ASSIGNMENT REQUIREMENT

MODULE : INTRODUCTION TO PROGRAMMING

COURSE : DICT/DNDFC

CODE : ITGP2008\V1.0G4

OVERVIEW

In this assignment student will demonstrate their competency in developing a program in a team using fundamentals of programming such as variables, conditional and iterative execution, collections (list, dictionary or tuple), method, packages and OOP.

The assignment consist of induvidual and group element which will be completed by 4 – 6 students in a team. The entire assignment are divided into four parts which covering (A) program development, (B) individual portfolio, (C) presentation/demonstration and (D) peer review.

You and your team are highly advised to carefully design the game structure and distribute the component evenly to each member. The program required collective effort and a good teamwork. Therefore, every member must contribute by designing and developing the component assigned and then integrate into the main program. Working alone/in silo may resulting in unfavourable outcome / marks for the module as a whole.

WEIGHTING:	**40%**
DUE DATE:	**Check Moodle Submission Link**

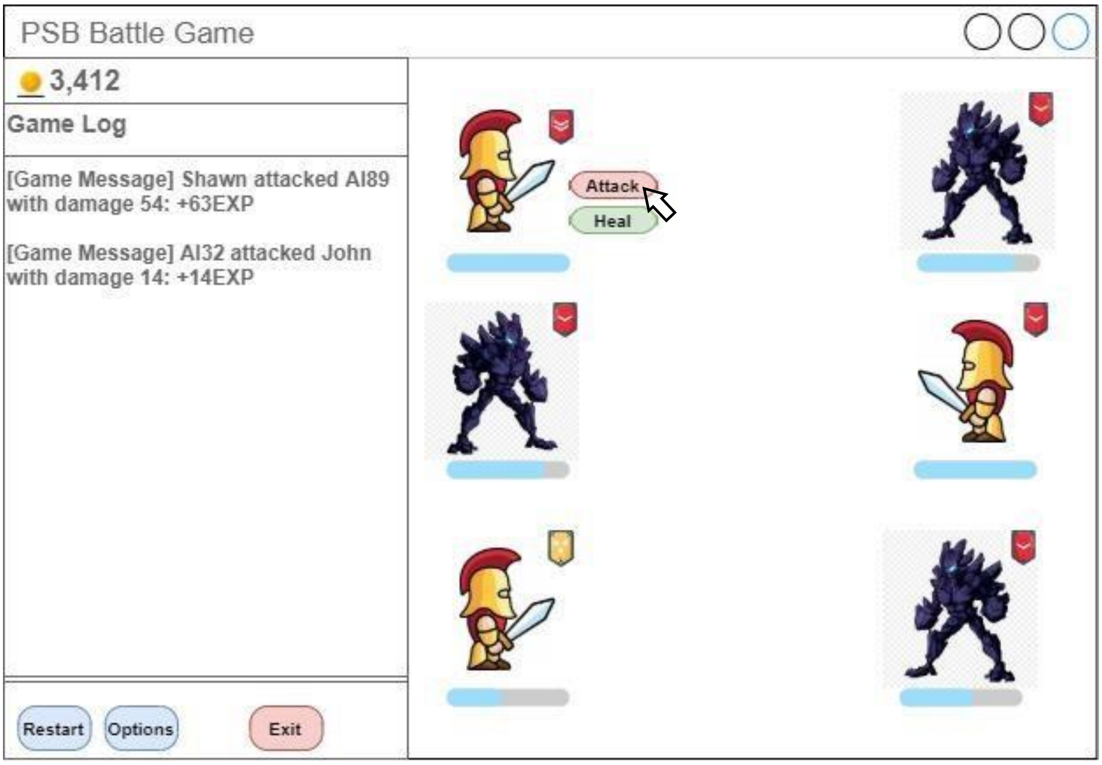
ASSIGNMENT REQUIREMENT

Project: Developing a Turn-Based Battle Game

Your team has been assigned to develop a console or GUI turn-based battle game. The game allows player setup his/her team which made up of a numbers of units (default is 3). Each unit has a name, health point (HP), attack point (ATK), defence point (DEF), experience (EXP) and a rank. On top of that, a unit can be either a Warrior or Tanker which having different strength in different range for ATK and DEF point. The initial value for each attribute point are describe in the details requirement in Part A – Table 1.

The game will then setup another team belongs to AI which made up of same number of unit as the player's team. The type of unit will be assigned randomly by chances. The name of each unit will be specify by the player while the name of each unit for AI team will be defined with prefix with "AI" follow by 2 random digits i.e. AI87.

A player can select a unit from his/her team and attack the other unit (the target) in the opponent / AI team. Unit which are severely damaged (i.e. HP equals to less than 0) will be defeated and removed from the team. The team (player / AI) which destroy all the opponent units first will be declared as winner of the game.



PSB Battle Game

3,412

Game Log

[Game Message] Shawn attacked AI89 with damage 54: +63EXP

[Game Message] AI32 attacked John with damage 14: +14EXP

A Program 30%

At a minimum, the program must contains the following core features: [20%]

<table>
<tr><th>Features</th><th>Description</th></tr>
<tr><td>Game Setup</td><td>This features allows player to setup a team of 3 units and select which profession (Warrior / Tanker) each unit will be. This features allows player assign name for each of the unit. The initial point for each attributes within a unit are assigned with the following rules:
<table>
<tr><th>Attribute</th><th>Warrior</th><th>Tanker</th></tr>
<tr><td>Health Point</td><td>100</td><td>100</td></tr>
<tr><td>Attack Point</td><td>Range between 5 - 20</td><td>Range between 1 - 10</td></tr>
<tr><td>Defence Point</td><td>Range between 1 - 10</td><td>Range between 5 - 15</td></tr>
<tr><td>Experience Point (EXP)</td><td>0</td><td>0</td></tr>
<tr><td>Rank/Level</td><td>> or 1</td><td>> or 1</td></tr>
</table>
Table 1: Example of initial value for each attribute point within a unit. Student may define their own range for ATK and DEF based on their game setting
This features then will setup the AI team which consist of 3 units with random name and profession as mentioned earlier.</td></tr>
</table>

Player Attack	This features allows player to select a unit from his/her team to attack and then select another unit from the opponent team as the target. The program will then initiate an attack based on the selected unit and the target. The attack resulting the health (HP) of opponent being deducted/no effect and both selected unit and targeted unit will gain experience (EXP). The following are sample calculation / game rules when an attack take place: • Damage = attacker.ATK – target.DEF + (random between negative 5 to 10) • target.HP deducted based on the calculated damage point. • attacker.EXP increase based on the calculated damage point. • Target.EXP increase based of its DEF point. • Extra EXP i.e. 20% - 50% gained by the targeted unit when the following incident happens: ○ Damage point more than 10 (gain extra 20% EXP) ○ Damage point equal or less than 0 (gain extra 50% EXP) A unit will be promoted (level up) when the EXP point reached 100. □ Rank will be updated upon promotion. □ EXP will be deducted by 100 point. A unit will be removed from the team when the HP become 0 or less.
AI / Opponent Attack	Once the player completed an attack, the AI team will then take its turn to initiate an attack. Same rules applied on the damage and experience point calculation above. You as the game developer will need to define your own logical/creative algorithm for the AI team to make the unit and target selection.
User Interactivity	The program allows player (user) to interact via console/python shell by asking question(s), getting input from keyboard, and provide options for actions. **Note**: Higher marks can be obtained if GUI application is develop (See enhanced features below)
Game Message	The program should display necessary messages i.e. status of all unit at the point of request, battle information, etc
Event Log	The program should record all game event, timestamp and its details and save in a plain text file on the local machine.
Documentation	Appropriate in-code commenting

Enhanced features, concept and packages applied [10%]

Features	Description
Game Setup & Game Play	On top of the core features for battle mode specified above, the program can be enhanced by the following features • Each team can collect/earn coin based on the damage point applied to the opponent units. • Each team can recruit/hire a new unit with the collected coins to strengthen the team. • Allowing user to save the state of the game and resume later when the program is loaded. • Any other interesting game play / battle rules can be applied.
OOP Concept	Appropriate use of Object-Oriented Programming paradigm covering the abstraction and encapsulation concept, object collaboration as well as separation of concerns.
User Interactivity	The enhance user experience, the program shall provide decent Graphical User Interface for interaction. (Recommended packages are tkinter, pygame, turtle, PyQt, Kivy)

Technical Specification and Guidelines:

You are expect to apply the concepts learn in this module when developing this game. You may use List to store the collection of units. Even better if you uses Object Oriented Programming concept to manage the units and team in the game. You may use other python libraries / packages for the development and must be included into your individual portfolio.

B Individual Portfolio 5%

An individual portfolio must be produce by every member within a team to document their individual contribution and to demonstrate their competency in designing, implementing, test and debugging part of feature/algorithm within the program. Each member must produce a report covering the technical context i.e. flow chart for the algorithm/function, source code, test plan/procedure and the test results. Besides, each member must prepare a personal learning reflection report towards this assignment.

The purpose of the personal learning reflection is to allow each individual student to summarise the insights and experiences he/she has gained from this assignment / class activities within the module (Introduction to Programming) at PSB Academy. Students are required to highlight their own personal perspectives, opinions and feelings. It provides an honest summary of the work undertaken throughout the module, and the skill sets that were developed.

The key of the assessment criteria is based on how well the student demonstrated genuine engagement with the module, and how well he/she apply the Python coding skills in the group project, or any other related class activities.

Below are the breakdown items required for the individual portfolio report (report outline):

1. Cover page.
2. Introduction
3. Technical Report (on assigned program feature – one only)
    a. Flow Chart
    b. Source code (appropriately formatted, screenshort are not acceptable) c. Test Report
4. Personal Learning Reflection Report (~600 – 1200 words)
    a. Challenges encountered and how it was overcome.

       These are typically short summaries of moments that significantly enhanced your learning in this module. The challenges / critical incidents can be either positive or negative experiences which provided strong opportunities for your professional development. When writing about such incidents, you should reflect on the ways that they prompted new skill development, or provided enhanced understanding of course material.

    b. Sense of achievement with evidence

       This part of reflection provides written evidence of your achievement. For example, you write about any accomplishment that made you feel proud, or any activities that you like to share as your success story.

    c. Personal statement / Conclusion

       The Personal Statement provides an opportunity for you to summarise your newly developed skills and professional philosophies. Based on the experiences you've gained, how would you describe yourself professionally? How will you use the skills you learnt in your professional development in the future?

Tips: How to write a good reflective report:

1. **Be critical.**

   Although the content of a portfolio will be more personalised than other assignments, you should use the same level of critical analysis as you do for any essay or exam.

2. **Be comprehensive.**

   Make sure that you include a good range of experiences that exemplify your work throughout the duration of your practical group work/class activities. You might choose to highlight one or two periods of your work, but these should be contextualised within your overall experience.

3. **Don’t be afraid to reveal your weaknesses.**

   Writing about your professional insecurities and weaknesses shows examiners how much you’ve developed throughout your course. It also enables you to reflect on theories and methods that might benefit you in future.

4. **Devise a plan for development.**

   Your Reflective Portfolio should testify to your development as a practitioner throughout the duration of your course. However, to write a really strong portfolio you should also demonstrate an action plan for future development. Think about what knowledge and skills might address the professional weaknesses that your reflections reveal, and indicate how you intend to develop these.

Mistakes to avoid in writing reflective portfolio:

The most common mistake in Reflective Writing is to be either too objective and scholarly, or too emotional and non-critical. Either mistake is equally wrong. Students should aim for a middle ground in their writing, in which they highlight their own personal feelings and reflections but analyse these with reference to the theoretical course material. Another common mistake is not providing enough relevant evidence to support your reflections. Be sure to include supporting evidence like photos, diagram, etc.

Finally, be sure to keep your portfolio well organised and professional-looking. It is true that Reflective Portfolios entail a less formal style of writing, but students sometimes believe that this allows for disorganised presentations with jumbled notes, illegible handwriting and poor grammar. Remember that this is still an academic assignment, and all the normal standards of achievement apply!

C Presentation / Demonstration [4%]

Each group will be required to give a 10-12 minutes presentation with the aid of presentation slides and a live demo of their program. The presentation will be judged based on the following criteria:

- Clarify of presentation / demonstration
- Flow of presentation / demonstration
- Audience engagement
- Teamwork
- 0 marks will be awarded for absentee(s).

PART D – Peer Review Evaluation Form [1%]

The peer-review evaluation provided opportunities for all students to rate their peers in working together in this assignment. The evaluation criteria are as follow:

- Effort of Contribution
- Teamwork sprite / effectiveness
- Participation in Meeting and Discussion

Do ensure you rate each and your peer rationally and fairly as this is a single attempt evaluation. All students must complete this evaluation or will be resulting in 0 marks to be awarded in the assignment.

WHAT TO DO?

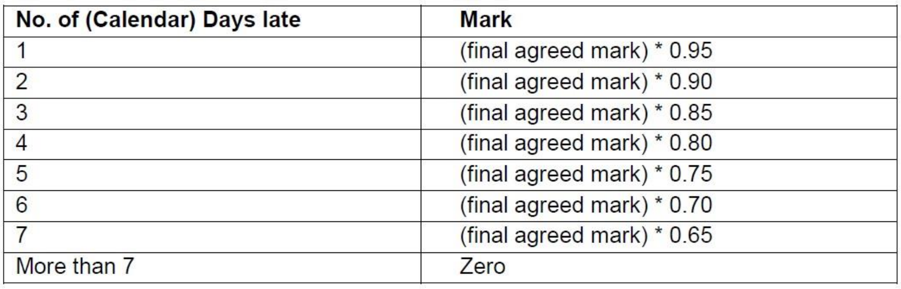
1. Form a team of 4 – 6 students within the same Lab Group.
2. Discuss and understand the requirement of the assignment and defined the parts / component to be implemented.
3. Develop the Program
    a. Distribute the task within team member and set the due date for each components
    b. Explore and research additional / suitable packages for the program
    c. Design, develop and test the component assigned
    d. Integrate the components and test the program as a whole.
    e. Submit the program with all source code file, and resources to Moodle (by Team Leader only)
4. Prepare and submit individual portfolio in Moodle. (by all)
5. Prepare and conduct the presentation / program demonstration (by all)
6. Complete the peer review evaluation form in Moodle (by all)

WHAT IS TO BE SUBMITTED?

Deliverables & Submission

1. A zip file (submitted by Team Leader)
    a. all source code file (.py) and resources required by the workable program
    b. Presentation slide in .ppt/.pptx
2. Individual Portfolio report in MS Word
3. Submission of Peer Review Evaluation (in Moodle)

Students are to submit their project prior the due date indicated in Moodle. Any late submission will attract mark deduction as follow:



No. of (Calendar) Days late	Mark
1	(final agreed mark) * 0.95
2	(final agreed mark) * 0.90
3	(final agreed mark) * 0.85
4	(final agreed mark) * 0.80
5	(final agreed mark) * 0.75
6	(final agreed mark) * 0.70
7	(final agreed mark) * 0.65
More than 7	Zero

PLAGARISM & PANALTY

Plagiarism is the practice of using another writer’s ideas or observations and presenting them as the author’s own. If students directly or indirectly use another author’s words without due acknowledgements to the original source, they are guilty of plagiarism and their work cannot be accepted as academic writing.

Plagiarism is a serious offence in PSB Academy and may lead to penalties in the student’s assessment, in most cases even failure of the assignment and / or module. In severe instances, plagiarism may lead to exclusion of the student from the programme of study. (*read more detail in Student Handbook on 4.12 Academic Misconduct : PAC-ADM-G02 R15 Student Handbook*)

ADDITONAL RESOURCES

Tkinter Resources

- Overview : https://www.youtube.com/watch?v=VMP1oQOxfM0
- Short Documentation : https://www.tutorialspoint.com/python3/python_gui_programming.htm
- Example of usage :https://www.datacamp.com/community/tutorials/gui-tkinter-python

MARKING RUBRICS

Deliverables	Components	Below Expectation	Meeting Expectation	Good	Excellence
Program [30 marks]	Game Setup & Game Play *[3 + 5 marks]*	0 - 2	3 - 4	5 - 6	7 - 8
	Player Attack	1 - 2	3 - 4	5	6
	AI Attack	1	2	3	4
	User Interaction *[3 + 2 marks]*	1	2 - 3	4	5
	Game Message	0	0.5	1	1
	Event Log	0	0.5	1	1
	OOP	0	1	2	3
	Documentation	0	1	1.5	2
Individual Portfolio [5 marks]	Introduction	0	0.5	1	1
	Technical Report	0	1	1.5	2
	Personal Learning Reflection Report	0	1	1.5	2
Presentation / Demonstration [4 marks]	Clarify of presentation / demonstration	0	0.5	1	1
	Flow of presentation / demonstration	0	0.5	1	1
	Audience engagement	0	0.5	1	1
	Teamwork	0	0.5	1	1
Peer Review [1 mark]	Peer Review Evaluation Form	Complete – 1 mark Incomplete / No Attempt – 0 mark			