

BANQUE D'EPREUVES DUT-BTS

- SESSION 2006 -

**EPREUVE
D'INFORMATIQUE**

CODE EPREUVE : BE-INFO

CALCULATRICE INTERDITE

L'épreuve comporte 20 questions regroupées suivant les thèmes : logique, électronique numérique, structure des machines, programmation, réseaux, graphes.

DUREE : 2H30

Logique

Question 1

Soient p et q deux propositions

La proposition 'si p alors q' se note ' $p \Rightarrow q$ '. La connectique logique \Rightarrow s'appelle implication.

Sa table de vérité est :

(A)

p	q	$p \Rightarrow q$
0	0	1
0	1	0
1	0	0
1	1	1

(B)

p	q	$p \Rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

Les résultats suivants sont-ils corrects ?

(C) $(p \Rightarrow q) \Leftrightarrow (\bar{p} \text{ ou } q)$

(D) $(p \Rightarrow q) \Leftrightarrow (\bar{q} \Rightarrow \bar{p})$

(E) $(p \Rightarrow q) \Leftrightarrow p \text{ et } \bar{q}$

Electronique numérique

Question 2

Soient deux nombres A et B codés sur 16 bits. Pour le nombre A, on appelle a_i le bit de rang i, avec i compris entre 0 et 15, a_0 étant le bit de poids fort. La notation est similaire pour B (b_j).

(A) Pour un codage en complément à deux, on a : $A = -2^0 \cdot a_0 + \sum_{k=1}^{15} a_k \cdot 2^k$

(B) Soit un convertisseur analogique numérique fournissant un résultat sur 16 bits en binaire décalé. Pour obtenir un nombre codé en complément à deux, il suffit de prendre le complément du bit de poids fort.

(C) Pour un codage en complément à deux, A est compris entre -32768 et $+32768$.

On appelle codage Q1.15 un codage en virgule fixe, la virgule étant placée après le premier bit depuis la gauche. Les poids affectés aux différents bits du nombre binaire sont donc :

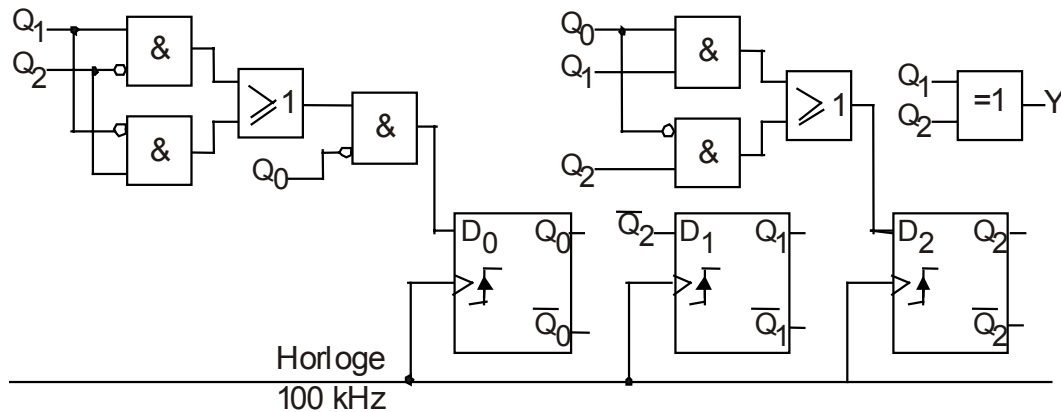
-2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}	2^{-11}	2^{-12}	2^{-13}	2^{-14}	2^{-15}
--------	----------	----------	----------	----------	----------	----------	----------	----------	----------	-----------	-----------	-----------	-----------	-----------	-----------

(D) Avec une telle représentation, on peut coder un nombre réel compris entre -1 et $+1$.

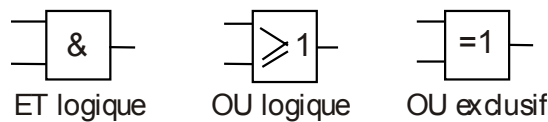
- (E) La somme de +0,0625 avec +0,125 codés en Q1.15 donne le nombre binaire 0001100000000000, soit 1800 en hexadécimal.

Question 3

Soit un système logique conçu à partir de trois bascules D sensibles sur front montant de l'horloge.



La signification des symboles est la suivante :



Les bascules sont initialisées telles que $Q_2Q_1Q_0 = 010$.

- (A) Ce système séquentiel fonctionne suivant un cycle de cinq états.
 (B) La fréquence du signal Q_0 est égale à 50 kHz.
 (C) Le rapport cyclique du signal Q_2 est égal à $\frac{1}{2}$.
 (D) La valeur moyenne du signal Y est égale à $\frac{2}{3} \cdot V_{DD}$, V_{DD} étant l'amplitude des signaux logiques.
 (E) Si les bascules sont initialisées avec $Q_2Q_1Q_0 = 111$, il faudra deux fronts actifs de l'horloge pour revenir dans le cycle normal.

Structure des machines

Question 4

- (A) Le registre d'état d'un processeur contient la température de celui-ci.
 (B) Le registre d'instruction d'un processeur contient l'adresse de la prochaine instruction à exécuter.
 (C) Certains processeurs permettent d'exécuter des traitements en parallèle.
 (D) Le temps d'exécution d'une instruction est uniquement proportionnelle à la fréquence de l'horloge associée.
 (E) L'Unité Arithmétique et Logique contient toutes les instructions d'un programme à exécuter.

Question 5

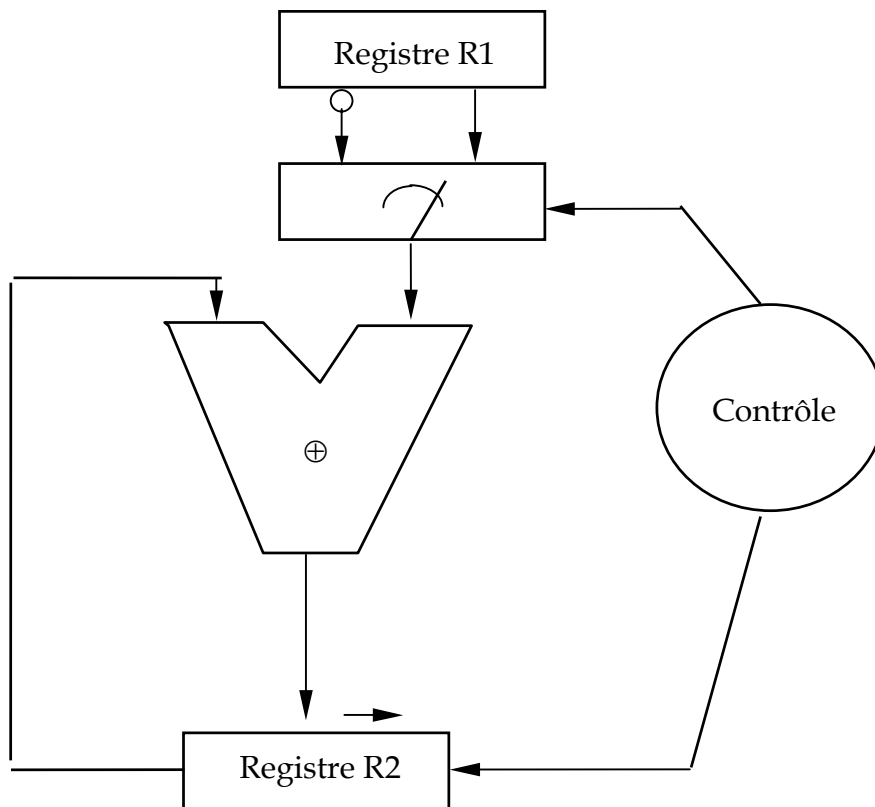
- (A) Les barrettes de mémoire ECC (*Error Correction Coding*) sont des mémoires capables d'auto corriger d'éventuelles erreurs internes.
- (B) Les barrettes de mémoire de type SDRAM possèdent leur propre horloge, d'où des accès plus rapides.
- (C) Il faut un bus d'adresse plus large pour adresser une mémoire de 256 koctets en mots de 16 bits que la même mémoire en mots de 8 bits.
- (D) Une mémoire dynamique peut perdre son contenu si elle n'est pas rafraîchie
- (E) Le BIOS d'un ordinateur est contenue dans une mémoire de type SDRAM.

Question 6

Soit la structure suivante construite autour d'un additionneur N bits.

R1 est un registre avec possibilité de sortie inversée.

R2 est un registre qui peut fonctionner en registre à décalage à droite.



- (A) En présentant successivement deux nombres à $t=0$ nombre A dans le registre R1, $t=1$ nombre B dans le registre R1, le registre R2 contient leur somme.
- (B) En présentant successivement deux nombres à $t=0$ nombre A dans le registre R1, $t=1$ nombre B dans le registre R1, le registre R2 contient leur différence.
- (C) En présentant les nombres suivants à l'entrée du registre R1
 $t=0 \rightarrow (0110)_2$ à $t=1 \rightarrow (0011)_2$ à $t=2 \rightarrow (0101)_2$
la sortie du registre R2 à $t=3$ est $(1001)_2$
- (D) Cette structure peut réaliser la multiplication de deux nombres en complétant la partie contrôle.
- (E) Cette structure peut réaliser la division de deux nombres.

Question 7

- (A) Une mémoire cache est une mémoire qui permet d'accélérer l'accès aux données.
- (B) La technologie utilisée pour ce type de mémoires est différente du reste de la mémoire vive de l'ordinateur.
- (C) Le principe de fonctionnement des caches est de stocker plus d'informations que celles demandées
- (D) La mémoire cache porte ce nom car elle permet d'occulter certaines données
- (E) Un disque dur peut servir de mémoire cache.

Question 8

Soit l'addition suivante en notation binaire complément à deux (représenté en hexadécimal). Les opérandes et le résultat sont codés sur 16 bits :

$$\begin{array}{rcccc} & A & D & C & F \\ + & 8 & F & 7 & A \\ \hline & - & - & - & - \end{array} \quad \begin{array}{cccc} C & V & N & Z \\ - & - & - & - \end{array}$$

où C est l'indicateur de retenue
V est l'indicateur de débordement
N est l'indicateur de signe
Z est l'indicateur de résultat nul

- (A) Le résultat de l'opération peut être représenté sur 16 bits en notation complément à 2
- (B) l'indicateur C vaudra 1 après l'opération
- (C) l'indicateur V vaudra 0 après l'opération
- (D) l'indicateur N vaudra 1 après l'opération
- (E) l'indicateur Z vaudra 0 après l'opération

Programmation

Question 9

Soit le programme suivant écrit en langage C

```
#include <stdio.h>

void fonct(int tab[ ], int lg, int *m, int *n)
{
int i;
*m=tab[0];
*n=tab[0];
for (i=1;i<lg;i++)
{ if (*m<tab[i]) *m=tab[i];
if (*n>tab[i]) *n=tab[i];
};
}
```

```

int main(void)
{
    int x,y ;
    int tableau[10]={-56,4,128,45,57,254,1024,5,-4,10};

    fonct(tableau, 10, &x, &y) ;
    printf(" x=%d\n",x);
    printf(" y=%d\n",y);
    return 0;
}

```

- (A) Ce programme comporte des erreurs de syntaxe qui ne lui permette pas d'être compilé.
 (B) La fonction « *fonct* » peut aussi s'écrire :

```

void fonct(int *t, int lg, int *m, int *n)
{
    int i;
    *m=*t;
    *n=*t;
    for (i=1;i<lg;i++)
        { if (*m<*(t+i)) *m=*(t+i);
          if (*n>*(t+i)) *n=*(t+i);
        };
}

```

- (C) Si le programme peut se compiler et s'exécuter, il affichera sur un écran :
 x=1024
 y=-56
 (D) Si le programme peut se compiler et s'exécuter, il affichera sur un écran :
 x=1024
 y=4
 (E) La fonction « *fonct* » peut aussi s'écrire :

```

int fonct(int tab[ ], int lg, int *m)
{
    int i,k;
    *m=tab[0];
    *n=tab[0];
    for (i=1;i<lg;i++)
        { if (*m<tab[i]) *m=tab[i];
          if (k>tab[i]) k=tab[i];
        };
    return k ;
}

```

Question 10

Le langage C est un langage typé. La déclaration, l'initialisation ainsi que l'allocation de la mémoire de chaque variable d'un programme sont donc réglementées.

- (A) Les variables doivent être déclarées dans chaque bloc ou fonction du programme.
 (B) Le type détermine la taille de la place mémoire réservée pour la variable uniquement.
 (C) Une variable de catégorie statique est initialisée à chaque entrée dans le bloc de la fonction où elle est déclarée.

- (D) Pour allouer dynamiquement en mémoire des objets pendant l'exécution d'un programme il faut utiliser l'instruction « *new* ».
- (E) Le code ci-dessous implique que $*(ptab+i)$ est équivalent à $tab[i]$.
- ```
int *ptab, tab[10], i;
ptab=&tab[i];
```

### Question 11

Un système d'exploitation multiprocessus (UNIX par exemple) alloue à chaque processus la mémoire nécessaire à son exécution. Pour une application développée en langage C, la mémoire allouée est divisée en quatre zones ( le code, les données statiques, la pile, le tas)..

- (A) La pile est utilisée pour le passage d'arguments entre fonctions.
- (B) Les variables globales sont stockées dans la pile.
- (C) Les variables locales aux blocs sont stockés dans le tas.
- (D) Les variables locales aux blocs sont stockées dans la pile.
- (E) Lors d'allocation dynamique les données sont stockées dans le tas.

### Question 12

Dans un système d'exploitation multiprocessus (UNIX)

- (A) Un processus peut-être suspendu par un autre processus.
- (B) Un processus père peut créer un processus fils.
- (C) Tous les processus ont une filiation à partir d'un processus initial
- (D) Sur une architecture mono-processeur plusieurs processus peuvent être actifs à un instant donné.
- (E) Un utilisateur quelconque peut arrêter l'exécution de n'importe quel processus.

### Question 13

Soit l'application suivante écrite en langage C++ :

```
#include <iostream>
using namespace std;

const int Max=20;
class yyyy_entier
{
 int dimension;
 int *adr;
 int nelem;
```

public:

```
 yyyy_entier (int n = Max)
 {
 dimension=n;
 adr= new int [n];
 nelem=0;
 }
 ~yyyy_entier()
 {
 delete adr;
 }
 void empile (int p)
 { if (nelem < dimension) adr[nelem++]=p;}

 int depile ()
 { if (nelem>0) return adr[--nelem];
 else return 0;
 }
 int pleine ()
 { return (nelem ==dimension);}

 int vide ()
 { return (nelem ==0);}

 yyyy_entier (yyyy_entier &);
};
```

```
yyyy_entier::yyyy_entier(yyyy_entier &y)
{
 int i;

 adr=new int [dimension = y.dimension];
 nelem=y.nelem;
 for (i=0; i<nelem;i++) adr[i]=y.adr[i];
}
```

```
int main(void)
```



```

{
int i;
yyyy_entier yy(5);
yy.empile(23);
yy.empile(72);
yy.empile(35);
yy.empile(125);
yy.empile(1250);
yyyy_entier xx=yy;
cout << "Contenu de xx: ";
while (!xx.vide()) cout << xx.depile()<< " ";
cout << endl;
return 0;
}

```

- (A) `yyyy_entier (int n = Max)` est la méthode constructeur de la classe `yyyy_entier`
- (B) `xx` est une variable de la fonction « `main` »
- (C) `~yyyy_entier()` correspond à une erreur de syntaxe
- (D) Ecrire `cout << xx.nelem` est une erreur
- (E) L'instruction « `new` » appelle le constructeur de la classe `int`

#### Question 14

```

yyyy_entier::yyyy_entier(yyyy_entier &y)
{
 int i;

 adr=new int [dimension = y.dimension];
 nelem=y.nelem;
 for (i=0; i<nelem;i++) adr[i]=y.adr[i];
}

```

- (A) Cette méthode est le constructeur par recopie de la classe `yyyy_entier`
- (B) Cette méthode n'appartient pas à la classe `yyyy_entier`
- (C) L'argument de cette méthode est transmis par valeur
- (D) Cette méthode crée une nouvelle instance de la classe `yyyy_entier`
- (E) L'écriture de cette méthode est superflue

#### Question 15

- (A) Le langage JAVA permet l'héritage multiple.
- (B) L'instruction « `new` » en JAVA appelle le constructeur d'une classe
- (C) Dans le langage JAVA les tableaux multidimensionnels n'existent pas
- (D) Une « interface » dans le langage JAVA ne contient que des méthodes abstraites
- (E) Le langage JAVA est un langage procédural.

## Réseaux

### Question 16

Le protocole IP (Internet Protocol) :

- (A) Effectue de la commutation de paquets en mode connecté
- (B) Assure le routage de datagrammes
- (C) Peut effectuer de la segmentation et du réassemblage de datagrammes
- (D) Peut fonctionner dans un réseau étendu dans lequel le protocole ICMP (Internet Control Message Protocol) n'est pas implanté.
- (E) N'a pas été conçu pour supporter des flux de données isochrones

### Question 17

Soit un protocole CSMA/CD utilisé sur un support de type 100 base T. On impose la "Tranche canal" (égale à 2 fois le délai de propagation) à 512 temps bits, et la longueur maximale d'un câble entre deux répéteurs égale à 100 mètres. On suppose que le temps de traversée d'un répéteur est égal à  $0,6 \mu\text{s}$  et que la vitesse de propagation du signal sur le câble est de  $4\text{ns/m}$ . Quel est dans ce cas le nombre maximal admissible de répéteurs ?

- (A) 2
- (B) 4
- (C) 8
- (D) 16
- (E) 32

### Question 18

Le protocole de transport UDP:

- (A) Est fonctionnellement équivalent au protocole X25
- (B) Permet d'obtenir un meilleur temps de réponse que TCP sur un réseau IP
- (C) Est incompatible avec IPV6
- (D) Sera utilisé préférentiellement par rapport à TCP pour transmettre des flux de données isochrones sur un réseau IP
- (E) Assure un contrôle de flux

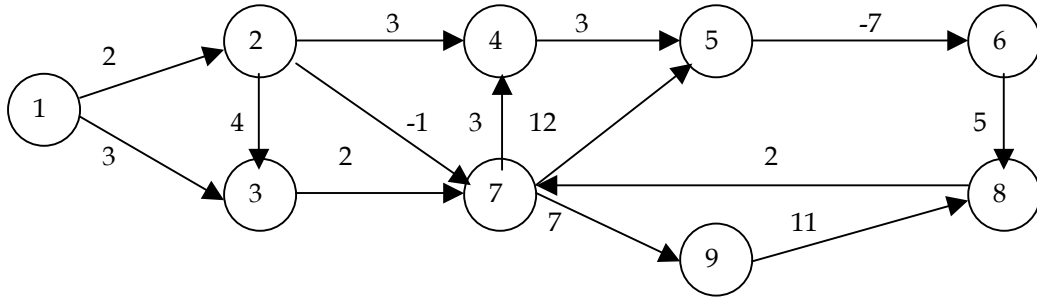
## Graphes

### Question 20

- (A) Un graphe est toujours planaire.
- (B) Dans un graphe orienté, tout sommet est son propre successeur.
- (C) Un graphe peut être formé de plusieurs composantes connexes.
- (D) Un sommet a toujours soit un successeur soit un prédécesseur.
- (E) Un graphe complet est connexe.

### Question 20

Soit le graphe valué suivant :



- (A) Ce graphe contient 3 cycles.
- (B) Le plus court chemin du sommet 1 au sommet 8 est (1,3,7,9,8) et vaut 23.
- (C) Le plus long chemin du sommet 1 au sommet 7 vaut 7.
- (D) Le plus court chemin du sommet 6 au sommet 5 vaut 19.
- (E) Le plus court chemin du sommet 2 au sommet 4 vaut 2.