

## Décomposition, Conception et Réalisation d'Applications

### 1. Extraction des commentaires d'un programme

On s'intéresse ici à l'extraction des commentaires généraux d'un programme. On pourra imaginer par exemple qu'il s'agit de lignes qui commencent par `###` en PHP, R, PERL... ou qui commencent par `/* ###` en C ou C++.

#### Question 1.1

En admettant que vous vouliez juste dans un premier temps extraire de telles lignes **d'un seul fichier**, comment feriez-vous ? Avec un outil en ligne de commandes ou par programme ? Vous indiquerez quel outil ou quel langage vous utiliseriez alors. On ne demande pas de fournir de programme ici.

#### Question 1.2

Si maintenant vous deviez produire un "vrai" document modifiable pour Word ou LaTeX contenant de telles lignes **pour tout un ensemble de fichiers**, disons par exemple pour le programme principal et ses fichiers inclus, comment feriez-vous ? Avec un outil en ligne de commandes ou par programme ? Vous indiquerez là aussi quel outil ou quel langage vous utiliseriez alors. Pas de programme à fournir ici non plus.

#### Question 1.3

Est-ce qu'il existe déjà des **outils de création automatique de documents à partir de commentaires** pour les programmes des langages classiques (C, C++, PERL, PHP...) ? Si oui, lesquels ?

### Question 1.4

Donner, dans la syntaxe des expressions régulières étendues au sens de PERL ou au sens de PCRE – la syntaxe utilisée par PHP :

- une expression régulière qui indique si une ligne contient ###
- une expression régulière qui indique si une ligne commence par ###
- une expression régulière qui indique si une ligne contient \* ###
- une expression régulière qui indique si une ligne contient une étoile suivie d'un ou plusieurs caractères "blancs" comme des espaces, des tabulations... puis de trois à cinq caractères # suivis d'au moins un espace.

### Question 1.5

Donner le code d'une fonction PHP nommé `extrCmt()` qui prend en paramètre le nom d'un fichier, qui teste si le fichier existe, et, si c'est le cas, extrait les lignes de ce fichier comportant au moins deux caractères # consécutifs à l'aide d'une expression régulière.

## 2. Comment afficher des graphes dont des arbres ?

On voudrait tracer des graphes et des arbres au sens de la théorie des graphes dans une page Web. On imagine facilement qu'il y a deux familles de solutions à ce problème :

- 2.1 une solution sur le client, avec un rendu graphique dans le navigateur via Javascript et une bibliothèque de fonctions adaptées, par exemple D3.js ou Raphael.js,
- 2.2 une solution sur le serveur, avec une image produite disons en PHP et un logiciel adapté, par exemple dot qui fait partie de la suite logicielle GraphViz.

Laquelle de ces solutions vous paraît la plus adaptée au problème, et surtout, pourquoi ?

Vous ne vous contenterez pas de choisir une des solutions, vous viendrez également expliquer pourquoi l'autre solution est moins bonne selon vous.

Si maintenant vous aviez le choix de programmer une solution, serait-ce une de ces solutions que vous utiliseriez ou une autre ? Dans ce dernier cas, quelle serait cette autre solution ? Là encore, vous détaillerez le pourquoi de votre choix. On ne demande ici aucun programme, mais juste une discussion.

### 3. Un peu de culture...

Essayez de répondre à la question suivante :

*Il est certainement intéressant d'utiliser un framework en développement Web, par exemple en initiation pour être sûr de bien utiliser le motif MVC (Modèle / Vue / Contrôleur), mais comment choisir un tel framework ?*

Votre réponse devra essayer de mettre en évidence votre culture naissante, votre recul et votre esprit de synthèse en matière de modélisation et de traitement de l'information.

Cette réponse devra faire 10 lignes au minimum, sans limite de maximum. On utilisera au moins 3 mots de 4 syllabes ou plus pour « transmettre un contenu rédactionnel fort ».

Vous pourrez, mais ce n'est pas obligatoire, discuter des *frameworks* Laravel, CodeIgniter et Symfony pour PHP dans votre rédaction.

## ESQUISSE DE CORRIGÉ

### Question 1.1

Un outil comme grep est sans doute suffisant mais rien n'interdit d'utiliser un script en Php, Python ou Perl.

### Question 1.2

Un outil comme grep est sans doute insuffisant ici, à cause des *includes* et des références possiblement circulaires. Donc utiliser un script en Php, Python ou Perl sera sans doute ici un bon choix.

### Question 1.3

Des outils connus : doxygen (multilangages), javadoc (pour java), sphynx (pour python).

### Question 1.4

Voici des expressions régulières solutions, que l'on peut tester sur le site <http://rubular.com/> :

```
/###/
```

```
/^###/
```

```
/\* ###/
```

```
/\*\s+#{3,5} /
```

### Question 1.5

On peut utiliser une méthode classique de parcours ligne par ligne ou une méthode plus "moderne" avec récupération de tout le contenu du fichier dans une variable texte, découpage par ligne et passage en revue du tableau.

L'expression régulière à utiliser est :

```
/#{2,}/
```

```

<?php

#####
#                                                                 #
#   Extraction des commentaires avec deux symboles # successifs #
#   dans un fichier-programme en PHP                    #
#                                                                 #
#####

function extrCmt($nomFichier="") {

if ($nomFichier=="") {
    echo "Aucun nom de fichier fourni. STOP\n" ;
    exit(-1) ;
} ; # fin si

if (!file_exists($nomFichier)) {
    echo "Fichier $nomFichier non vu. STOP\n" ;
    exit(-1) ;
} ; # fin si

# méthode moderne

$texteFichier = file_get_contents( $nomFichier) ;
$tabLignes = preg_split("/\n/", $texteFichier) ;
foreach ($tabLignes as $ligne) {
    if (preg_match("/#{2,}/", $ligne)) {
        echo $ligne." \n" ;
    } ; # fin si
} ; # fin pour chaque ligne

# méthode classique

$fh = fopen($nomFichier,"r") ;
while ($ligne = fgets($fh,4096)) {
    if (preg_match("/#{2,}/", $ligne)) {
        echo $ligne ; # pas de \n !
    } ; # fin si
} ; # fin pour chaque ligne
fclose($fh) ;

} # fin de fonction extrCmt

?>

```

## Question 2

Sans plus de renseignements sur la génération du graphe, les deux solutions se valent sans doute.

Si les données ne sont pas trop volumineuses et en tous cas pas confidentielles ou privées, privilégier un traitement en local sur le client permettra de ne pas encombrer le serveur.

Si les données sont extraites d'une base de données, utiliser un script sur le serveur est sans doute plus adapté.

Une autre solution possible serait de télécharger les données et de les traiter en local sur l'ordinateur du client, indépendamment d'une page Web, quitte à ajouter ensuite un habillage en HTML.