**Part A:**

In this part, you are required to write a technical article based on EITHER ONE of the two the

following topics.

1. Buffer Overflow Attacks

Format of Report:

For the purpose of this written assessment, you are assumed to be a postgraduate student capable of producing a high-quality report. As such, you should attempt at your best to reflect this quality in the preparation of the written assessment. The report should contain the following six sections, but not limited to:

Section 1: Introduction – Provide a general overview of the topics

Section 2: Buffer Overflow Attack Discuss the security problems with sample Python codes. Please explore stack overflow, heap overflow, and so on. Screenshots of runtime errors would help.

Section 3: Mitigation Strategy – Discuss mitigation strategies. Analyze why the code is not perfect and what can be improved. Screenshots of successful runtime executions would help.

Section 4: Conclusion – Summarize your findings in this section.

Section 5: References – Follow Harvard or APA referencing style. Please provide in-text citations.

Part B:

In this part of the assignment, you will be writing a Python code to demonstrate your understanding and knowledge about secure programming. You are not required to submit additional information, such as flowcharts. However, please add necessary comments (https://www.w3schools.com/python/python\_comments.asp ) into your Python source code if applicable, to improve its readability.

**Develop a secure user registration and login system**

Problem Description: Recently a software company has hired you. You are assigned to a team of three members. Your team is currently developing an online course-providing platform. Your job is to develop a secure user registration and login system. The requirements are as follows:

1. The username should not be empty and should be no more than 20 characters.

2. The password must be at least 8 characters long and should contain at least one lowercase letter, one uppercase letter, and one numeric value.

3. After successful registration (with username and password), the program will show a successful message

4. Then the program will ask for login (providing username and password).

5. If username and password are correct, it will allow the user to login.

6. User will be allowed three attempts to login to the system. If unsuccessful, it will print a message like "the username and/or password are not correct".

7. After 3 failure attempts, the login system will lock the user down (Note: you can mimic this requirement by displaying a relevant message.)

 You have to write a Python program meeting the above-mentioned requirements. In addition to the functionalities, you have to take care of the security measures such as input validations, buffer overflows, etc. that are relevant to the system.