

Mid-Term Test
Theoretical Part

EEE243 Applied Computer Programming
22 October 2015, 08:00 – 10:50 hrs

Examiner: W. Greg Philips, CD, PhD, PEng

Instructions:

- **Do not turn this page until instructed to do so.**
- You have 170 minutes to complete the test.
- Questions have the values indicated in the centre column.
- This test has two parts.
 - The *theoretical* portion of the test is closed-book, out of a total of 20 marks.
 - The *practical* portion of the test is open-book (you may have course notes and textbook and you may access the internet), and is worth 30 marks.
- You must hand in the *theoretical* exam booklet before you can begin the *practical* part of the test.
- Answer all questions in the test booklet.
- Immediately fill out your name, college number and the number of your workstation on the information card.
- If a question seems unclear, make a *reasonable* assumption, document it, and answer the question as though the assumption were correct. *The examiners will not clarify the meaning of questions during the test.*
- Good luck!

Test de mi-session
Partie théorique

GEF243 Programmation informatique appliquée
22 octobre 2015, 08h00–10h50

Examinateur: Capt Adrien Lapointe, CD, MSc

Instructions:

- **Ne tournez pas cette page avant l’instruction de l’examinateur.**
- Vous avez 170 minutes pour compléter le test.
- Les questions ont les valeurs indiquées dans la colonne centrale.
- Ce test a deux parties.
 - La partie *théorique* est à livre fermé et compte pour 20 points.
 - La partie *pratique* est à livre ouvert (vous avez droit à vos notes et à votre livre, et vous pouvez accéder à l’Internet), et compte pour un total de 30 points.
- Vous devez remettre le livret d’examen *théorique* avant de pouvoir commencer la partie pratique.
- Répondez à toutes les questions dans le livret d'examen.
- Écrivez immédiatement votre nom, numéro de collège et numéro de station de travail sur la carte d’information.
- Si une question ne vous semble pas claire, faites des suppositions raisonnables, documentez-les et répondez à la question en tenant compte des suppositions. *Les examinateurs ne clarifieront pas le sens des questions pendant le test.*
- Bonne chance!

Beginning of THEORETICAL Portion of Test

1. A type in C provides two different things. What are those two things?
2. Explain why derived types are used in C.
3. It is possible to declare a character string as either a pointer or an array.
 - a. Explain the difference between these two types of declarations.
 - b. Explain the difference between physical storage and logical storage, as these terms relate to character strings.
4. Explain the difference between passing a parameter by value and by reference. Describe a situation where passing a parameter by reference is advantageous.
5. Consider the following code.

```

1  #include <stdio.h>
2  #include <string.h>
3
4  void reverse(char *str);
5
6  int main(void) {
7      char str[] = "Hello World!";
8      reverse(str);
9      printf("%s\n", str);
10     return 0;
11 }
12
13 void reverse(char *str) {
14     int x = strlen(str);
15     char temp_str[x + 1];
16     char *temp = str;
17
18     while (*temp != '\0') {
19         temp_str[x--] = *temp;
20         temp++;
21     }
22 }
```

The intent of the call to `reverse` on line 8 is that the contents of `str` will be reversed, i.e., the expected

Début de la partie THÉORIQUE du test

1. Un type en C nous donne deux choses. Quelles sont ces deux choses?
2. Expliquez pourquoi les types dérivés sont utiles en C.
3. Il est possible de déclarer une chaîne de caractères en tant que tableau et en tant que pointeur.
 - a. Expliquez la différence entre ces deux types de déclaration.
 - b. Expliquez la différence entre l'entreposage logique et physique des chaînes de caractères.
4. Expliquez la différence entre passez un paramètre par valeur et par référence. Donnez une situation où passer par référence est avantageux.
5. Considérez le code suivant.

Le but de l'appel à la fonction `reverse` à la ligne 8 est d'inverser le contenu de `str`. La sortie attendue à

output from line 9 is !dlrow olleH. There are two errors in the program. Identify the errors and suggest a correction for each.

la ligne 9 est !dlrow olleH. Il y a deux erreurs dans le programme. Identifiez-les et suggérez une correction pour chacune.

6. Explain how C modules can be used to implement information hiding.
7. Consider the following program which prints the path of a robot:

- 1 6. Expliquez comment les modules C peuvent être utilisés pour masquer l'information
- 5 7. Considérez le programme suivant qui affiche le parcours d'un robot:

```

1  #include <stdio.h>
2
3  #define DISTANCE_CALIB 10
4
5  int main(void) {
6      const int distance = 9;
7      float temps = 1;
8      float position = 0.0;
9      float step = 0;
10     float temps_avance = (distance / DISTANCE_CALIB) * temps;
11
12     while (step < temps_avance) {
13         printf("%0.2fs - position:%0.2f\n", step, position);
14         position += (10.0 * 0.1);
15         step += 0.1;
16     }
17     return 0;
18 }
```

- a. Are there errors in the code that would cause it to display incorrect information? If so, identify them and indicate their line numbers.
- b. Two kinds of constants are used in this program. What are they?
- c. Are there other places in the program where constants should be used? Where and why?

(2)

(2)

(1)

- a. Y-a-t-il des erreurs dans le code qui font en sorte qu'il n'affiche pas la bonne information? Si oui, identifiez-les et indiquez leurs numéros de ligne.
- b. Deux types de constantes sont utilisés ici, quels sont-ils?
- c. Y-a-t-il d'autres endroits où des constantes auraient dû être utilisées? Où et pourquoi?

End of THEORETICAL Portion of Test

Fin de la partie THÉORIQUE du test