

Trimestre Été, 2008

Mohamed Lokbani

IFT1166 – Examen Intra –

Inscrivez tout de suite votre nom et le code permanent.

Nom: _____ | Prénom(s): _____ |

Signature: _____ | Code perm: _____ |

Date : mercredi 4 juin 2008

Durée : 2 heures (de 17h30 à 19h30)

Local : Z-317 ; Pavillon Claire McNicoll

Directives:

- Toute documentation permise.
- Calculatrice **non** permise.
- Répondre directement sur le questionnaire.
- Les réponses **doivent être brèves, précises, claires et nettement présentées.**

1. _____ /08 (1.1, 1.2, 1.3, 1.4)

2. _____ /20 (2.1, 2.2, 2.3, 2.4)

3. _____ /12 (3.1, 3.2, 3.3, 3.4)

4. _____ /20 (4.1, 4.2)

5. _____ /20 (5.1, 5.2)

6. _____ /20 (6.1)

Total: _____ /100

Directives officielles

* Interdiction de toute communication verbale pendant l'examen.

* Interdiction de quitter la salle pendant la première heure.

* L'étudiant qui doit s'absenter après la première heure remettra sa carte d'étudiant au surveillant, l'absence ne devant pas dépasser 5 minutes. Un seul étudiant à la fois peut quitter la salle.

* Toute infraction relative à une fraude, un plagiat ou un copiage est signalée par le surveillant au directeur de département ou au professeur qui suspend l'évaluation.

F.A.S

Exercice 1 (08 points) Encercler la bonne réponse et donnez une courte explication.

1.1 [Vrai | Faux] Le langage C++ est apparu après le langage Java ?

1.2 [Vrai | Faux] Pour utiliser le manipulateur de flux « setprecision() », il faudra inclure le fichier d'en-tête « iostream » ?

1.3 [Vrai | Faux] L'expression à évaluer dans l'instruction de contrôle « switch » peut avoir n'importe quel type ?

1.4 [Vrai | Faux] Nous ne pouvons passer un paramètre à une fonction que de deux manières différentes ?

Exercice 2 (20 points) Tout ce qui suit est syntaxiquement correct (compile et s'exécute correctement)

2.1 À l'aide d'un exemple, expliquez le rôle joué par la macro suivante :

```
01 | #define A(x) ((x==0)?0:1)
```

2.2 Tout en détaillant votre réponse, que va afficher en sortie le fragment suivant :

```
01 | enum Semaine {Lu, Mar, Mer, Je, Ve, Sa, Di};
02 | Semaine a[] = { Mer, Je, Mar, Di, Ve };
03 | for (int i = 0; i < 5; i++)
04 |     a[i] = a[a[i]];
05 | for (int i = 0; i < 5; i++)
06 |     cout << "a[" << i << "] = " << a[i] << endl;
```

2.3 À l'aide d'un exemple, quel est le rôle joué par cette macro ?

```
01 | #define B(x,i) ((x)|(1<<(i)))
```

2.4 Tout en détaillant votre réponse, à quoi sert la fonction suivante :

```
01 | int A(const char *s1, const char *s2) {
02 |     while ((*s1 != '\0') && (*s1 == *s2)) {
03 |         s1++; s2++;
04 |     }
05 |     return *s1 - *s2;
06 | }
```

Exercice 3 (12 points) Soit « x » un nombre non signé initialisé avec la valeur « 5 ». Ce nombre est représenté comme suit sur 1 octet : « 00000101 ». Tout en développant votre réponse, dites ce que vont afficher en sortie les fragments de code suivants :

3.1 `unsigned char c = 5;`
`(c << 2) << endl;`

3.2 `unsigned char c = 5;`
`unsigned char x = ~c;`
`cout << (int) x << endl;`

3.3 `unsigned char c = 5;`
`cout << (c & 2) << endl;`

3.4 `unsigned char c = 5;`
`cout << (c | 12) << endl;`

Exercice 4 (20 points) Tout en développant votre réponse, que vont afficher en sortie les fragments de code suivants qui compilent et s'exécutent correctement (dans un programme complet).

4.1

```
01 | char s[]="CHAINE", *p=s+3;
02 | cout << p << endl;
03 | cout << p[-1] << endl;
04 | cout << --*++p << endl;
```

4.2

```
01 | int a[4]={10,20,30,40};
02 | cout << a[1/2] << endl;
03 | int* ptr=a+1;
04 | cout << a-ptr << endl;
05 | cout << ptr[1] << endl;
06 | const int& r = *ptr;
07 | cout << r+1 << endl;
08 | cout << --*++ptr << endl;
```

Exercice 5 (20 points) Le programme suivant compile correctement. Mais lors de son exécution, il plante royalement et affiche en sortie le message suivant « Segmentation fault (core dumped) ». Ce message correspond à un incident de segmentation.

5.1 Expliquez brièvement l'une des causes d'une telle erreur.

5.2 Tout en développant votre réponse, corrigez les différentes erreurs conceptuelles pour que le programme puisse s'exécuter correctement et faire ce qu'il est supposé i.e. déclarer une structure, initialiser les éléments de la structure et afficher ces éléments en sortie. Vous pouvez inclure de nouvelles instructions si vous le jugez nécessaire. Dans ce cas, précisez entre quelles lignes.

```
01 #include <iostream>
02
03 using namespace std;
04
05 typedef struct {
06     char *nom;
07     char *prenom;
08     int age;
09 } type_fiche;
10
11 int main(){
12     type_fiche *fiche=NULL;
13
14     fiche->nom = new char;
15
16     strcpy(fiche->nom, "Sasseur");
17
18     strcpy(fiche->prenom, "Marlene");
19
20     fiche->age = 3;
21
22     cout << "Nom: " << fiche->nom << endl;
23     cout << "Prenom: " << fiche->prenom << endl;
24     cout << "Age: " << fiche->age << endl;
25
26     return 0;
27 }
```


Exercice 6 (20 points) Écrivez un programme C++ qui réalise ce qui suit :

- 1- Il inverse d'abord l'ordre des arguments entrés sur la ligne de commandes sauf l'argument « 0 » qui représente le nom du programme. Par exemple si la ligne de commandes contenait la suite [arg1, arg2, arg3, arg4], le résultat sera [arg4, arg3, arg2, arg1].
- 2- Il inverse par la suite l'ordre des caractères pour chaque argument. Par exemple si l'argument 3 contenait la chaîne « exemple », le résultat sera « elpmexe ».
- 3- Une fonction « main » pour tester ces fonctionnalités.

Vous ne pouvez utiliser aucune fonction de la librairie « C » (par extension « C++ ») permettant de manipuler les chaînes de caractères. Vous devez réinventer la roue s'il le faut !

