| 1 | A | En attente | 2 | 1 | B(0) |
| 2 | B | En attente | 5 | 1 | A(0), |
| 3 | C | En attente | 3 | 1 | A(0), B(0) |
| 4 | D | En attente | 1 | 1 | A(0),B(0),C(1) |
| 5 | A | En attente | 1 | 1 | B(0),C(0),D(1) |
| 6 | E | En attente | 2 | 1 | B(0),C(0),D(0),A(1) |
| 7 | D | Arrêté | Terminé | \_ | B(0),C(0),A(0),E(1) |
| 8 | A | Arrêté | Terminé | \_ | B(0),C(0),E(0) |
| 9 | E | En attente | 1 | 1 | B(0),C(0) |
| 10 | C | En attente | 2 | 1 | B(0),E(1) |
| 11 | E | Arrêté | Terminé | \_ | B(0),C(1) |
| 12 | C | En attente | 1 | 1 | B(0) |
| 13 | B | En attente | 4 | 1 | C(0) |
| 14 | C | Arrêté | Terminé | \_ | B(0) |
| 15 | B | En attente | 3 | 1 | \_ |
| 16 | B | En attente | 2 | 2 | \_ |
| 17 | B | En attente | 1 | 3 | \_ |
| 18 | B | Arrêté | Terminé | \_ | \_ |

|  |  |  |
| --- | --- | --- |
| Processus | Temps d’attente | Temps d’exécution |
| A | 0 | 7 |
| B | 0 | 16 |
| C | 0 | 11 |
| D | 0 | 3 |
| E | 0 | 5 |