

CIS*1500 W16 - Assignment #1

Due: Friday, January 29, 3pm.

Upload the first 6 questions in a typed document in .pdf or .txt format.

Upload your C program for question 7 separately.

1. [4 marks] Provide a series of printf statements to output the following:

- (a) **x "hi" x**
`printf ("x \"hi\" x\n");`
- (b) **x *hi* x**
`printf ("x *hi* x\n");`
- (c) **x \hi\ x**
`printf ("x \\hi\\ x\n");`
- (d) **x %hi% x**
`printf ("x %%hi%% x\n");`

2. [5 marks] True or False:

- (a) **Every C program must have a function called main.**
TRUE.
- (b) **Every C program must include the line #include <stdio.h>.**
TRUE; for our course, but not necessarily, in theory.
- (c) **Variable names are not case sensitive. Thus the identifiers radius and Radius refer to the same variable.**
FALSE.
- (d) **There is a limit to the maximum value that can be stored in the integer variable.**
TRUE.
- (e) **A variable represents a memory location used to store data.**
TRUE.

3. [5 marks] Which of the following are valid C identifier names for variables?

- (a) **HiThere** **YES.**
- (b) **double** **NO.**
- (c) **_score** **YES.**
- (d) **j&p** **NO.**
- (e) **last name** **NO.**
- (f) **triple** **YES.**

- (g) name123 **YES**. (k) 123name **NO**. (o) unless **YES**.
 (h) ALLCAPS **YES**. (l) john.smith **NO**. (p) iAMhere#8 **NO**.
 (i) single **YES**. (m) union **NO**.
 (j) under_score **YES**. (n) intersection **YES**.

4. [15 marks] True or False, the following statements follow valid C syntax:

- (a) **#DEFINE PI 3.14**
 FALSE; "define" must be lowercase.
 (b) **#define pi 3.145;**
 FALSE; "#define" doesn't require a concluding semicolon.
 (c) **c = a%b;**
 TRUE.
 (d) **a+b = c;**
 FALSE; "=" must be before the operation.
 (e) **a+=5;**
 TRUE.

5. [4 marks] Output Art: Provide a series of at least 5 printf statements to produce a visually pleasing output.

If your creativity is low, output the first 3 characters of your name.

```
printf (" DDDDDDDDD \n");
printf (" DDD      DDD \n");
printf (" DDD      DDD \n");
printf (" DDD      DDD      eeeee   vv      v \n");
printf (" DDD      DDD   ee      ee   vv      v \n");
printf (" DDD      DDD   ee      ee   vv      v \n");
printf (" DDD      DDD   eeeeeeeee   vv      v \n");
printf (" DDD      DDD   ee                      vv   v \n");
printf (" DDD      DDD   eee                      vv v \n");
printf (" DDDDDDDDD      eeeee           vv \n");
```

6. [8 marks] You are designing a new online census. As part of the interface, you wish to display how old the user is in terms of both years and months.

For example, if the current day is Jan 25, 2011 and the user is born on Jan 26, 2001, then you want to display:

"You are 9 years old and have been alive for 119 months". But currently you do not have any information about the user. How can you get and display this information?

(a) **What are the inputs and outputs required in this problem?**

Inputs: Current date, birth date;

Outputs: Years lived, months lived.

(b) **What are the sub-problems that should be considered when designing an algorithm?**

- Acquire the inputs.
- Separate dates into days, months, and years.
- Evaluate the differences between days, months, and years.
- Adjust differences for past birthdays.
- Output adjusted evaluations of years and months lived.

(c) **Design an algorithm (pseudocode) for this problem.**

- Acquire inputs (today's date, birth date).
- Determine years lived:
 - If the difference between current and birth month is less than or equal to zero, and the difference in current day and birth day is less than zero, evaluate the years lived as the difference in the current and birth year.
 - Otherwise, evaluate years lived as the difference in the current and birth year, minus one.
- Determine months lived:
 - If the difference in current and birth day is less than zero, then evaluate months lived as the difference in the years multiplied by 12 (convert units to months), and add to the difference in months.
 - Otherwise, evaluate months lived as the previous operation, minus 1.
- Output evaluated years and months lived to the user.
 - "You are *years_lived*. And have been alive for *months_lived*."

7. [10 marks] Write a C program to compute the area and perimeter of a rectangle. You should prompt the user to enter two integers: the length and the width.

The output should be formatted exactly as follows given the user inputs a length of 15 and width of 4:

The area of the rectangle is $15 \times 4 = 60$.

The perimeter of the rectangle is equal to $(2 \times 15) + (2 \times 4) = 38$.

Include appropriate comments.

A marking key for this program is provided separately in the Assignments folder.

Upload the C program into the Assignment 1 - Q7 folder using the Courselink Dropbox.

Your file must be named username_ass1.c where username is your University username.