16.070 Introduction to Computers and Programming

Due: 28 February  Problem Set 3  Spring 2001

All of the homework guidelines should be followed. Be sure to comment your code and make it readable. Points will be deducted if the guidelines are not followed. Indicate the amount of time you spent working on each problem.

Problem 1 (10%)
Assume that all of the following variables are of type int. Find the value of each of the following statements. In your answers indicate what order the operations are performed in.

\[ x = (2 + 3) \times 6; \]
\[ x = (12 + 6) / 2 \times 3; \]
\[ y = x = (2 + 3) / 4; \]
\[ y = 3 + 2 \times (x = 7 / 2); \]
\[ x = (\text{int}) 3.8 + 3.3; \]
\[ x = (2 + 3) \times 10.5; \]
\[ x = 3 / 5 \times 22.0; \]
\[ x = 22.0 \times 3 / 5; \]

Turn In: Typed answers to each of the above and the time you spent working on this problem.

Problem 2 (15%)
Write a program that will calculate the value of \( \pi \) using the following series expansion:

\[ \pi = -\frac{4}{3} + \frac{4}{5} - \frac{4}{7} + ... + (-1)^n \frac{4}{2n+1} + ... \]

Your program should prompt the user for the number of terms of the series to evaluate and display the calculated value of \( \pi \) before exiting. Implement the program first using a for-loop and then re-implement it using an equivalent while-loop (i.e. two different ways to solve the same problem). Note that \( n \) is an index starting at 0 for term 1.

Turn In: Your program code (for both cases) and a screen dump of an output run. Also include the time you spent working on this problem.

Problem 3 (20%)
Write a program for a cash register in Texastan. Texastan charges federal and state sales tax on products depending on their product code or class. The following table indicates the percentage of taxation for each product code:

<table>
<thead>
<tr>
<th>Code</th>
<th>Federal Tax</th>
<th>State Tax</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.5%</td>
<td>0%</td>
</tr>
<tr>
<td>1</td>
<td>2%</td>
<td>7%</td>
</tr>
<tr>
<td>2</td>
<td>0%</td>
<td>3%</td>
</tr>
<tr>
<td>3</td>
<td>0%</td>
<td>0%</td>
</tr>
</tbody>
</table>

The cash register should prompt a user for the price and product code of a product, then output the amount of federal and state sales tax to be charged for an item and add up the total cost of a product.
Typical output:

Welcome to Texastan
Please Enter Product Code: 0
Please Enter USD Price (before tax): 100.00

Unit Price: 100.0 USD
Federal Tax: 1.5 USD
State Tax: 0 USD
Total: 101.5 USD

Press any key

Turn In: Printout of source code and screen shot of a sample run for each product code for an item costing 100 USD before tax. Also turn in a flow chart describing your program, and indicate the amount of time you spent working on this problem.

Also: Copy your program code as username_PS3_P3.c to \CDIO-Prime\16.070HW\ps3\p3 , replacing username with your Athena username.

Problem 4 (15%)

A student writes a short program intended to compute the moment of inertia of a pendulum with mass concentrated at the end of an arm 10 meters long, using the formula:

\[ \text{Moment of Inertia} = \text{mass} \times (\text{arm length})^2 \]

Since various masses can be attached to the pendulum arm, the program prompts a user for an input from the set of numbers 1,2 and 3, signifying masses of 3, 5 and 8kg respectively at the end of the arm. The program then determines the moment of inertia using the selected mass. When running the program though, the moment of inertia printed to the screen always seems to turn out to be 0.

Identify the fundamental error that the student made, and explain how you would correct the problem.

```c
/*******************
* TJ Feb 2001 *
* Problem Set 3 *
* Question 4 *
*******************/

/* This program determines the moment of inertia of a pendulum */
#include <stdio.h>

/* Function Prototyping */
void determine_mass_from_configuration(int config, double total_mass);
determine_moment_of_inertia(double total_mass, double pendulum_length);

/* Global variables */
float total_mass=0;  /* Pendulum mass */
float pendulum_length=10;  /* Length of pendulum arm */
double moment_of_inertia;  /* Determined moment of inertia */

int main(void)
{
    /* Retrieve config info from operator */
    int config;
    printf("Choose Configuration (1 to 3): ");
    scanf("%d",&config);

    /* Determine mass according to configuration data received */
    determine_mass_from_configuration(config,total_mass);

    /* Calculate the moment of inertia */
    determine_moment_of_inertia(total_mass, pendulum_length);

    /* Output results to screen */
    printf("\n\nMoment of Inertia is: %f \n\n",moment_of_inertia);
```
Turn In: Typed answer.

Problem 5 (40%)
The following problem should be considered a requirement specification for a software design problem. You should follow the spiral model of the software development process. Be sure to practice top-down design techniques and follow proper modular programming guidelines.

An astronaut needs to pilot a lander module as it descends to Mars. Before he can start using his thrusters he needs to complete the following thruster start-up sequence:

1. Communicate intent to descend to Mars to an orbital platform, by pressing the “c” key, followed by “Enter”, on his keyboard.
2. Enable propulsion electronics, by pressing the “e” key, followed by “Enter”, on his keyboard.
3. Pre-heat hot-gas thrusters, by pressing the “p” key, followed by “Enter”, on his keyboard.

Once the start-up sequence has been completed, if the ‘n’ key is pressed, followed by “Enter”, the thrusters must be turned on. If the “o” key is pressed, followed by “Enter”, the thrusters must be turned off.

Write a program that will monitor the keyboard for commands. Thrust can only be switched on and off if the start-up sequence was completed in the correct order. Attempts to thrust before completion of the start-up sequence should be met with a suitable error message. Likewise, entering the start-up sequence in an incorrect order should result in an error message. The state of the thrust must also be displayed to the screen. However, the state should only be written to the display when the thrust state changes.

ALL errors should result in restart of the start-up sequence and any key input, other than the specified characters, should be treated as an error with a resulting error message.

Turn In: Printouts of the 5 steps of software design as detailed in Recitation 2. You may choose any one of the following three methods of implementing part 2 of step 3 (Design): Pseudo-code, Flow chart or State diagrams. Indicate the amount of time you spent working on this problem.
Also: Copy your program code as username_PS3_P5.c to \CDIO-Prime\16.070HW\ps3\p5, replacing username with your Athena username.