



OFPPT

مكتب التكوين المهني وإنعاش الشغل

Office de la Formation Professionnelle et de la Promotion du Travail

Direction Recherche et Ingénierie de la Formation

Examen de passage à la 2<sup>ème</sup> année

Session Juillet 2018

Filière : Techniques de Développement Informatique

Epreuve : Synthèse

Niveau : TS

Variante : V1

Durée : 5 heures

Barème : / 120 pts

Eléments de réponse

## Partie I : Théorie

40 pts

## Dossier 1 : L'essentiel en technologies de l'information

12 pts

## 1. 0.5 pt pour chaque conversion

6 pts

| Binaire  | Octal | Décimal | Hexadécimal |
|----------|-------|---------|-------------|
| 11000011 | 303   | 195     | C3          |
| 110111   | 67    | 55      | 37          |
| 1000011  | 103   | 67      | 43          |
| 1100111  | 147   | 103     | 67          |

## 2.

## a. La fonction f :

$$f(a, b, c, d) = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + ab\bar{c} + abc$$

1.5 pts

Simplification analytique :

$$f(a, b, c, d) = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + ab\bar{c} + abc$$

$$f(a, b, c, d) = \bar{a}\bar{b}(\bar{c}+c) + ab(\bar{c}+c)$$

$$f(a, b, c, d) = \bar{a}\bar{b} + ab$$

1.5 pts

## b. Simplification moyennant le tableau de Karnaugh :

|           | $\bar{a}\bar{b}$ | $\bar{a}b$ | $ab$ | $a\bar{b}$ |
|-----------|------------------|------------|------|------------|
| $\bar{c}$ | 1                | 0          | 1    | 0          |
| $c$       | 1                | 0          | 1    | 0          |

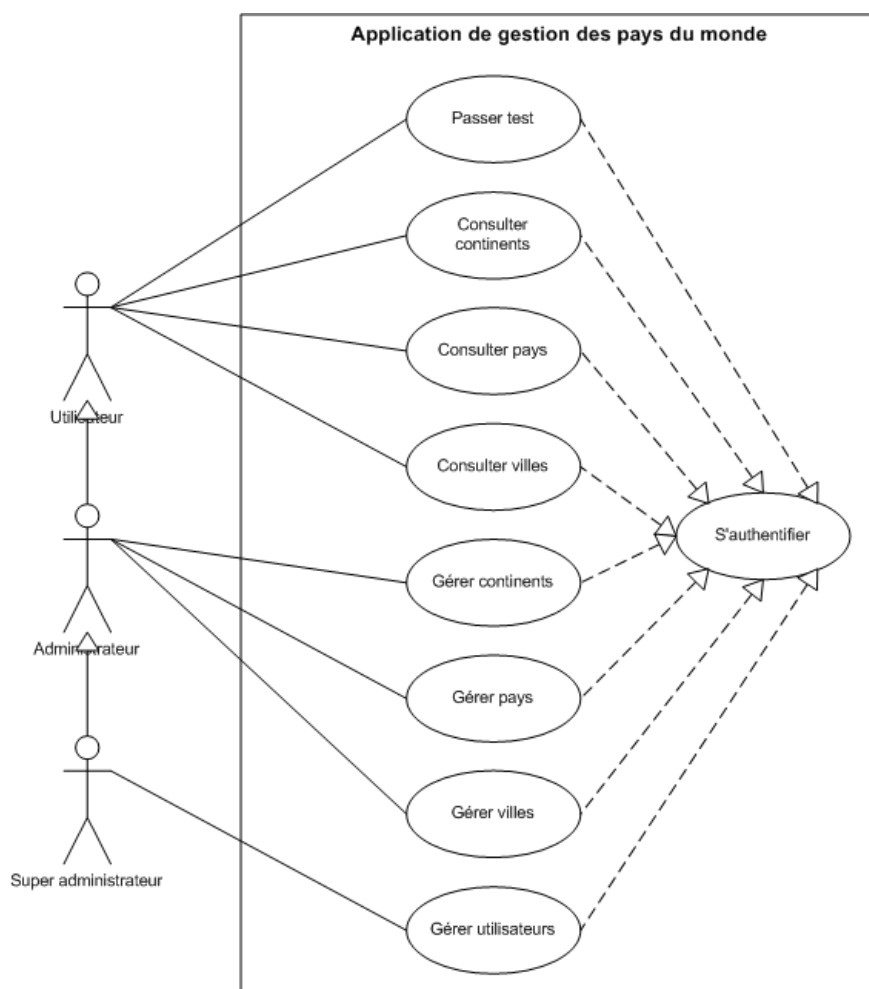
$$f(a, b, c, d) = \bar{a}\bar{b} + ab$$

3 pts

| Filière | Epreuve                             | Session      | 1 / 8 |
|---------|-------------------------------------|--------------|-------|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |       |

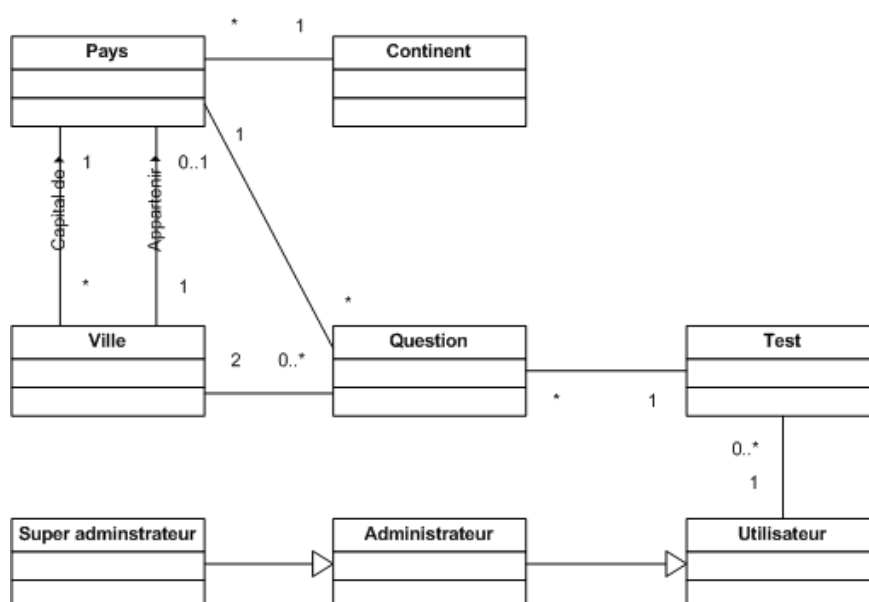
1. Diagramme de cas d'utilisation : 0.25 pt pour chaque élément (acteur, Cas d'utilisation et association)

7.5 pts



2. Diagramme de classes : 0.5 pt pour chaque élément (classe et association)

8.5 pts



| Filière | Epreuve                             | Session      | 2 / 8 |
|---------|-------------------------------------|--------------|-------|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |       |

**Dossier 3 : Programmation structurée****12 pts**

Algorithme    Gestion des villes

Structure    Ville

Nom                :    Chaîne

Population       :    Entier

Pays              :    Chaîne

**1 pt**

FinStructure

Constante    nMax  $\leftarrow$  20

Variables    villes[nMax]                :    Villes

**1 pt**

n, i  
populationTotal,  
populationMax,  
indiceVillePeuplee :    Entier  
nomPays                :    Chaîne

Début

Répéter

Ecrire("Nombre de villes ? ")  
Lire(n)

**1 pt**

TantQue(NON(n &gt; 0 ET n &lt;= nMax))

Pour i  $\leftarrow$  0 jusqu'à n - 1

Ecrire("Ville ", i + 1, " : \n")

Ecrire("Nom : ")  
Lire(villes[i].Nom)

Ecrire("Population : ")  
Lire(villes[i].Population)

**2 pts**

Ecrire("Pays : ")  
Lire(villes[i].Pays)

FinPour

Ecrire("Pays dont vous voulez calculer la population totale : ")  
Lire(nomPays)

populationTotale  $\leftarrow$  0Pour i  $\leftarrow$  0 jusqu'à n - 1

Si(nomPays = villes[i].Pays)  
    populationTotale += villes[i].Population

**3 pts**

FinSi

FinPour

Ecrire("La population totale est : ", populationTotale)

populationMax  $\leftarrow$  0**4 pts**

| Filière | Epreuve                             | Session      | 3 / 8 |
|---------|-------------------------------------|--------------|-------|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |       |

```

Pour i ← 0 jusqu'à n - 1
    Si(populationMax < villes[i].Population)
        populationMax ← villes[i].Population
        indiceVillePeuplee ← i
    FinSi
FinPour
Ecrire("Ville la plus peuplée : ", villes[indiceVillePeuplee].Nom)
Fin

```

## Partie II : Pratique

80 pts

### Dossier 1 : Programmation structurée

16 pts

1.

- |    |                      |   |       |
|----|----------------------|---|-------|
| a. | $f(275) \rightarrow$ | 3 | 2 pts |
| b. | $f(27) \rightarrow$  | 2 | 2 pts |
| c. | $f(2) \rightarrow$   | 1 | 2 pts |

2.

```

int chiffre(int nombre, int place)
{
    int i = 1;

    while(1 == 1)
    {
        if(i == place) return nombre % 10;

        nombre = nombre / 10;
        i++;
    }

    return 0;
}

```

3.

```

void main()
{
    int e, nombreChiffres;

    //a.
    do
    {
        printf("Donner un nombre compris entre 1 et 99999 : ");
        scanf("%d", &e);
    } while(!(e >= 1 && e <= 99999));

    //b.
    nombreChiffres = f(e);
    printf("%d appartient a l'intervalle [%.0f, %.0f]",
        e, pow(10, nombreChiffres - 1), pow(10, nombreChiffres) - 1);

    getch();
}

```

| Filière | Epreuve                             | Session      | 4/8 |
|---------|-------------------------------------|--------------|-----|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |     |

```

class Pays
{
    //1. 1.5 pt
    private string _Nom;
    public string Nom
    {
        get { return _Nom; }
        set
        {
            //1.a 3 pts
            if (value.Length < 4)
                throw new Exception("Le nom doit contenir au moins 4 caractères");

            foreach (char caractère in value)
            {
                if (char.IsDigit(caractère))
                    throw new Exception("Le nom ne peut pas contenir des chiffres");
            }

            _Nom = value;
        }
    }

    //1.(suite) 1.5 pt
    private List<Ville> _Villes = new List<Ville>();
    public List<Ville> Villes
    {
        get { return _Villes; }
        set { _Villes = value; }
    }

    //1.(suite) 1.5 pt
    private Ville _Capitale;
    public Ville Capitale
    {
        get { return _Capitale; }
        set
        {
            //1.b 3 pts
            Ajouter(value);
            //La méthode vérifie l'existence de la ville

            _Capitale = value;
        }
    }

    //2.a 1.5 pt
    public Pays(string nom)
    {
        Nom = nom;
    }

    //2.b 2 pts
    public Pays(string nom, Ville capitale)
    {
        Nom = nom;
        Capitale = capitale;
    }
}

```

| Filière | Epreuve                             | Session      | 5/8 |
|---------|-------------------------------------|--------------|-----|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |     |

```

//3
public int NombreVilles
{
    get { return Villes.Count; }
}

//4
public int Rechercher(String nomVille)
{
    foreach (Ville ville in Villes)
        if (ville.Nom == nomVille) return Villes.IndexOf(ville);

    return -1;
}

//5
public void Ajouter(Ville ville)
{
    if (Rechercher(ville.Nom) == -1) Villes.Add(ville);
}

//6
public void Supprimer(String nomVille)
{
    int indiceVille = Rechercher(nomVille);
    if (indiceVille != -1)
    {
        Villes.RemoveAt(indiceVille);
        if (Capitale.Nom == nomVille) Capitale = null;
    }
}

//7
public List<Ville> TrierParPopulation(string ordre)
{
    List<Ville> villes = new List<Ville>();

    villes.AddRange(Villes);

    if (ordre.ToLower() == "croissant")
        villes.Sort(new PopulationCroissantComparer());
    else if (ordre.ToLower() == "décroissant")
        villes.Sort(new PopulationDécroissantComparer());

    return villes;
}

//8
public int PopulationTotale()
{
    int populationTotale = 0;

    foreach (Ville ville in Villes)
        populationTotale += ville.Population;

    return populationTotale;
}

//7(suite)
class PopulationCroissantComparer : IComparer<Ville>
{

```

2 pts

3 pts

3 pts

4 pts

5 pts

3 pts

5 pts

| Filière | Epreuve                             | Session      | 6/8 |
|---------|-------------------------------------|--------------|-----|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |     |

```

    public int Compare(Ville x, Ville y)
    {
        return x.Population.CompareTo(y.Population);
    }
}

//7(suite) 5 pts
class PopulationDécroissantComparer : IComparer<Ville>
{
    public int Compare(Ville x, Ville y)
    {
        return -x.Population.CompareTo(y.Population);
    }
}

```

---

```

public partial class Gestion_des_pays : Form
{
    //9.a 1 pt
    List<Pays> LesPays = new List<Pays>();

    public Gestion_des_pays()
    {
        InitializeComponent();
    }

    //9.b 1 pt
    private void Gestion_des_pays_Load(object sender, EventArgs e)
    {
        //9.b.i 6 pts
        Pays maroc = new Pays("Maroc");
        maroc.Ajouter(new Ville("Casablanca", 3359000));
        maroc.Ajouter(new Ville("Rabat", 577000));
        maroc.Ajouter(new Ville("Fès", 1112000));
        maroc.Ajouter(new Ville("Tanger", 974000));
        maroc.Ajouter(new Ville("Marrakech", 928850));
        maroc.Capitale = maroc.Villes[1];
        LesPays.Add(maroc);

        Pays tunisie = new Pays("Tunisie");
        LesPays.Add(tunisie);

        Pays jordanie = new Pays("Jordanie");
        jordanie.Ajouter(new Ville("Zarqa", 395227));
        jordanie.Ajouter(new Ville("Irbid", 250645));
        jordanie.Capitale = new Ville("Amman", 994199);
        LesPays.Add(jordanie);

        //9.b.ii 3 pts
        comboBox_Pays.DataSource = LesPays;
        comboBox_Pays.DisplayMember = "Nom";
    }

    //9.c 4 pts
    private void comboBox_Pays_SelectedIndexChanged(object sender, EventArgs e)
    {
        Pays pays = comboBox_Pays.SelectedItem as Pays;

        listBox_Villes.Items.Clear();

        if (pays.NombreVilles != 0)
        {

```

| Filière | Epreuve                             | Session      | 7/8 |
|---------|-------------------------------------|--------------|-----|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |     |

```

        foreach (Ville ville in pays.Villes)
        {
            //9.c.i
            if (pays.Capitale != null && ville.Nom == pays.Capitale.Nom)
                listBox_Villes.Items.Add("--> " + ville);
            else
                listBox_Villes.Items.Add(ville);
        }
    }
    else
    {
        //9.c.ii
        listBox_Villes.Items.Add("Aucune ville !!!");
    }

//9.d
private void button_Trier_Click(object sender, EventArgs e)
{
    Pays pays = comboBox_Pays.SelectedItem as Pays;

    if (pays.NombreVilles != 0)
    {
        string ordre = "croissant";
        List<Ville> villes;

        if (radioButton_Décroissant.Checked) ordre = "décroissant";

        villes = pays.TrierParPopulation(ordre);

        listBox_Villes.Items.Clear();

        foreach (Ville ville in villes)
        {
            if (pays.Capitale != null && ville.Nom == pays.Capitale.Nom)
                listBox_Villes.Items.Add("--> " + ville);
            else
                listBox_Villes.Items.Add(ville);
        }
    }
}

//9.e
private void button_OrdreInitial_Click(object sender, EventArgs e)
{
    comboBox_Pays_SelectedIndexChanged(comboBox_Pays, new EventArgs());
}

//9.f
void Gestion_des_pays_FormClosing(object sender, FormClosingEventArgs e)
{
    if (MessageBox.Show("Voulez-vous vraiment fermer l'application ?",
        "Attention",
        MessageBoxButtons.YesNo,
        MessageBoxIcon.Question) == DialogResult.No)
        e.Cancel = true;
}
}

```

| Filière | Epreuve                             | Session      | 8 / 8 |
|---------|-------------------------------------|--------------|-------|
| TDI     | Synthèse - V1 - Eléments de réponse | Juillet 2018 |       |