CS5138 Malware Analysis, Spring 2023
Project Proposal

Task 1: The names of your team members.

Student 1 : Adel Alshappip
Student 2: Edwin Cervantes

Task 2: An executive summary of your project.

For our final project we have chosen to create a disassembler with the purpose of disassembling Python code. This project was decided after being given a recommendation from Professor Hawkins and his examples of disassemblers that he made. Our product will be a command line based tool following the syntax of:

Here are just examples.

python3 disassembler.py -d test.py (using the -d option for disassembler tool 1)

python3 disassembler.py -t test.py (using the -t option for disassembler tool 2)

disassembler.py is our tool, while the test.py is the file we want to disassemble.

The deliverables of this project include a Python disassembler tool, a user guide, code documentation, and a set of test cases. It also provides a rubric for evaluating the tool. Our goal is to create a well-designed and well-documented disassembler tool that accurately disassembles Python code.

We will try to make our tool officiant to display/disassemble the code perfectly (printout useful info). In addition to disassembling Python code, we aim to create a tool that is user-friendly, well-documented, so it can be easy to update and use. We will test the tool thoroughly to ensure that it works as intended in all scenarios and handles unexpected inputs.

Task 3: Your deliverables.

Code repository: Create a GitHub repository that contains all of our code, including the disassembler tool.

User manual: Create a user guide that explains how to install and use the disassembler tool. This guide explains how to run the tool, and provides examples of using the tool to disassemble Python.

Technical documentation: This document is useful for anyone who wants to understand how the disassembler works, or who wants to modify it for their own use.

Test cases: Create a test suite with a large number of test cases to ensure that the disassembler tool works properly and produces correct output.

Task 4: The distribution of duties among teammates.

Adel- Developer:
Define the overall architecture and design the base of the disassembler.
Implement some of the core functionality of the disassembler.
Check the code for speed and memory usage.
Document the code and write a user manual.
Manage the project and ensure timely completion.
Test and provide feedback.

Edwin- Developer:
Implement additional core functionality of the disassembler.
Assist in implementing the core functionality of the disassembler.
Test the code with many different test cases.
Test the disassembler to ensure it works as we wanted.
Write test cases to cover all possible scenarios and to check unexpected inputs.
Provide feedback and suggestions for improvements.

This division of tasks allows each team member to focus on specific responsibilities and work efficiently towards completion of the project.

Task 5: A rubric that I can use to assess your final product.

#	Criteria	Excellent	Good	Fair	Poor
1	Does the disassembler correctly disassemble Python code?				
2	Are there any errors in the code?				
3	Are all test cases written and executed successfully?				
4	Is the code well-organized and easy to understand?				
5	Is the code properly documented?				

6	Does the disassembler work as intended in all scenarios?				
7	Does the disassembler produce accurate results?				
8	Do disassembler tools provide enough information to the users ?				
9	Does the disassembler handle unexpected inputs?				
10	Is the disassembler user-friendly and easy to use?				
11	Was the project completed on time?				