|  |  |  |
| --- | --- | --- |
| **Université Mohamed Khider Biskra**  **Faculté des sciences exactes et des sciences de la nature et de la vie.**  **Département d’Informatique** | | |
| **Niveau** : 2LMD | **Date** : Janvier 2019 | **Module** : Systèmes d‘exploitation 1 |
| **Série 1 : Introduction aux systèmes d’exploitation (Correction)** | | |

**Exercice 1 :**

1. **Un programme :** est un ensemble d’instructions bien organisées écrit dans un langage de programmation qui a un objectif bien défini.

**Un processus :** est une entité dynamique qui résulte de l’exécution d’un programme.

**Un langage de programmation** : destinée à formuler des [algorithmes](https://fr.wikipedia.org/wiki/Algorithme) et produire des [programmes informatiques](https://fr.wikipedia.org/wiki/Programme_informatique) qui les appliquent. D'une manière similaire à une langue naturelle, un langage de programmation est composé d'un [alphabet](https://fr.wikipedia.org/wiki/Alphabet), d'un [vocabulaire](https://fr.wikipedia.org/wiki/Vocabulaire), de règles de [grammaire](https://fr.wikipedia.org/wiki/Grammaire) et de [significations](https://fr.wikipedia.org/wiki/Sens_(linguistique))

**Un environnement de programmation** : est un ensemble d'outils qui permet d'augmenter la productivité des programmeurs qui [développent des logiciels](https://fr.wikipedia.org/wiki/D%C3%A9veloppement_logiciel). Il comporte un [éditeur de texte](https://fr.wikipedia.org/wiki/%C3%89diteur_de_texte) destiné à la programmation, un [compilateur](https://fr.wikipedia.org/wiki/Compilateur) et l'[éditeur de liens](https://fr.wikipedia.org/wiki/%C3%89diteur_de_liens)…etc

2**. La monoprogrammation :** ne permet qu'à un seul processus utilisateur d'être en mémoire centrale à un instant donné, la gestion de la mémoire est énormément simplifiée.

**La multiprogrammation :** le système d’exploitation maintient plusieurs travaux en mémoire à la fois.

3. Les étapes de développement d’un programme :

## La conception : L'analyse et la traduction dans un langage de programmation

## La production du programme :

### 1. L'édition du code source

### 2. La traduction en langage machine :

### a.L'interprétation

### b . La compilation

L'analyse lexicographique

L'analyse syntaxique

 L'analyse sémantique

La génération du code et son optimisation

### 3. L'édition de liens

4 Le chargement

5 Le Debug

**Exercice 2 :**

1. Ensemble de programmes permettant de gérer et contrôler les ressources de l’ordinateur de façon à constituer une interface entre l’utilisateur et la machine physique.



gestion des ressources

fournir une machine virtuelle à l’utilisateur

3.

gestion des processus

gestion de la mémoire

gestion de fichiers

gestion des entrées/sorties

protection et sécurité des données

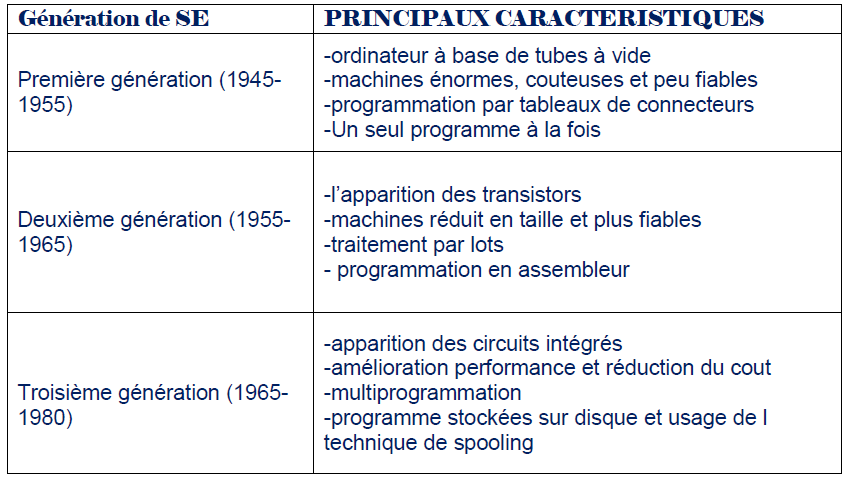
4.

noyau

interpréteur de commande

système de fichiers

4.



**Quatrième génération 1980 jusqu ‘à présent :** l'amélioration constante du taux d'intégration des circuits (LSI : Large Scale Integration, VLSI, Very LSI), développement des réseaux locaux, l'explosion d'internet. Les interfaces Homme/Machine sont toujours améliorées….

Vous pouvez parlez sur l’évolution des systèmes d’exploitation aussi : DOS,Windows avec ses différentes versions, UNIX, LINUX……

**Exercice 3 :**

**Question A**

Notons que le temps de lecture de 300 cartes est de 0.3 mn, et le temps d'impression de 500 lignes est de 0.5 mn.

**Question A.1**

Le temps pour faire un passage est donc de 0.3 + 1 + 0.5 = 1.8 minutes. Comme entre deux passages l'usager a besoin de 4 mn pour corriger, le nombre de passages pour 15 minutes est au plus n tel que 1.8 \* n + 4 \* (n - 1) ≤ 15. En prenant n = 3, la durée de sa session sera de 13.4 mn. Il s'ensuit que h = 3  15 = 0.2, et *D* = 3 \* 4 = 12

**Question A.2**

Lorsque le système est exploité avec un moniteur d'enchaînement des travaux, le temps de passage est le même, mais il n'y a pas d'attente entre deux passages. Il s'ensuit que *D* = 60  1.8 = 33, et h = 33  60 = 0.55.

**Question B**

**Question B.1**

Le débit maximum est limité à 60 travaux à l'heure, par l'unité de traitement. Le débit maximum d'entrée des cartes, est de 60  0.3 = 200 travaux à l'heure. Le débit maximum d'impression est de 60  0.5 = 120 travaux à l'heure. D'où *D* = 60, et h = 60  60 = 1.0. Ces valeurs ne pourront en fait être atteintes que si la planification est correcte.

**Question B.2**

La planification doit tenir compte du fait que l'opérateur ne peut faire qu'une chose à la fois, comme d'ailleurs l'ordinateur d'entrées sorties. Par ailleurs elle doit respecter l'ordre suivant pour un train:

1 préparation du train de cartes 10 minutes

2 lecture de 50 travaux. 15 minutes

3 transfert de la bande vers l'ordinateur central, 5 minutes

4 exécution de ces 50 travaux, 50 minutes

5 transfert de la bande d'impression 5 minutes

6 impression des 50 travaux 25 minutes

7 distribution des résultats 10 minutes

L'opération 4 du train n est effectuée en même temps que les opérations suivantes, dans l'ordre:

5 étape n - 1

6 étape n - 1 et 1 de l'étape n + 1

7 étape n - 1 et 2 étape n + 1

1. étape n + 1

http://deptinfo.cnam.fr/Enseignement/CycleA/AMSI/exercices_systemes/introd01.gif

Il s'ensuit que le temps d'attente moyen est:

10 + 15 + 5 + 50 + 5 + 25 + 10 = 120 soit 2 heures.

**Question C**

La lecture du travail n + 1 demande 0.3 mn, ce qui est inférieur au temps de traitement, qui est de 1 mn. L'impression du travail n - 1 demande 0.5 mn, ce qui est encore inférieur au temps de traitement. D'où *D* = 60, et h = 1.0.

Si on a 1200 cartes à lire, le temps de lecture du travail n + 1 devient de 1.2mn, ce qui est cette fois supérieur au temps de traitement. De même, si on a 1500 lignes, le temps d'écriture du travail n - 1 devient 1.5mn, ce qui est également supérieur au temps de traitement. Le plus grand des 3 est le temps d'impression. On a donc *D* = 60  1.5 = 40, et h = 40  60 = 0.67.