

**Sujet d'examen**

**Semestre 2 / session 1**

**Année 2016/2017**

**CODE UE : NFA017**

**INTITULE** : Développement web (2) : sites dynamiques et développement côté serveur

**DATE** : Samedi 03/06/2017

**HORAIRES** : 14:00 - 16:00

**DUREE** : 2H

**CONSIGNES** : Tout document papier autorisé

**NOMBRE DE PAGES** : 8

**QCM : (10 points)**

Répondre sur le même document.

Bonne réponse : 1 point

Mauvaise réponse : -0.25 point

Aucune réponse : 0 point

Cochez uniquement la bonne réponse.

Q1. Quelle instruction permet d'ouvrir un fichier en écriture afin d'ajouter des données à la fin de son contenu ?

- ☐ fopen("fichier","r+")
- ☐ fopen("fichier","w")
- ☐ fopen("fichier","a")

Q2. Quelle fonction retourne la longueur d'une chaîne de texte ?

- ☐ strlen
- ☐ strlenlength
- ☐ length
- ☐ Substr

Q3. Par défaut, dans quel ordre de priorité sont affectées les variables envoyées par HTTP ?

- ☐ get, cookie, post
- ☐ post, get, cookie
- ☐ get, post, cookie
- ☐ post,cookie, get

Q4. Comment mémorise-t-on de façon pratique une variable d'un script à l'autre sur un serveur web ?

- Grâce à `serialize()`
- En utilisant `$_SESSION`
- Au moyen de `register_globals()`

Q5. Quel est l'effet de l'utilisation de la structure `return();` dans un fichier inclus (sans être dans une fonction) ?

- Retour au script parent sans exécuter la suite du fichier.
- Une erreur PHP.
- Arrêt du script à la manière de `exit()`.
- Aucun, il est ignoré.
- Cela dépend s'il a été inclus avec `require()` ou `include()`.

Q6. Que donne le code suivant en PHP5 ?

```
$b = false;  
$a = unset($b) ;  
var_dump($a) ;
```

- NULL
- Une erreur PHP.
- `bool(true)`
- `bool(false)`

Q7. Quelle fonction retourne le nombre de secondes écoulées depuis le 1er janvier 1970 ?

- ☐ time
- ☐ timestamp
- ☐ mktime
- ☐ microtime

Q8. Quelle fonction permet d'envoyer des en-têtes HTTP au navigateur avant le contenu de la page ?

- ☐ parse\_url()
- ☐ http\_post()
- ☐ header()

Q9. Comment peut-on trier un tableau en ordre inverse ?

- ☐ \$tableau = ksort(\$tableau);
- ☐ ksort(\$tableau)
- ☐ \$tableau = rsort(\$tableau);
- ☐ rsort(\$tableau);

Q10. Comment définit-on une constante ?

- `set('maconstante' = 'valeur');`
- `define("maconstante","valeur");`
- `const $maconstante = valeur;`

**Exercice 1 (5 points):**

Effectuer des opérations sur les tableaux associatifs.

Créer un tableau associatif qui contient les notes des étudiants, ce tableau se compose de couples clé=>valeur (les clés sont les noms des étudiant et les valeurs représentent les notes).

1. Créer et initialiser un tableau \$notes avec les valeurs suivantes :

clé	valeur
Arnaud	13
Fabien	16
Nico	15

2. Ajouter au tableau la note 10 de l'étudiant "karim".
3. supprimer la note de l'étudiant " Fabien ".
4. Déterminer la note maximale et la note minimale du groupe.
5. Afficher le tableau après l'avoir trier par ordre alphabétique.
6. Classer les étudiants par ordre de mérite et afficher le tableau.
7. Déterminer la moyenne de la classe.

```
C:\wamp64\www\note.php:14:
array (size=4)
  'Arnaud' => int 13
  'Fabien' => int 16
  'Nico' => int 15
  'karim' => int 10
```

```
-----

C:\wamp64\www\note.php:18:
array (size=3)
  'Arnaud' => int 13
  'Nico' => int 15
  'karim' => int 10
```

```
-----
la note maximale est: 15 la note minimale est: 10
-----
```

```
la note de l'étudiant Arnaud est : 13
la note de l'étudiant Nico est : 15
la note de l'étudiant karim est : 10
-----
```

```
la note de l'étudiant Nico est : 15
la note de l'étudiant Arnaud est : 13
la note de l'étudiant karim est : 10
-----
```

```
la moyenne de la classe: 12.67
```

### Exercice 2 (5 points):

1. Écrire une classe **Complexe** permettant de représenter des nombres complexes.

#### Note

Un **nombre complexe** est un nombre qui peut s'écrire sous la forme :

$$Z = \text{PartieRéelle} + \text{PartieImaginaire} * i$$

Où **PartieRéelle** et **PartieImaginaire** sont des nombres réels et  $i$  un nombre imaginaire tel que  $i^2 = -1$ .

### Calcul avec des nombres complexes :

Soit les deux nombres complexes suivants :  $z_1=a+ib$  et  $z_2=x+iy$ .

**Addition :**  $z_1+z_2=(a+x)+i*(b+y)$

**Soustraction :**  $z_1-z_2=(a-x)+i*(b-y)$

**Produit :**  $z_1*z_2=(a*x)-(b*y)+i*(x*b+a*y)$

**Division :**  $z_1/z_2=((a*x+b*y)/(x^2+y^2)) + i*((b*x-a*y)/(x^2+y^2))$

2. Définir les méthodes d'accès aux attributs de la classe.
3. Définir un constructeur d'initialisation pour la classe.
4. Ajouter les méthodes suivantes :

**Ajouter(Complexe) :** Elle permet de renvoyer le nombre complexe obtenu en ajoutant au nombre en cours un nombre complexe passé en argument.

**Soustraire(Complexe)**

**Multiplier(Complexe).**

**Diviser(Complexe).**

**la méthode magique \_\_toString() :** qui permet de représenter le nombre complexe sous la forme : **a+b\*i**.

6. Écrire un script permettant de tester la classe Complexe :

**Exemple d'exécution :**

le premier nombre complexe est :

$$3+1*I$$

le deuxieme nombre complexe est :

$$1+2*I$$

L'Addition des deux nombres renvoie :

$$4+3*I$$

La soustraction des deux nombres renvoie :

$$2-1*I$$

La multiplication des deux nombres renvoie :

$$1+7*I$$

La division des deux nombres renvoie :

$$1-0.2*I$$