29 octobre 1996

UNIVERSITÉ DU QUÉBEC À HULL
DÉPARTEMENT D'INFORMATIQUE

# TECHNIQUES DE PROGRAMMATION I
# INF4033

EXAMEN INTRA

**Directives:**

1. C'est un examen à livre ouvert.
2. Répondre directement dans ce formulaire.
3. Utiliser le verso de page comme brouillon.
4. L'examen dure trois heures.
5. Le barème est montré au debut de chaque question.
6. Questions éventuelles doivent être posées durant les 30 premières minutes de l'examen.
7. Circulation est interdite (y compris les sorties à la toilette).

**Nom, prénom:**_______________________________________________

**Code permanent:**_______________________________________________

**Signature:**_______________________________________________

**Numéro du formulaire:**_______________________________________________

| Ex.# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Σ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Note | | | | | | | | | | | |
| sur | 8 | 12 | 6 | 8 | 13.5 | 11 | 11 | 10.5 | 10 | 10 | 100 |

**Ex. #1 (8 pts)**

Pour chacune des expressions en C, réécrire l'expression en ajoutant les parenthèses qui reflètent l'ordre de l'exécution des opérations:

a) a = - b * c + b || a

b) c = b -= a + b >= a / b * - a - c && a

**Ex. #2 (12 pts)**

Pour chacune des expressions de l'ex. #1, produire son arbre de décomposition.

a)

b)

**Ex. #3 (6 pts)**

Soient les déclarations suivantes:

```
int a,b;
double c;
```

Pour chacune des expressions de l'ex. #1, déterminer son type.

a)

b)

**Ex. #4 (8 pts)**

Soient les déclarations

```
int a,b;
double c;
```

et les initialisations

```
a = 4;
b = 9;
c = 7.6;
```

Pour chacune des expressions de l'ex. #1, déterminer sa valeur.

a)

b)

**Ex. #5 (13.5 pts)**

Parmi les déclarations qui suivent encercler celles qui ne sont pas valables en C.

a) int SWITCH;
b) int 3_petits_cochons;
c) int RomeoEtJuliette;
d) int scanf;
e) int define;
f) double 3.14;
g) int xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx;
h) double printf("%f",c);;
i) Char fin_de_ligne;

**Ex. #6 (11 pts)**

Donner un fragment du programme qui calcule la somme de tous les nombres entiers positifs entre 0 et 300 qui ont ‘3’ comme dernièr chiffre, c’est-à-dire

3 + 13 + 23 + 33 + . . . + 283 + 293

**Ex. #7 (11 pts)**

Donner la sortie du programme suivant:

```
main()
{
        int i, j;

        for (i = 1; i < 10; i+=3) {
                j = 4;
                while (j <= 7) {
                        printf(" i = %d; j = %d\n", i, j);
                        j++;
                }
        }
}
```

**Ex. #8 (10.5 pts)**

Déterminer l'effet de chaque appel de l’énoncé printf qui suit, en supposant que la valeur de la variable c = -34.24567, et la valeur de la variable a = 9876. Écrivez dans les cases prévues à cet effet, un caractère par case.

a) printf ("a=%d;", a);

b) printf ("a=%6d;", a);

c) printf ("a=%-8d;", a);

d) printf ("c=%f;", c);

e) printf ("c=%8.3f;", c);

f) printf ("c=%-8f;", a);

g) printf ("c=%e;", c);

**Ex. #9 (10 pts)**

Soit le programme suivant:

```
/* Ce programme affiche les diviseurs du nombre introduit a partir du clavier */

/* 1 */   #include <stdio.h>
/* 2 */   #include "genlib.h"
/* 3 */   #include "simpio.h"
/* 4 */
/* 5 */   main()
/* 6 */   {
/* 7 */       int nbreDiviseurs, nombre;
/* 8 */
/* 9 */       printf ("nombre = ");
/* 10 */      nombre = GetInteger;          /* introduire nombre
/* 11 */      nbreDiviseurs = 0;            /* initialiser a zero */
/* 12 */      for (i==1; i<=nombre; i++) {
/* 13 */          if (nombre % i == 0) {
/* 14 */              printf ("%d est un diviseur de %d", nombre, i);
/* 15 */              nbreDiviseurs++;
/* 16 */              }
/* 17 */      };
/* 18 */      printf ("nombre %d possede %d diviseurs\n", nombre, nombreDiviseurs);
/* 19 */      nombre = 0;
/* 20 */  }
```

Énumérer les erreurs (syntaxiques, logiques et méthodologiques de ce programme).
NB: Les numéros au debut de chaque ligne servent à identifier les lignes auxquelles vous allez faire référence dans votre liste d’erreurs.

**Ex. #10 (10 pts)**

Donner un autre fragment du programme qui est équivalent au fragment du programme de l'Ex. #7. Utiliser une boucle **while** à la place de la boucle **for** du fragment de l.Ex #7 et une boucle **for** à la place de sa boucle **while.**