

Ce contrôle comporte 5 questions; le nombre maximal possible de points est de 20. Les réponses sont à porter sur une copie comportant votre nom. Il n'est pas nécessaire de répondre aux questions dans l'ordre — commencez par celles où vous vous sentez le plus à l'aise.

Les calculatrices ne sont pas autorisées.

1. Questions de cours

- (a) (3 points) QCM – les réponses sont à porter sur votre copie; il n'est pas nécessaire de justifier vos réponses; une seule réponse par question.

- i. Quel est le rôle principal d'un système d'exploitation?
- A. Compiler les programmes.
  - B. Gérer les ressources matérielles et logicielles de l'ordinateur.
  - C. Créer des documents.
  - D. Réaliser des calculs mathématiques.

**Solution:** B. Gérer les ressources matérielles et logicielles de l'ordinateur.

- ii. Parmi les éléments suivants lequel n'est pas une tâche du système d'exploitation?
- A. Gestion de la mémoire.
  - B. Gestion des fichiers
  - C. Traitement de texte.
  - D. Gestion des périphériques.

**Solution:** C. Traitement de texte.

- iii. Quel est le rôle d'un routeur dans un réseau informatique?
- A. Connecter des ordinateurs à un réseau local.
  - B. Traduire les adresses web en adresses IP.
  - C. Acheminer les paquets de données entre différents réseaux.
  - D. Stocker les données des utilisateurs.

**Solution:** C. Acheminer les paquets de données entre différents réseaux.

- iv. Quelle composante de l'architecture de von Neumann est responsable de l'exécution des instructions?
- A. Unité de contrôle.
  - B. Mémoire.
  - C. Unité de stockage.
  - D. Périphériques d'entrée/sortie.

**Solution:** A. Unité de contrôle.

- v. Le langage assembleur est un type de:
- A. Langage de haut niveau.
  - B. Langage de programmation visuel.
  - C. Langage de bas niveau.
  - D. Langage de script.

**Solution:** C. Langage de bas niveau.

- vi. Quelle est la différence principale entre le langage machine et le langage assembleur ?

- A. Le langage machine utilise des instructions compréhensibles par les humains.
- B. Le langage assembleur utilise des instructions symboliques compréhensibles par les humains.
- C. Le langage assembleur est plus rapide à exécuter que le langage machine.
- D. Le langage machine est plus facile à lire et à écrire que le langage assembleur.

**Solution:** B. Le langage assembleur utilise des instructions symboliques compréhensibles par les humains.

- (b) (1 point) Lorsqu'on gère les droits qu'un utilisateur a sur un fichier dans un environnement Linux, on utilise les lettres "r", "w", et "x". Expliquez brièvement à quoi chacune correspond.

**Solution:**

- r (read) : Permet la lecture du fichier.
- w (write) : Permet la modification du fichier.
- x (execute) : Permet l'exécution du fichier s'il s'agit d'un programme ou d'un script.

- (c) (1 point) Dans le contexte des réseaux informatiques, qu'est-ce qu'un protocole? À quoi cela sert-il?

**Solution:** Tout ou partie des éléments suivants était attendu: un protocole est un ensemble de règles et de conventions définissant le format, la séquence et la gestion des erreurs des échanges de données sur un réseau. Il sert à assurer une communication cohérente et efficace entre les périphériques sur un réseau.

- (d) (1 point) Qu'entend-on par l'expression "passerelle par défaut"? Que désigne-t-elle et à quoi sert-elle?

**Solution:** Tout ou partie des éléments suivants était attendu: la passerelle par défaut est l'adresse IP du routeur utilisée par un périphérique pour envoyer des paquets de données à des destinations hors de son réseau local. Elle sert de point de sortie pour les données à destination de réseaux distants — internet notamment.

- (e) (1 point) Listez deux différences entre une adresse MAC et une adresse IP.

**Solution:** Deux parmi les différences suivantes étaient attendues:

- Une adresse MAC est une adresse matérielle associée à une carte réseau, tandis qu'une adresse IP est une adresse logique attribuée à un périphérique sur un réseau.
- Les adresses MAC sont assignées par le fabricant de la carte réseau et sont généralement permanentes, tandis que les adresses IP sont attribuées dynamiquement par un protocole réseau et peuvent changer.
- Les adresses MAC fonctionnent au niveau de la couche réseau du modèle TCP/IP tandis que les adresses IP fonctionnent au niveau de la couche internet.
- Les adresses MAC sont codées sur 48 bits (6 octets), tandis que les adresses IPv4 sont codées sur 32 bits (4 octets) et les adresses IPv6 sur 128 bits (16 octets).
- Une adresse MAC est unique tandis qu'une adresse IP peut exister simultanément à l'intérieur de différents réseaux.

- (f) (1 point) Pourquoi utilise-t-on des registres en assembleur alors qu'on n'en parle pas en Python?

**Solution:** Les registres (espaces mémoire à l'intérieur du processeur, d'accès extrêmement rapide) en assembleur sont utilisés pour stocker des données temporaires et pour effectuer des opérations arithmétiques et logiques. Ils sont essentiels pour manipuler efficacement les données directement sur

0,5 si non de "proche machine".



le processeur. En Python, les registres ne sont pas directement accessibles car le langage est de haut niveau et délègue donc la gestion de la mémoire au système d'exploitation.

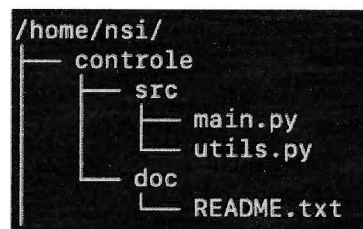
## 2. Commandes shell

Je me trouve dans le répertoire `/home/nsi` et j'exécute les commandes suivantes:

```
1 mkdir controle
2 cd controle
3 mkdir src
4 mkdir doc
5 cd src
6 touch main.py
7 touch utils.py
8 cd ../doc
9 touch README.txt
10 cd ../../..
11 pwd
```

- (a) (1 point) Dessinez l'arborescence créée par ces commandes.

**Solution:** Une manière possible de représenter cette arborescence était la suivante:



- (b) (1 point) Quel sera le retour de la dernière commande (qu'est-ce qui va s'afficher dans la console à l'exécution de `"pwd"`)?

**Solution:** On est, depuis le point de départ `/home/nsi/` "descendus" deux fois dans les lignes précédentes (d'abord dans `controle` puis dans `doc`) puis, dans la ligne juste au-dessus, on est "remontés" de trois niveaux (`"../../.."`). On est donc un cran au-dessus de notre point de départ et le retour du `pwd` sera:

`/home`

*0,5 si /home/nsi*

- (c) (1 point) En une, deux, ou trois commandes successives à la suite des précédentes, faites afficher dans la console le contenu du fichier `README.txt`.

**Solution:** En deux commandes (je vous laisse deviner ce que cela donne en une ou en trois), cela donne:

```
1 cd controle/doc
2 cat README.txt
```

*0,5 si cd mais pas cat.*

- (d) (1 point) On souhaite que le fichier `utils.py` soit: lisible, modifiable, et exécutable uniquement par le propriétaire — et qu'il n'y ait aucun droit pour les autres. Quelle commande `chmod` doit-on utiliser pour cela?

**Solution:** On veut tous les droits pour le propriétaire, donc `r`, `w` et `x` à 1, donc au total:  $(111)_2 = 7_{10}$ ; on ne veut aucun droit pour les autres, donc 0 dans les deux cas:

`chmod 700 /home/nsi/controle/src/utils.py`

*0,5 sur 1 si on a écrit un nombre mais faux les 2 fois.*

- (e) (1 point) On souhaite que le fichier `main.py` soit: exécutable uniquement par le propriétaire; modifiable par le propriétaire et par le groupe; et lisible par tout le monde (propriétaire, groupe, et autres). Quelle commande `chmod` doit-on utiliser pour cela?

**Solution:** On veut tous les droits pour le propriétaire, donc 7 (voir ci-dessus); pour le groupe on veut les droits `r` et `w` à 1, donc  $(110)_2 = 6_{10}$ ; et pour les autres on ne veut que `r` à 1, donc  $(100)_2 = 4_{10}$ . Donc:

```
chmod 764 /home/nsi/contrôle/src/main.py
```

### 3. (2 points) Masques de sous-réseau

Soit l'adresse IP et masque de sous-réseau (en notation CIDR) `192.168.1.50/26`. Combien d'adresses IP distinctes peuvent-elles exister dans ce réseau? Quelle est la première (avec la valeur la plus faible)? Quelle est la dernière?

**Solution:** En notation CIDR `192.168.1.50/26` signifie que les 26 premiers bits de l'adresse IP correspondent à l'adresse réseau. Étant donné qu'une adresse IP est codée sur 4 octets, donc 32 bits, cela signifie qu'il reste 6 bits pour l'adresse de l'hôte, pour un total de  $2^6 = 64$  adresses possibles.

Pour déterminer la Appliquons d'abord le "ET" du masque de sous-réseau pour déterminer l'adresse exacte du réseau. Le masque de sous réseau est constitué des trois premiers octets tous à 1 (24 bits) puis de deux bits à 1 et enfin des six derniers à 0. L'adresse IP, elle, est donc `192.168.1` puis 50 que nous convertissons en binaire:  $50 = 32 + 16 + 2 = 2^5 + 2^4 + 2^1 = (110010)_2$

*1 si raisonnement, calcul OK mais résultat faux.*

	192.168.1.000110010
ET	255.255.255.11000000
	192.168.1.00000000

*0,5 si "26 bits".*

*1,5 si 63 ou 64*

*mais 1ère et dernière*

La première adresse est ainsi celle qui a les 6 derniers bits à 0 et la dernière celle qui a les 6 derniers bits à 1, soit, une fois converti en décimal: `192.168.1.0` et `192.168.1.63` (pour un total qui fait donc bien les 64 que l'on a calculés plus haut).

Remarque: strictement parlant, ces deux adresses sont, respectivement, l'adresse du réseau lui-même et l'adresse dite "de broadcast", c'est-à-dire l'adresse utilisée pour envoyer simultanément un message à tous les périphériques connectés à ce réseau. Donc les adresses disponibles pour de "vrais" périphériques sont en fait celles allant de `192.168.1.1` à `192.168.1.62`.

### 4. (2 points) Masques de sous-réseau — suite

Soit l'adresse IP `192.168.10.5` et le masque de sous-réseau `255.255.255.240`. Parmi les adresses IP suivantes, lesquelles appartiennent au même réseau que `192.168.10.5`? [ATTENTION: il est demandé le détail du calcul pour chacune des adresses – une réponse sans justification ne rapportera aucun point]

- `192.168.10.9`
- `192.168.10.14`
- `192.168.10.18`
- `192.168.10.3`

**Solution:** Le masque de sous réseau étant constitué de trois 255 pour les trois premiers octets, toutes les adresses du réseau commenceront bien par `192.168.10`. Pour ce qui est du dernier octet on va appliquer à chacune des adresses listées le "ET" du masque de sous-réseau pour pouvoir conclure. On commence par convertir 240 en binaire:  $240 = 255 - 15 = 255 - 1 - 2 - 4 - 8 = 255 - 2^0 - 2^1 - 2^2 - 2^3 = (11110000)_2$ .

On applique le masque de sous-réseau aux 5 adresses (celle de base et les quatre de l'énoncé). On commence par convertir en binaire les valeurs du dernier octet de chacune d'entre elles:

- $5 = 4 + 1 = 2^2 + 2^0 = (101)_2$
- $9 = 8 + 1 = 2^3 + 2^0 = (1001)_2$
- $14 = 8 + 4 + 2 = 2^3 + 2^2 + 2^1 = (1110)_2$
- $18 = 16 + 2 = 2^4 + 2^1 = (10010)_2$
- $3 = 2 + 1 = 2^1 + 2^0 = (11)_2$

*1 si ET faux.*

$$\begin{array}{r} 5: \\ 00000101 \\ \text{ET } 11110000 \\ \hline 00000000 \end{array}$$

$$\begin{array}{r} 9: \\ 00001001 \\ \text{ET } 11110000 \\ \hline 00000000 \end{array}$$

$$\begin{array}{r} 14: \\ 00001110 \\ \text{ET } 11110000 \\ \hline 00000000 \end{array}$$

$$\begin{array}{r} 18: \\ 00010010 \\ \text{ET } 11110000 \\ \hline 00010000 \end{array}$$

$$\begin{array}{r} 3: \\ 00000011 \\ \text{ET } 11110000 \\ \hline 00000000 \end{array}$$

On constate immédiatement que tous les résultats sont à 0 à l'exception de l'avant-dernier qui est à  $00010000_2$ . On peut donc en conclure que toutes les adresses listées font partie du même réseau à l'exception de 192.168.10.18.

5. (3 points) Consultation d'une page web

Lorsqu'un serveur web envoie des données à un navigateur, les données sont encapsulées dans des segments UDP à la couche transport, utilisant le port 80. Ces segments sont ensuite encapsulés dans des paquets TCP (couche internet), qui sont à leur tour encapsulés dans des trames Wi-Fi à la couche réseau. Chaque trame contient une adresse IP unique pour identifier les appareils. Le protocole TCP assure un transfert rapide des données sans vérifier leur intégrité. Finalement, les trames sont transmises sur le réseau via des ondes radio.

Le texte ci-dessus contient 6 erreurs: identifiez-en (au moins) 3 et donnez-en la correction. (Si vous en identifiez plus, cela vous vaudra des points bonus - 0.5 point par erreur supplémentaire.)

**Solution:** Voici le texte, ré-écrit avec en gras les 6 erreurs corrigées:

Lorsqu'un serveur web envoie des données à un navigateur, les données sont encapsulées dans des segments **TCP** à la couche transport, utilisant le port 80. Ces segments sont ensuite encapsulés dans des paquets **IP** (couche Internet), qui sont à leur tour encapsulés dans des trames **Ethernet** à la couche réseau. Chaque trame contient une adresse **MAC** unique pour identifier les appareils. Le protocole TCP assure un transfert **fiable des données en vérifiant leur intégrité**. Finalement, les trames sont transmises sur le réseau via, principalement, **des câbles Ethernet**.