GNG1106 Winter 2019 - Assignment 1

Available: Jan 13
Due: Jan 27, 11:59pm

Instructions

This assignment is to be done INDIVIDUALLY. Use the following instructions to complete and submit this assignment:

- You will need to submit your assignment electronically to Brightspace. Prepare the following:
  - An assignment file in PDF format (this allows you to use your favorite editor to create the PDF file). For question 1, insert the programming models for parts (a) and (b) filled in as per the question instructions. You may fill in the programming model using drawing features of your editor or by hand on paper which is then scanned and inserted into your document (be sure that the hand drawn diagram is legible). For Questions 2 and 3, insert in your assignment file the source code (take care in its appearance), and capture the output from running the program for all test cases. Also submit your **source code** files for questions 2 and 3.
- Place all your files (PDF file and C source code files) in a directory A1_xxxxxxxx where xxxxxxxx is your student number.
- Zip your PDF document and the C source files in a zip file with the name A1_xxxxxx.zip where xxxxxx is your student number.
- Submit the zip file before the assignment deadline via Brightspace. In Brightspace, navigate to the Assignment page and click on "Click to submit Assignment 1" to reach the assignment 1 submission folder. You can also select the Assignment tab to see the Assignment folder pages. The Brightspace video "Assignments" (found in the page https://documentation.brightspace.com/EN/le/assignments/learner/submit_assignments.htm) provides details to help you submit the zip file.
- The questions are provided in both PDF and Word files. You may use the Word file to enter your answers in the document. An rtf file is also provided so that you may edit the file with a word processor other than Word. Be sure to submit a **PDF** file.
- It is NOT permitted to use instructions such as branches and loops that have not yet been covered in the lectures.

➢ Do start the assignment soon and do **not** wait until the last minute. You will be more efficient with a number of smaller efforts over a few weeks before the deadline than one large effort just before the deadline.

Marking Scheme (total 20 marks)

- Question 1: 10 marks
- Question 2: 10 marks
- Question 3: 15 marks

Question 1 (10 marks)

1. (5 marks) The following programming model contains the indicated C program in its code memory. You will be showing how the working memory is used during the execution of this program and how the CPU evaluates the arithmetic expression that assigns a value to the variable **`u`**.

   Show the variables created in the working memory and how their values change during the execution of the program.

   - Show the values are assigned to the variables. Be sure to show all values that are assigned and replaced. Record successive assignments to variables/parameters as follows:

     Variable | ~~?~~ ~~2~~, ~~6~~, ~~4~~, 10 |

   - For the operation, show how contents from the working memory are moved to the CPU to calculate the value assigned to variable u. Show the results of **all** operations carried out in the CPU, that is, **one line** per operation.
   - Finally show the output of the program in the console window.

Code Memory

```
#include <stdio.h>

void main()
{
    double v1, v2, v3;
    v1 = 36.8;
    v2 = 4.6;
    v3 = (3.6/(4.3 + 0.4*v1))*v2;
    printf("Variable 1 = %f\n",v1);
    printf("Variable 2 = %f\n",v2);
    printf("Variable 3 = %f\n",v3);
}
```

Working Memory

CPU

Console

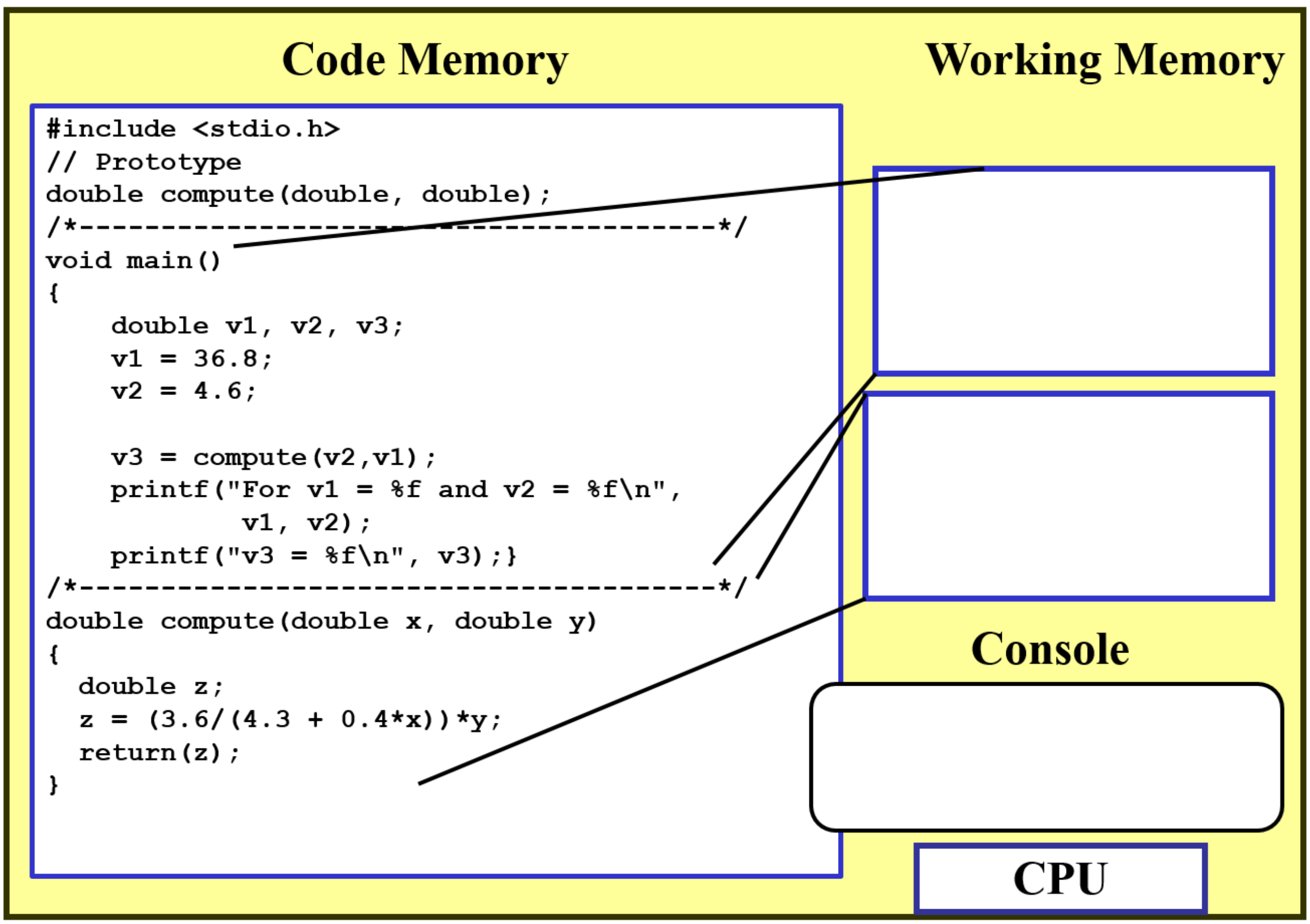
2. (5 marks) The following programming model contains in its code memory the indicated C program composed of 2 functions. You will be showing how the working memory is used during the execution of the two functions. Each piece of working memory is associated to a function using a pair of lines. (Note: the first pair of lines associates the piece of working memory allocated to the function **`main`** and the second pair of lines associates the piece allocated to the function **`compute`**).

Show how the variables (and parameters) are created in each piece of working memory during the execution of the program. It is **not** necessary to show how the operations are carried out in the CPU as in the case of part (a).

- Show the values are assigned to the variables. Be sure to show all values that are assigned and replaced. Record successive assignments to variables/parameters as follows:

Variable | ~~?~~ ~~2~~, ~~6~~, ~~4~~, 10 |

- Using arrows show how values are copied between the working memory allocated to the function **main** and the working memory allocated to the function **compute**.
- Finally show the output of the program in the console window.



Question 2 (10 marks)

You are involved in the development of an aircraft engine, the propfan or open rotor engine (see https://en.wikipedia.org/wiki/Propfan). A plane with mass 20 000 kg reaches a speed of 180 m/s where the engine applies a force of 40 000 newtons. When the pilot increases the engine's power to 60 000 newtons and the plane accelerates, the following equation gives the change in speed after the increase in power (a new speed is reached after about 120 s).

$$v = 0.00001t^3 - 0.00488t^2 + 0.75795t + 181.3566$$

where v is the plane's speed in m/s and t is the time in seconds ($t = 0$ when the engine's power is increased to 60 000 newtons).

Develop a program that computes the speed of the plane at a given time t. The user will provide the value of the time. Test your program with the following test cases:

Time t (s)	Speed *v* (m/s)
0.00	181.35660
2.00	182.85306
10.00	188.45810
50.00	208.30410
100.00	218.35160
120.00	219.31860

Guidelines:

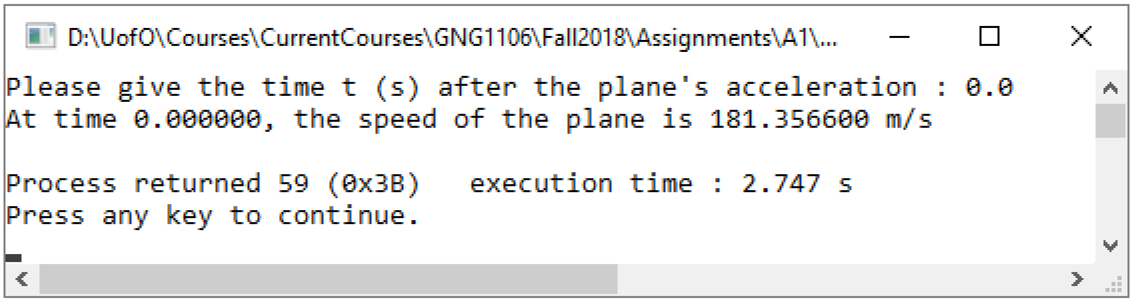
Logic/Strategies

- In the **`main`** function
  - Consider using the following local variables.
    - **`v`**: for the speed of the plane,
    - **`t`**: for the time *t*,
  - Prompt the user for the time *t*. Use short and appropriate prompt messages. Read the values from the keyboard and assign them respectively to the variable t.
  - Call a function, say **`calculateSpeed`**, to calculate the speed of the plane. Store the result returned by the function in the variable **`v`**. You will need to define this function in your program.
  - Display the results with a message of the following form:
    `At time 0.0, the speed of the plane is 181.3566 m/s`
- For the function **`calculateSpeed`**
  - Consider using the following parameters
    - **`t`**: for the value of time,
  - Consider using the following local variables
    - **`speed`**: for storing the speed of the plane at time t (note that this variable will contain the value returned).
  - The function calculates speed of the plane (see the equation given at the beginning of the question), stores it in the variable **`speed`** and returns the value stored in **`speed`**.

Test the program using the values provided in the table above. In your assignment file, give the output of your program for all test cases.

The answer to this question should provide:

1) The source code to your program (also insert the source code into the assignment file).
2) The output showing the results of all the test cases; insert the output into the assignment file. The following is an example of the output for the first test case.



Question 3 (15 marks)

The electrical resistance of a material depends on temperature. The resistance temperature coefficient, α, allows adjustment of the resistance of a material for a given temperature. In addition to the coefficient, the resistance of the material must be known at a standard temperature (normally 20 degrees Celsius). The resistance, R, of a material at a temperature T can be found using

$$R = R_{ref}\left[1 + \alpha(T - T_{ref})\right] \qquad (1)$$

where

R is the material's resistance in ohms at the temperature T in ^{o}C.
R_{ref} is the material's resistance at the reference temperature T_{ref} (normally at 20 ^{o}C)
α is the resistance temperature coefficient (/^{0}C) of the material
T is the material's temperature in ^{o}C
T_{ref} is the reference temperature (^{0}C) at which α is defined for the material.

(reference: https://www.allaboutcircuits.com/textbook/direct-current/chpt-12/temperature-coefficient-resistance/)

For metals, the coefficient is positive, which indicates that resistance increases as temperature increases. In the case of the elements carbon, germanium, and silicone, the coefficient is negative, which indicates that the resistance decreases as temperature increases. For certain metal alloys, the coefficient is almost zero and thus their resistance varies very little with change in temperature. The following table provides examples for the resistance temperature coefficient.

Material	Element/Alloy	α (/^{o}C)
Nickel	Element	0.005866
Iron	Element	0.005671
Tungsten	Element	0.004403
Aluminium	Element	0.004308
Copper	Element	0.004041
Steel (99.5% iron, 0.5% carbon)	**Alloy**	0.003
Constantan	**Alloy**	-0.000074
Silicon	Element	-0.075
Germanium	Element	-0.048

You are involved in a project for the development of electrical transformers which contains field coils. Field coils consist of a conductor wrapped around an iron core. Your role is to develop software which can calculate the conductor resistance of the coils at different temperatures.

The resistance of a coil is computed with the following steps

1. Using the resistance value per unit length, R_L (ohms/meter) and the length of the conductor, L (meters), compute the coil resistance at the reference temperature, T_{ref}, (20 ^{o}C) as $R_{ref} = R_L L$.
2. To adjust the resistance at the given temperature, T (^{o}C), equation 1 is used with a given resistance temperature coefficient, α (/^{o}C), at the reference temperature, T_{ref} (20 ^{o}C).

Apply the following guidelines to answer this question:

1) First develop a set of test cases (Excel is practical spreadsheet software that allows you to create test cases). Provide at least 5 test cases. Ensure that the test cases cover a wide range of values:
   a) Vary the resistance per unit length, R_L, between 0.00327 ohms/m and 3.55 ohms/m
   b) Use the coefficient values for aluminum and copper given in the above table.
   c) Vary the length of the conductor between 0.1 meters and 100 meters.

2) Develop your program using the GNG1106 C template (GNG1106template.c), that is, your program will contain a `main` function and a function that calculates the resistance of the field coil conductor. The `main` function contains instructions to get data from the user, calls the function to compute the resistance, and displays the results to the user. Be sure to document well your program and follow programming conventions.
3) A hint: The value 5.65 x 10^{-8} is represented in a C program with `5.65e-8`.
4) Include in your assignment report, a table of your test cases, your source code, and the captures output for all your test cases. Also submit your source code file.