

## ÉPREUVE REGROUPEE

Discipline: INFORMATIQUE

Date: 5 juin 2024

Cours: 1 IN DP 06/07/08/09/10

Durée: 105 minutes

Maître: JC Barros

Matériel autorisé:

- Aucun (aide-mémoire en fin d'épreuve)

Consignes de travail:

- Ecriture lisible et propre demandée
- Mettre en évidence les résultats

Nom et prénom:

Total: 68 points

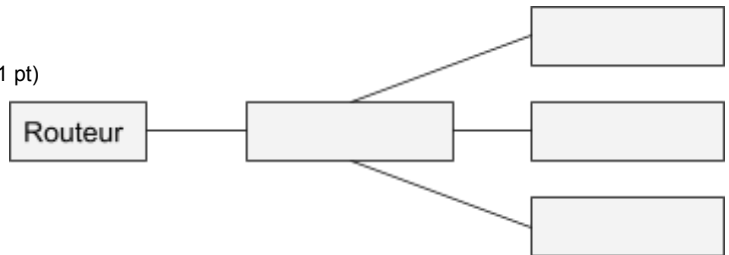
### RÉSEAU

1. / 3 pts

Le schéma ci-contre représente un réseau local.

a. Quelle est sa topologie ? (1 pt)

b. Expliquer à quoi sert un routeur? (1 pt)



c. Écrire dans le rectangle concerné le nom de l'unique appareil (hormis le routeur) dont on est sûr de l'identité. (1 pt)

2. / 5 pts

Entourer chaque bonne réponse

a. Pour envoyer un e-mail à Mamie, il y a seulement deux appareils qui sont sollicités sur le réseau : mon ordinateur et le sien.

Vrai / Faux

b. Pour envoyer un e-mail à Mamie, il n'y a qu'un seul programme sollicité sur le réseau.

Vrai / Faux

c. Pour envoyer un e-mail à Mamie, les données peuvent circuler sous la mer ou dans l'air.

Vrai / Faux

d. Pour envoyer un e-mail à Mamie, on peut écrire son adresse mail dans l'URL.

Vrai / Faux

e. Pour envoyer un e-mail à Mamie, un seul chemin est toujours utilisé pour acheminer ma lettre sur le réseau comme un envoi postal.

Vrai / Faux

3.

/ 2 pts

Citer deux protocoles utilisés dans la couche **application** du modèle OSI

a.

b.

4.

/ 3 pts

Expliquer la fonction de DNS en quelques lignes (donner 3 informations pertinentes).

5.

/ 1 pt

Quel est le nom du langage principalement utilisé pour écrire des pages web ?  
*voir l'exemple ci-dessous*

```
<!DOCTYPE html PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html>
  <head>
    <title>
      Exemple de HTML
    </title>
  </head>
  <body>
    Ceci est une phrase avec un <a href="cible.html">hyperlien</a>.
  <p>
    Ceci est un paragraphe où il n'y a pas d'hyperlien.
  </p>
</body>
</html>
```

6.

/ 4 pts

a. Citer deux différences entre les protocoles TCP et UDP. (2 pts)

i.

ii.

b. Relier les exemples d'applications à chacun des protocoles correspondants. (1 pt)

streaming (Youtube, Spotify, Netflix, etc.) • • UDP

web, email, transfert de fichiers • • TCP

c. Dans quelle couche du modèle OSI ces protocoles sont-ils utilisés? (1 pt)

7.

/ 2 pts

Quelle est la différence entre **internet** et **web**?

*Donner au moins deux notions importantes*

8.

/ 4 pts

Relier chacune des adresses suivantes avec son type:

<https://fr.wikipedia.org/wiki/Informatique> •

• URL

A7:BF:21:2C:FE:14 •

• Adresse (IP)

10.23.567.18 •

• aucune...

192.168.1.202 •

• Adresse (MAC)

9.

/ 3 pts

**Relier les notions correspondantes**

|                 |   |   |                                                                                                      |
|-----------------|---|---|------------------------------------------------------------------------------------------------------|
| Intégrité       | • | • | Maintenir le bon fonctionnement du système d'information                                             |
| Disponibilité   | • | • | Rendre les données inutilisables pour les personnes qui ne sont pas les destinataires de ces données |
| Confidentialité | • | • | Garantir que les données ne sont pas modifiées pendant le transport                                  |

10.

/ 7 pts

**Vocabulaire lié à la sécurité : relier les notions correspondantes**

|                                                                                                                                                                 |   |   |                   |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---|-------------------|
| Attaque impliquant un chantage en échange de la récupération de données ou d'un service.                                                                        | • | • | Malware           |
| Récupération de données personnelles (mot de passe ou autre) par un service se faisant passer pour un autre (votre banque, la poste, un magasin en ligne, etc.) | • | • | Spyware           |
| Méthode pour décrypter un message ou trouver un mot de passe en testant toutes les possibilités une par une.                                                    | • | • | Man in the middle |
| Pirate informatique                                                                                                                                             | • | • | Force brute       |
| Terme générique désignant une application malveillante                                                                                                          | • | • | Phishing          |
| Application destinée à intercepter des données confidentielles                                                                                                  | • | • | Ransomware        |
| Interception des communications entre deux parties, sans que celles-ci s'en rendent compte                                                                      | • | • | Hacker            |

11.

/ 10 pts

Pour chacun des programmes ci-dessous, indiquer tout ce qu'il affiche à l'écran (rien de plus, rien de moins) et la valeur de la variable **k** à la fin de son exécution.

| PROGRAMMES                                                                                                                                                          | Affichage à l'écran | Valeur finale de k |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|--------------------|
| <pre># EXEMPLE k = 10 print(k) k = k + 1 print(k)</pre>                                                                                                             | <pre>10 11</pre>    | 11                 |
| <pre># Programme 1 for k in range(3):     print(k * 2) print('Bye!')</pre>                                                                                          |                     |                    |
| <pre># Programme 2 for k in ['Y', 'e', 's']:     print(k * 2)</pre>                                                                                                 |                     |                    |
| <pre># Programme 3 k = 3 while k &gt; 0:     print(k)     k = k - 1</pre>                                                                                           |                     |                    |
| <pre># Programme 4 k = 3 while k &gt; 0:     k = k - 1     print(k)</pre>                                                                                           |                     |                    |
| <pre># Programme 5 k = 10 if k * 3 &lt;= 0:     print(1) elif k + 2 &gt;= 0:     print(2) elif k ** 2 &gt; 0:     print(3) else:     print(4)     print('OK')</pre> |                     |                    |

12.

/ 5 pts

Indiquer au bout de chaque flèche ce que les lignes concernées affichent à l'écran.

```

1  S = [ 'a', 'b' , 'b' , 'a' ]
2
3  print( S[2]+S[3] )
4
5  print(len(S))
6
7  S.append('m')
8
9  print(S)
10
11 print(S[1:3])
12
13 S.remove('a')
14 print(S)
15

```

13.

/ 5 pts

- a. Tracer le dessin correspondant au programme ci-contre. (2 pts)

*La tortue est dirigée vers la droite au départ.*

*Point de départ : X*

X

```

1  from turtle import *
2
3  forward(10)
4  left(90)
5  forward(10)
6  right(90)
7
8  forward(10)
9  left(90)
10 forward(10)
11 right(90)
12
13 forward(10)
14 left(90)
15 forward(10)
16 right(90)
17
18 forward(10)
19 left(90)
20 forward(10)
21 right(90)

```

- b. Écrire ci-dessous un programme qui donne le même dessin en moins de 10 lignes. (3 pts)

14.

/ 2 pts

Voici un programme qui ne fait pas ce qui est prévu (selon le commentaire à la ligne 1).

**Corriger** le programme pour qu'il affiche ce qui est prévu.

Réponse claire ci-dessous.

```
1 # affiche 21, 23, 25, 27
2 n = 20
3 for i in range(1, 5):
4     print(n + i)
```

<

Console x

```
>>> %Run -c $EDITOR_CONTENT
21
22
23
24
>>>
```

15.

/ 4 pts

Choisir **une seule** question entre **A** ou **B**.

**A.** Adresses IP et masque de réseau

Selon l'**adresse IP** suivante: **192.168.2.114**

et son **masque de réseau**: **255.255.255.0**

- a. Quelle est l'adresse IP du réseau ?
- b. Quelle est l'adresse de l'hôte ?
- c. A quoi correspond l'adresse hôte 255 ?
- d. Combien d'hôtes peuvent se connecter dans ce réseau au maximum?

**B.** Construire ci-dessous le logigramme de ce programme :

```
1 nb = 60
2
3 while nb > 5 :
4
5     if nb // 3 > 10 :
6         nb = nb//3
7
8     else :
9         nb = nb//2
10
11 print('Fin')
12
```

Voici un programme Python:

```

1  def mini(liste):                # mini est une .....
2      candidat = liste[0]
3      for element in liste:
4          if element < candidat:
5              candidat = element
6      return candidat
7
8
9  def trier(liste):
10     resultat = []               # resultat est une ..... vide
11     for i in range(len(liste)): # ici, la boucle se répète .... fois
12         m = mini(liste)         # m correspond au minimum dans liste
13         resultat.append(m)      # On ajoute ..... à .....
14         liste.remove(m)         # On ..... m de liste
15     return resultat
16
17
18 liste=[2, 5, 1, 9, 6, 0]        # liste contient des nombres désordonnés
19 nouv_liste = trier(liste)
20 print(nouv_liste)              # affiche la liste .....

```

- a) A quoi servent les symboles `#` ? (0,5 pt)
- b) Remplir (dans le programme) les zones pointillées avec le mot, la liste ou le nombre qui manque.  
7 réponses sont attendues. (3,5 pts)
- c) Que fait ce programme Python ? (1 pt)
- d) Proposer des modifications à ce programme pour qu'il affiche la liste dans l'ordre **décroissant**.  
*Indiquer clairement les modifications, par exemple en mentionnant avec chaque modification la ligne du programme concernée.* (3 pts)

# Programmation (Python)

## Calculs :

**opérations** de base: + - \* / \*\*

reste et quotient de division euclidienne: % //

var % div vaut 0 si var est divisible par div

## Variables et affectations :

**types:** int, float, str, bool, list ...

int(), float() pour **convertir** en int ou en float

**affectation** simple : ma\_variable = -5

**permutation** de variables: a, b = b, a

**affectations combinées:** +=, -=, \*=, etc.

ex: x += 1 est comme x = x + 1

## Listes :

**liste:** L = [-1, 3.2, "abc"]

**liste vide :** []

**multiplication:** [0]\*3 -> [0, 0, 0]

**élément** d'une liste: L[1] = 5

doit déjà exister, l'indice commence à 0.

**ajouter** un élément à une liste :

L.append(-2.5)

**supprimer** élément à la position i :

del L[i] ou L.pop(i)

**supprimer** élément de valeur v dans L :

L.remove(v)

**longueur** d'une liste L :

len(L)

**somme** des valeurs d'une liste:

sum(liste)

**slice:** L[i:j] (i inclus ! j exclu)

## Chaînes de caractères :

**chaîne:** txt="bonjour" ou txt='bonjour'

**concaténation:** txt + " est un texte"

**élément, slice** d'une chaîne: texte[i]

comme listes mais non modifiable

## Entrées/sorties :

**sortie:** print(valeur,... , end='\n')

valeur de type quelconque, end optionnel

**entrée:** texte = input("message")

Attention: retourne toujours un str

## Commentaires :

sur une ligne : # texte

plusieurs lignes : '''texte''' ou """texte"""

## Structures de contrôle :

1<sup>ère</sup> ligne finit par ':' puis indentation des lignes du bloc

tests simples: ==, !=, <, <=, >, >=, in

tests combinés: and, or, not, a < b < c

### conditionnelle

**if** condition1:

instructions

**elif** condition2:

# elif = contraction de else if

instructions

**else:**

instructions

**boucle while** (boucle non bornée) :

**while** condition:

instructions

**boucle for** (boucle bornée) :

**for** i in liste: # boucle parcourt la liste

instructions

**for** i in range(5): # pour i de 0 à 4

instructions

range(i,j,k): de i (inclus) à j (exclu) par pas de k

**sauter, quitter** une boucle : continue, break

## Fonctions:

**définition** de la fonction f à deux paramètres:

**def** f(x, y):

reponse = x\*\*2 + y\*\*2

return reponse

**appel** de la fonction: f()

variable = f(3, 5)

print( f(2, 6) )

etc.

# Modèle OSI

