Workshop Grading and Promotion Policy

Workshops for this course will be assessed using the following criteria:

- Workshops are graded based on two components:
  1. Individual Logic Assignment (40%)
     - Individual work is due **2 days after the assigned date** (class) by **end of day 23:59 EST**
     - Individual logic assignments are to be done individually
     - Members who do not submit work on-time, will receive a zero grade for the workshop
     - Members who receive a zero grade for the individual part, will not be eligible to receive grades for the group solution part
  2. Sub-Group Overall Solution (60%)
     - Group solution is due **4 days after the assigned date** (class) by **end of day 23:59 EST**
     - **Name and ID of all contributing members must be stated at the top of all file submissions**
     - If not submitted on-time, a zero grade will be applied for the group portion of the workshop
     - If the submitted solution is essentially a copy of the individual parts thrown together containing no effort to properly integrate as a seamless overall solution, a zero grade will be applied for the group portion of the workshop

- A zero grade on a workshop will not be counted towards the minimum necessary number of completed workshops

- Video presentations are due **1 day after your next class** by **end of day 23:59 EST**
  - Each student must do a video presentation **at least once** by the end of the term and should minimally consist of the following:
    - Description of the problem and its solution in non-technical terms. You should assume your audience is non-technical and interested in using your application solution.
    - Market your application solution by providing sample screenshots of how you envision your application to look which should include a sample workflow demonstrating how easy it is to use

- You must **successfully complete 9 workshops** (if > 9 are completed, the best 9 will be used)

- Workshop solutions and presentations will be evaluated using the published workshop rubrics

Group Breakdown

Each group has **two sub-groups** determined by the assigned **member number**:

Sub-Group 1: **Members 1-3**

- **Member-1**: Responsible for doing workshop **Logic 1**
- **Member-2**: Responsible for doing workshop **Logic 2**
- **Member-3**: Responsible for doing workshop **Logic 3**

Sub-Group 2: **Members 4-6**

- **Member-4**: Responsible for doing workshop **Logic 1**
- **Member-5**: Responsible for doing workshop **Logic 2**
- **Member-6**: Responsible for doing workshop **Logic 3**

Sub-Group Solution

- Each sub-group is a team and **must work together** creating the overall group solution
- The group solution is not to be done by an individual. The group solution is expected to be a seamless solution (looking as though one person has done it) and has undergone refinement and testing to ensure the logic properly addresses the workshop problem.
- If the submitted work amounts to essentially copying and pasting everyone's logic part together, a zero grade will be applied for the group work portion.

Work Submission

All work must be emailed to your instructor. You must follow the email guidelines described below.

- All work submitted (applied to both individual and group submissions) **requires all contributing members names to be stated at the top of all files being submitted**

Email Subject Line

- Highlighted parts indicate your specific information
- There are no spaces
- **APS145-[SECTION]-WS[#]:Group[#]**
  - Example: **APS145-NAA-WS1:Group3**

File Attachment

Individual Work Submissions

Attach a file containing your work (**pseudo code** OR **flowchart**)

- Highlighted parts indicate your specific information
- **Pseudo code**: **logic[#].fullname.pseudocode.txt**
  - Example: **logic2.Cameron Gray.pseudocode.txt**
- **Flowchart**: **logic[#].fullname.flowchart.jpg** *(Note: .jpg or .png)*
  - Example: **logic3.Cameron Gray.flowchart.png**

Sub-Group Solution Submission

Attach a file containing your group work (**pseudo code** OR **flowchart**)

- Highlighted parts indicate your specific information
- There are no spaces
- **Pseudo code**: **ws[#].group.pseudocode.txt**
  - Example: **ws1.group.pseudocode.txt**
- **Flowchart**: **ws[#].group.flowchart.jpg** *(Note: .jpg or .png)*
  - Example: **ws3.group.flowchart.png**

Presentation Submission

Video files can be quite large and will most likely be rejected by Seneca's email services. Therefore, you will have to **SHARE** your video file using your Seneca account Microsoft **ONE drive**.

- **Video file name**: **WS[#].fullname.video.mp4**
  - Example: **WS4.Cameron Gray.video.mp4**
- Go to https://myseneca.ca, click on (top left corner) and select the One Drive application option
- Upload your video file: 

- Share  the file with your instructor: **Copy the shared link** 

- Paste the shared link into your email

Workshop - 1

Workshop Value: 10 marks (5% of your final grade)

Learning Outcomes

Upon successful completion of this workshop, you will have demonstrated the abilities:

- to decipher and identify a problem
- to analyze and decompose a problem
- to identify the required detailed steps to solve a problem
- to communicate the solution to fellow peers and non-technical business persons

---

Workshop Overview

Computational thinking to a software developer/computer programmer, is a critical skill that is applied all the time. This workshop requires you to develop a **phone application** that simulates a basketball shootout game. For the exercise described below, apply the necessary computational thinking steps to solve the problem.

Workshop Details

Logic 1 (members 1 & 4)

Two students challenge each other to a basketball shootout. Before the challenge begins, a simulated die will be rolled once by each student. The lowest number will determine which student goes first. If there is a tie, the game will repeat until a winner can be declared (this can potentially repeat many times).

Logic 2 (members 2 & 5)

The students agree to limit the number of ball throw attempts to 3 throws each. The first player will make all 3 throw attempts (keep track of the successful baskets made where the ball goes into the basket). After the first player makes all three shots, the second player will make all 3 throw attempts. The student who makes the most baskets (gets the ball in the hoop) will be declared the winner. In the case of a tie, the game will be repeated until a winner can be determined.

Logic 3 (members 3 & 6)

The losing player of the shootout will have to give the winner lottery ticket(s). The number of lottery tickets is determined by the total number of baskets made by the winner less the total number of baskets made by the losing player. For every tied game (if there were any tied games), the number of lottery tickets is reduced by ½. If the final calculated number of lottery tickets has a decimal value, it should be rounded up to the nearest whole number since you can't purchase half a ticket!

Example: If the player-1 made a total of 9 baskets, and player-2 made a total of 7, and they had 3 tied games, the number of lottery tickets would initially be 9-7=2, but reduced by 3 X 0.5=1.5, making the owed number of tickets 0.5 which must be rounded up to 1 lottery ticket.

If there are many tied games that contribute towards a negative number of lottery tickets owed, then the losing player must give 1 lottery ticket.

Your Task

1. Where applicable, apply the core components of the **computational thinking** approach to problem solving to help you synthesize a solution
2. **Sub-group members 1-3** should describe the solution to the above scenario in a sequence of detailed step-by-step instructions using **pseudocode** (review the grading rubric for workshops and be sure to address each gradable component)
3. **Sub-group members 4-6** draw a detailed **flowchart** to communicate the process (should be easy to understand by **non-technical business persons**) (review the grading rubric for workshops and be sure to address each gradable component)
4. **Test** your defined processes using test-cases and **refine** as necessary!
5. **Each sub-group**, create an overall solution for the simulated basketball shootout application game
6. **Presentation**: Decide among yourselves which member among the entire group will be doing the presentation. Priority should be given to those who have not yet done one.

The following table summarizes the responsibilities of each group member.

Sub-Group 1 (pseudo code)				
Task	**Subtask**	**Member(s)**	**Marks**	**Comments**
Pseudocode	Logic 1	**1**	40%	Members are graded individually
	Logic 2	**2**	40%	
	Logic 3	**3**	40%	
	Group Solution	**1-3**	60%	Eligible members get same mark
Sub-Group 2 (flowchart)				
Task	**Subtask**	**Member(s)**	**Marks**	**Comments**
FlowChart	Logic 1	**4**	40%	Members are graded individually
	Logic 2	**5**	40%	
	Logic 3	**6**	40%	
	Group Solution	**4-6**	60%	Eligible members get same mark
Video	Presentation	**1 or 4**	100%	Members rotate weekly