

## Examen

L'examen est sur 20 points et comprend 4 pages. Les quatre exercices sont indépendants. Les supports de cours sont les seuls documents autorisés. Les réponses doivent être soigneusement justifiées. L'utilisation de téléphones portables ou calculatrices est interdite. Le barème est donné uniquement à titre indicatif. Un aide mémoire HTML et PHP est donné en page 4.

**An english version of this exam is available on demand.**

**Les instructions suivantes doivent être respectées scrupuleusement, sous peine de non-correction :**

- Reporter sur **toutes** les copies le numéro d'anonymat imprimé sur la copie principale. Cette copie **doit** être rendue, même blanche.
- Chaque copie doit être numérotée et mentionner le nombre total de copies rendues (1/3, 2/3, ...). Si une seule copie est rendue, la numéroté 1/1. Les copies non numérotées **ne seront pas corrigées**.
- Toutes les réponses doivent être rédigées **au stylo**, sur des copies. Les parties rédigées au crayon de papier ou sur les feuilles de brouillons **ne seront pas corrigées**.
- Les copies doivent être anonymées, mais uniquement après vérification d'identité par un surveillant et signature de la liste d'émargement.

### Questions de cours (3 points)

1. (1 point) Qu'est-ce qu'un *processus* ?
2. (1 point) Quel est le rôle d'un DNS (serveur de nom) ?
3. (1 point) Quel est le rôle d'un routeur ?

### Programmation PHP (7 points)

On souhaite réaliser un formulaire Web permettant de répondre à des sondages. L'idée est de générer *automatiquement* le questionnaire au moyen d'une description stockée dans un fichier texte. Le fichier texte, nommé « `sondage.txt` », contient sur chaque ligne un ensemble de champs, séparés par le caractère « % ». La figure 1 donne un exemple de fichier texte et un aperçu de la page générée.

```
Quelle est votre couleur favorite ?%Bleu%Rouge%Vert%Jaune%Noir
À quelle classe d'age appartenez vous?%0-19%20-29%30-39%40-49%50+
Combien d'heures par jour passez-vous devant un ordinateur?%0-1h%2-4h%plus de 4h
```

Quelle est votre couleur favorite ?

À quelle classe d'age appartenez vous?

Combien d'heures par jour passez-vous devant un ordinateur?

FIGURE 1 – Exemple de questionnaire

- (2 points) Écrire une fonction PHP `split_question` prenant en argument une chaîne de caractères *par référence*, composée de champs séparés par des « % » et renvoyant comme résultat un tableau à deux entrées :
  - Une entrée ayant pour indice "question" et contenant le premier champ de la chaîne (champ 0)
  - Une entrée ayant pour indice "choix" et contenant un tableau contenant les champs 1, ...,  $n - 1$  de la ligne aux indices 1, ...,  $n - 1$ .
 Vous pouvez supposer que la chaîne passée en argument est du bon format. Par exemple pour la première ligne du fichier donné à la figure 1, le tableau renvoyé sera de la forme :
 

```
"question" => "Quelle est votre couleur favorite ?"
"choix" => (1 => "Bleu"
            2 => "Rouge"
            3 => "Vert"
            4 => "Jaune"
            5 => "Noir")
```
- (1 point) Écrire une fonction `read_file` prenant en argument le nom d'un fichier texte sous forme d'une chaîne de caractères. La fonction renvoie un tableau dont les indices sont 1, ...,  $n$  et tel que l'indice  $i$  contienne le résultat de `split_question` appelé sur la  $i^{\text{ème}}$  ligne du fichier.
- (2.5 points) Écrire le fragment de code HTML/PHP réalisant la page Web. Votre code charge le contenu du fichier « `sondage.txt` » au moyen de la fonction « `read_file` ». Pour chaque question, le formulaire présente les choix sous la forme d'une liste déroulante. La  $i^{\text{ème}}$  liste déroulante a pour nom «  $q_i$  » et pour chaque liste, la valeur du choix est son indice. Par exemple, dans la figure 1, on souhaite que les valeurs passées par le formulaire HTML soient : «  $q_1=4$  », «  $q_2=3$  », «  $q_3=3$  ». Le formulaire envoie ces valeurs sur la page « `validation.php` » au moyen de la méthode GET, lorsque l'utilisateur presse sur le bouton « Valider ».
- (1.5 point) On souhaite maintenant compléter le fichier « `validation.php` ». Écrire une fonction « `validate` » qui recharge le fichier « `sondage.txt` » et vérifie que les paramètres passés en GET sont valides. Vous pouvez réutiliser les fonctions des questions précédentes sans les recopier. Le paramètre «  $q_i$  » est valide si :
  - il est présent
  - sa valeur est comprise entre 1 et  $n$ , où  $n$  est le nombre de choix possibles pour la  $i^{\text{ème}}$  ligne du fichier « `sondage.txt` »
 La fonction renvoie vrai si les paramètres sont corrects et faux sinon.

## Ligne de commande Unix (6 points)

On considère que le répertoire courant contient les fichiers et répertoires suivants :

|              |              |              |             |             |        |
|--------------|--------------|--------------|-------------|-------------|--------|
| img_1001.jpg | img_1005.jpg | img_2003.jpg | img_10.jpg  | track03.mp3 | public |
| img_1002.jpg | img_1006.jpg | img_2004.jpg | img_100.jpg | test.exe    |        |
| img_1003.jpg | img_2001.jpg | img_2005.jpg | track01.mp3 | divers      |        |
| img_1004.jpg | img_2002.jpg | img_1.jpg    | track02.mp3 | perso       |        |

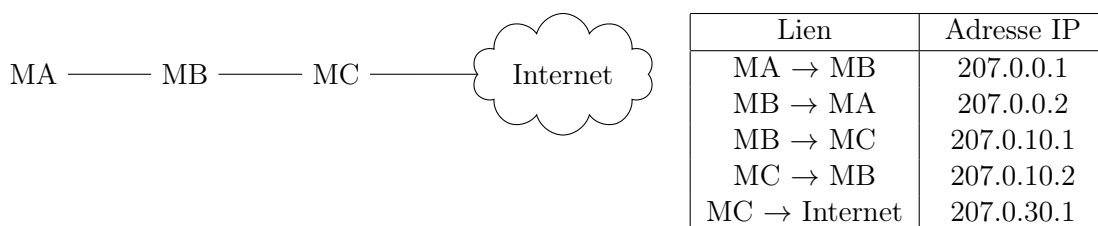
- `divers`, `public` et `perso` sont des répertoires, les autres noms dénotent des fichiers
  - tous les fichiers et répertoires appartiennent à l'utilisateur `lambda` (de même que le répertoire courant)
- (0.5 point/réponse) Pour chacun des ensembles de fichiers suivants, donner une *unique* expression régulière *glob* qui sélectionne cet ensemble :
    - tous les fichiers ayant l'extension « `mp3` »
    - tous les fichiers dont le nom contient un chiffre
    - tous les fichiers image (extension « `jpg` ») dont le numéro est pair
    - tous les fichiers image (extension « `jpg` ») dont le numéro est supérieur à 1000
    - tous les fichiers dont l'extension contient un chiffre ou un « `x` »
    - tous les fichiers dont le nom ne commence ni par un « `i` » ni par un « `p` »
  - (0.5 point/réponse) Donner les commandes Unix permettant d'effectuer les actions suivantes (les questions sont indépendantes, on suppose que l'on revient à la situation de départ pour chaque question) :

- rendre le fichier « **test.exe** » exécutable et modifiable par l'utilisateur et lisible par l'utilisateur, le groupe et les autres.
- afficher le contenu du répertoire « **perso** »
- rendre le répertoire « **public** » lisible par tous (utilisateur, groupe, autres), modifiable par l'utilisateur uniquement et traversable par l'utilisateur et le groupe auquel il appartient
- effacer tous les fichiers d'extension « **.jpg** »
- faire de « **divers** » le répertoire courant
- afficher le contenu détaillé du repertoire courant (liste de fichiers avec leur permissions, propriétaire/groupe, taille *etc.*)

## Réseaux, TCP/IP (4 points)

Répondre aux questions suivantes **en justifiant** :

- (0.5 point) On considère un réseau de 300 machines, avec un adressage par classe (la classe de l'adresse IP détermine le numéro de réseau et l'adresse de la machine). À quelle(s) classe(s) peuvent appartenir les adresses de ce réseau ?
- On considère le réseau suivant :



De plus, les tables de routage sur les machines MB et MC sont les suivantes :

|             |            |             |             |         |       |
|-------------|------------|-------------|-------------|---------|-------|
| <b>MC :</b> | 207.0.0.*  | MB-MC       | <b>MB :</b> | *.*.*.* | MB-MC |
|             | 207.0.10.* | MB-MC       |             |         |       |
|             | *.*.*.*    | MC-Internet |             |         |       |

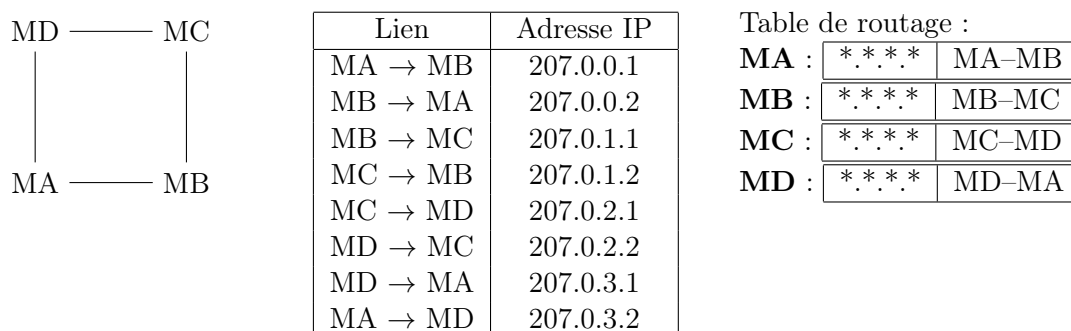
On rappelle qu'une table de routage indique quel lien utiliser selon l'adresse de destination. Par exemple, sur la machine MC, les paquets a destination 207.0.0.44 sont renvoyés sur le lien MB-MC. Le symbole \* signifie n'importe quel nombre entre 0 et 255. Si plusieurs entrées de la table de routage sont possibles, la première en partant du haut est utilisée.

Décrire **précisément** les étapes des opérations suivantes (quels nœuds sont visités, vers quel nœud un paquet est retransmis et pourquoi, *etc.*) :

- (0.5 point) Envoi d'un paquet IP de MA vers une machine sur Internet
- (0.5 point) Envoi d'un paquet IP d'une machine sur Internet vers MA. On suppose que le paquet arrive en MC avec un TTL de 3

(1 point) Quel est le problème et comment le corriger ?

- On considère maintenant le réseau ci-dessous :



- (0.5 point) Toutes les machines sont-elles atteignables depuis n'importe quelle source ?
- (1 point) On suppose que le lien MA-MD est coupé. Modifier les tables de routage pour que chaque machine puisse atteindre toutes les autres.

# Aide mémoire HTML et PHP

## HTML

| Élément                                                           | Description                                                                                                                                                      |
|-------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>&lt;a href="url"&gt; texte &lt;/a&gt;</code>                | Affiche <i>texte</i> comme un lien hypertexte dont la cible est <i>url</i> .                                                                                     |
| <code>&lt;b&gt; texte &lt;/b&gt;</code>                           | Affiche <i>texte</i> en gras.                                                                                                                                    |
| <code>&lt;body&gt;...&lt;/body&gt;</code>                         | Élément principal contenant le code HTML de la page.                                                                                                             |
| <code>&lt;br /&gt;</code>                                         | Retour à la ligne.                                                                                                                                               |
| <code>&lt;button type="submit"&gt; texte &lt;/button&gt;</code>   | Bouton dans un formulaire (à utiliser sous un <b>form</b> ). <i>texte</i> est utilisé comme étiquette du bouton.                                                 |
| <code>&lt;form method="m" action="url"&gt;...&lt;/form&gt;</code> | Formulaire. <i>url</i> est l'adresse cible du formulaire. <i>m</i> doit valoir "post" ou "get".                                                                  |
| <code>&lt;h1&gt;...&lt;/h1&gt;</code>                             | Titre de section.                                                                                                                                                |
| <code>&lt;head&gt;...&lt;/head&gt;</code>                         | En-tête du document HTML contenant le titre et d'autres méta-données                                                                                             |
| <code>&lt;html&gt;...&lt;/html&gt;</code>                         | Racine du document, ouverte en premier et fermée en dernier.                                                                                                     |
| <code>&lt;i&gt;texte&lt;/i&gt;</code>                             | Affiche <i>texte</i> en italique.                                                                                                                                |
| <code>&lt;img src="url" /&gt;</code>                              | Insère une image dont l'URL est <i>url</i> .                                                                                                                     |
| <code>&lt;input type="text" name="n" /&gt;</code>                 | Zone de saisie dans un formulaire (à utiliser sous un <b>form</b> ). Le nom du paramètre associé est <i>n</i> .                                                  |
| <code>&lt;li&gt;...&lt;/li&gt;</code>                             | Élément d'une liste (à utiliser en dessous d'un <b>ul</b> ou <b>ol</b> )                                                                                         |
| <code>&lt;ol&gt;...&lt;/ol&gt;</code>                             | Liste ordonnée.                                                                                                                                                  |
| <code>&lt;option value="v"&gt; texte &lt;/option&gt;</code>       | Choix dans une liste déroulante (doit être utilisé sous un <b>select</b> ). La valeur du choix est <i>v</i> et le texte affiché dans la liste est <i>texte</i> . |
| <code>&lt;p&gt;...&lt;/p&gt;</code>                               | Paragraphe.                                                                                                                                                      |
| <code>&lt;select name="n"&gt; ... &lt;/select&gt;</code>          | Liste déroulante (contient une suite de <b>option</b> et doit apparaître sous un <b>form</b> ). Le nom du paramètre est <i>n</i> .                               |
| <code>&lt;ul&gt;...&lt;/ul&gt;</code>                             | Liste non-ordonnée.                                                                                                                                              |

## PHP

| Fonction ou variable   | Description                                                                                                                                                             |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| .                      | Concaténation de chaînes                                                                                                                                                |
| array()                | Création d'un nouveau tableau vide                                                                                                                                      |
| count(\$a)             | Renvoie le nombre d'éléments du tableau <b>\$a</b>                                                                                                                      |
| echo <i>e</i>          | Écrit la valeur de l'expression <i>e</i> dans le document HTML                                                                                                          |
| file(\$f)              | Renvoie un tableau indicé par 0, ..., $n - 1$ contenant les <i>n</i> lignes du fichier dont le nom est passé en argument. Les retour à la lignes finaux sont conservés. |
| file_get_contents(\$f) | Renvoie le contenu du fichier dont le nom est passé comme paramètre sous form de chaîne                                                                                 |
| preg_match(\$r, \$s)   | Renvoie le nombre d'occurences de l'expression régulière <b>\$r</b> dans la chaîne <b>\$s</b> .                                                                         |
| preg_split(\$r, \$s)   | Sépare la chaîne <b>\$s</b> en fonction de l'expression régulière <b>\$r</b> et renvoie le résultat comme un tableau.                                                   |
| \$_GET                 | Tableau global contenant les paramètres de formulaires passés avec la méthode <b>get</b> .                                                                              |
| \$_POST                | Tableau global contenant les paramètres de formulaires passés avec la méthode <b>post</b> .                                                                             |
| \$_COOKIE              | Tableau global contenant les <i>cookies</i> .                                                                                                                           |
| \$_SESSION             | Tableau global contenant les variables de session.                                                                                                                      |