



مكتب التكوين المهني وإنعاش الشغل

Office de la Formation Professionnelle  
et de la Promotion du Travail

Direction Recherche et Ingénierie de Formation

**Examen de passage - Session Juin 2011**

**Eléments de Correction**

**Filière :** Techniques de Développement Multimédia

**Epreuve :** Théorique

**Niveau :** Technicien Spécialisé

**Durée :** 4 h 30 mn

**Barème :** 40 Pts

**PARTIE 1 : CONNAISSANCES GENERALES ET LOGIQUES (5 Pts)**

**Exercice 1 :**

- 1) le nombre entier positif maximal que l'on peut stocker dans  $M_1$  ou  $M_2$  est 63
- 2) le nombre entier négatif minimal que l'on peut stocker dans  $M_1$  ou  $M_2$  est -64
- 3) Résultat de l'addition de  $M_1+M_2$ 
  - a. Le résultat est 39 ( 00011000 + 00001111)
  - b. Le résultat est -3 ( 00101010 + 00011001 )

**Exercice 2 :**

481 en binaire est : 000111100001  
481 en complément à 1 : 111000011110  
481 en complément à 2 : 111000011111  
481 en code DCB : 010010000001

**Exercice 3 :**

- La simplification de  $(x + \bar{y}) + (x + y) \oplus \bar{y}$  donne **1**
- La simplification de  $(x + y + y.z) . (y.z)$  donne **y.z**

**PARTIE 2 : MULTIMEDIA (8 Pts)**

**Propositions de réponses :**

- 1- Les images, dessins, logos... de types "vectoriels" sont constitués de droites et de courbes calculées de façon mathématique et peuvent être créés généralement à l'aide du logiciel Illustrator. A l'opposé, les documents en mode bitmap sont uniquement pixélisés (stockés sous forme de matrice dont les points représentent des pixels) travaillés à l'aide de logiciels comme Photoshop.

2- Un document destiné à être imprimé doit être d'une meilleure qualité (plus grande résolution, plus grande taille en mémoire...) qu'un document destiné à être simplement affiché à l'écran.

3- Les principales fonctionnalités d'un logiciel de traitement d'images sont :

- Retouches chromatiques.
- Montage photos.
- Créations d'interfaces et de galeries Web pour Internet.
- Réalisations graphiques (outils de dessin)

4- Voici quelques outils de « sélection » dans un logiciel de traitement d'images : le rectangle de sélection, le lasso et la baguette magique.

5-

- ◆ Résolution d'une image :

La résolution est une mesure de la finesse de l'affichage ou de la capture d'une image, exprimée en nombre de pixels par unité de surface, c'est à dire la « densité » en pixels , en pixels par pouce (ppp) ou dots per inch (dpi) en anglais. Les pixels sont de petits carrés pouvant contenir des informations de couleur. Une image est donc un ensemble de pixels, placés les uns à côté des autres. Dans une photo, plus le nombre de pixels par pouce est élevé, plus il y a de détails .

- ◆ Retouche d'une image :

La retouche d'image désigne tout procédé qui consiste à modifier une image. En photographie, la retouche consiste à corriger des défauts d'aspect, éclaircir ou assombrir l'image, augmenter ou diminuer les contrastes, saturer des couleurs etc.. La retouche d'image peut aussi faire référence à des modifications plus importantes. Si on transforme une photographie pour ajouter une personne ou en enlever une (par exemple), on parle alors de photomontage ou de trucage photographique.

6- Pour choisir un format de fichier, cela suppose de savoir à quoi va servir l'image. Vidéo, Multimédia ou Imprimerie. Voici quelques formats d'enregistrement d'images :

- **Le JPEG** (Joint Photographic Experts Group)
- **Le Gif** (Graphic Interchange Format)
- **Le PNG** (Portable Network Graphics)
- **Le PDF** (PostScript Printer Description)
- **Le PSD** pour travailler une image dans Photoshop, ce format permet de conserver les calques créés au fur et à mesure du travail. C'est le format natif du logiciel Photoshop.

Voici quelques formats de fichiers audio :

- Les formats audio sans compression : RAW , WAV , BWF , AIFF , CAF, CDA.
- Les formats audio compressés sans perte : ATRAC , FLAC
- Les formats audio compressés avec perte : AC3 , MP3 , mp3PRO, Ogg Vorbis, VQF ou TwinVQ , WMA, AU , ASF, AA, AAC ou MPEG-2 AAC
- Formats multipistes : iKlax, U-MYX, MXP4

**PARTIE 3 : ALGORITHMES ET PROGRAMMATION STRUCTUREE (9 PTS)**

**Exercice 1 : (3 Pts)**

**Fonction puissance( x : réel, n :entier) : réel**

Var : i : entier ;

p :réel ;

**Début**

Si n=0 alors

Return 1 ;

Sinon si n >0 alors

P ← 1 ;

Pour i de 1 à n faire

P ← p\*x ;

Finpour

return p ;

Sinon

Return puissance (1/x, -n) ;

Finsi

**Fin**

**Exercice 2 : Recherche du signe astrologique correspondant à une date : (6 Pts)**

1. type : **sDate** = Structure

jour : entier(2) ;

mois : entier(2) ;

Fin structure

2. type : **signeAstro** = Structure

dateDebut : sDate ;

dateFin : sDate ;

signe :chaîne de caractères ( 20) ;

Fin structure

3. Var :

T : **Tableau**(30) de **sDate** ;

i, taille : entiers ;

**Debut**

Lire la taille du tableau ;

Ecrire (" Entrer la taille du tableau : " ) ;

Lire(taille) ;

Pour i allant de 1 à taille faire

Ecrire (" Entrer le jour" ,i) ;

Lire (T(i).jour) ;

Ecrire (" Entrer le mois " ,i) ;

Lire (T(i).mois) ;

<<Comparer avec le tableau des signes astrologiques>> ;

<<Afficher le résultat>> ;

**FinPour**

**Fin**

#### **PARTIE 4 : PROGRAMMATION ORIENTEE OBJET (12 PTS)**

Le formateur doit analyser chaque réponse en fonction du langage utilisé ( VB.Net, C# ou Java)

#### **Proposition de solution en java :**

```
1- Public class montre{
    int heure; // entre 0 et 23
    int minute; // entre 0 et 59
    int seconde; // entre 0 et 59
2- // constructeur avec paramètres
    public montre(int h, int m, int s)
    {
        if((h>=0) && (h<=23) && (m>=0) &&(m<=59) && (s>=0) && (s<=59)){
            this.heure = h;
            This.minute = m;
            This.seconde = s; }
        else {
            system.out.println( "Les valeurs proposees sont incorrectes" );
            system.out.println ("le temps sera fixé par défaut : 00:00:00 " );
            this.heure = 0;
            This.minute = 0;
            This.seconde = 0; }
        }
// getters **** accesseurs en lecture
    Int getHeure(){
        Return this.heure ;
    }
    Int getMinute(){
        Return this.minute ;
    }
    Int getSeconde(){
        Return this.seconde ;
    }

// setters **** accesseurs en modifications
    Void setHeure(int h){
        This.heure=h ;
    }
    Void setMinute(int m){
        This.minute=m ;
    }
    Void setSeconde(int s){
        This.seconde=s ;}

// affichage des informations d'une montre
    String toString(){
        Return ( "Il est :" +this.heure + "et " + this.minute + "minutes et " + this.seconde " secondes" );
    }
3- // avancer le temps marqué par une montre d'une seconde
    void TicTac(){
```

```

if(this.seconde <59)  this.seconde++;
else {
  this.seconde=0;
  if(this.minute <59)  this.minute++;
  else {
    this. minute=0;
    if(this.heure<23)
this. heure++;
    else {
      this.heure=0;
      this.seconde=0 ;}}}}

```

4- // comparer le temps de deux montre

```

Boolean compareMontre( montre m) {
if ( this.heure = m .heure ) & ( this.minute = m .minute ) & ( this.seconde = m .seconde ){
  return true;
else
  return false;} } }

```

#### **PARTIE 5 : DÉVELOPPEMENT WEB (6 PTS)**

#### **Voici une proposition de solution :**

```

<html>
  <head>
  <title>Calcule du périmètre et de l'aire d'un rectangle</title>
  </head>
  <body>
  <SCRIPT LANGAGE="JavaScript">
  function calcul()
  {Largeur = window.document.forme1.champ1.value;
  Langueur = window.document.forme1.champ2.value;
  Aire = Largeur*Langueur;
  Perimetre=2*(parseInt(Largeur)+parseInt(Langueur));
  window.document.forme1.champ3.value='Le périmètre est de ' + Perimetre ;
  window.document.forme1.champ4.value='L aire est de ' + Aire ;}
  </script>
  <form name="forme1">
  <p> Largeur <input style="background-color: rgb(0, 255, 0);" size="20" name="champ1"
type="text">
  <p> Langueur <input style="background-color: rgb(0, 255, 0);" size="20" name="champ2"
type="text">
  Périmètre: <input size="35" name="champ3" disabled="disabled" type="text">
  Aire:<input size="35" name="champ4" disabled="disabled" type="text">
  <input name="resultype1" value="Calculer" type="button" onclick="calcul()">
  </p>
  </form></body> </html>

```