FIT1008 - Intro to Computer Science
Assessed Prac 3 - Marking Rubric

Semester 1, 2018

CRITERIA	POOR	SATISFACTORY	GOOD
Task 1 - 7 marks			
Program quality and documentation	Low quality solution which could include no documentation, no testing or a poorly designed solution. (0 marks)	Acceptable level of documentation, some test cases with an average implementation. (1 mark)	Well documented, thoroughly tested with clear and readable Python. (2 marks)
Student understanding	No evidence for understanding the solution. (0 marks)	Small gaps in understanding the solution, student is capable of answering most questions about the code. (1.75 marks)	The student clearly articulates the logic behind the solution and can explain and tweak the approach undertaken. (3.5 marks)
Correct output	Hash Table not complete, missing methods. (0 marks)	Incorrect implementation for some of the methods. (0.75 marks)	All 4 methods implemented correctly, including hash function. (1.5 mark)

Task 2 - 6 marks			
Program quality and documentation	Low quality solution which could include no documentation, no testing or a poorly designed solution. (0 marks)	Acceptable level of documentation, some test cases with an average implementation. (1 mark)	Well documented, thoroughly tested with clear and readable Python. (2 marks)
Student understanding	No evidence for understanding the solution. (0 marks)	Small gaps in understanding the solution, student is capable of answering most questions about the code. (1.5 marks)	The student clearly articulates the logic behind the solution and can explain and tweak the approach undertaken. (3 mark)
Correct output	Only a few combinations were tested or no analysis was provided. (0 marks)	Times provided for all 15 combinations, however the analysis is incomplete. (0.5 marks)	Times provided for all 15 combinations including a complete analysis of the results obtained. (1 mark)

Task 3 - 3 marks			
Program quality and documentation	Low quality solution which could include no documentation, no testing or a poorly designed solution. (0 marks)	Acceptable level of documentation, some test cases with an average implementation. (0.5 marks)	Well documented, thoroughly tested with clear and readable Python. (1 mark)
Student understanding	No evidence for understanding the solution. (0 marks)	Small gaps in understanding the solution, student is capable of answering most questions about the code. (0.75 marks)	The student clearly articulates the logic behind the solution and can explain and tweak the approach undertaken. (1.5 marks)
Correct output	Only a few combinations were tested or no analysis was provided. (0 marks)	Incomplete analysis, this may include only testing one of the provided files or less than 10 combination of **a** and **table_size**. (0.25 marks)	Thorough testing of each file including different a values and different table sizes (10 combinations in total). A detailed analysis was included for times, collisions and average probe length. (0.5 marks)

Task 4 - 4 marks			
Program quality including documentation	Low quality solution which could include no documentation, no testing or a poorly designed solution. (0 marks)	Acceptable level of documentation, some test cases with an average implementation. (0.5 marks)	Well documented, thoroughly tested with clear and readable Python. (1 mark)
Student understanding	No evidence for understanding the solution. (0 marks)	Small gaps in understanding the solution, student is capable of answering most questions about the code. (1 mark)	The student clearly articulates the logic behind the solution and can explain and tweak the approach undertaken. (2 marks)
Correct output	Incorrect implementation of quadratic probing, results in a sequence other than h+0, h+1, h+4, h+9, ... (0 marks)	Quadratic Probing implemented, however no further analysis was completed. (0.5 marks)	Correct implementation of Quadratic Probing, dynamic hashing and including a detailed analysis (1 mark)

Task 5 - 4 marks			
Program quality including documentation	Low quality solution which could include no documentation, no testing or a poorly designed solution. (0 marks)	Acceptable level of documentation, some test cases with an average implementation. (0.5 marks)	Well documented, thoroughly tested with clear and readable Python. (1 mark)
Student understanding	No evidence for understanding the solution. (0 marks)	Small gaps in understanding the solution, student is capable of answering most questions about the code. (1 mark)	The student clearly articulates the logic behind the solution and can explain and tweak the approach undertaken. (2 marks)
Correct output	Incorrect implementation of Separate Chaining . (0 marks)	Separate Chaining implemented, with errors in the logic of collisions, the implementation of the Linked List, or incomplete analysis (0.5 marks)	Correct implementation of Separate Chaining including a detailed analysis. (1 mark)

Task 6 - 6 marks			
Program quality and documentation	Low quality solution which could include no documentation, no testing or a poorly designed solution. (0 marks)	Acceptable level of documentation, some test cases with an average implementation. (1 mark)	Well documented, thoroughly tested with clear and readable Python. (2 marks)
Student understanding	No evidence for understanding the solution. (0 marks)	Small gaps in understanding the solution, student is capable of answering most questions about the code. (1.5 marks)	The student clearly articulates the logic behind the solution and can explain and tweak the approach undertaken. (3 marks)
Correct output	Incomplete implementation of the word count or poor choice of hash table parameters. (0 marks)	Word count is present but frequency information is incorrect (0.5 marks)	Word count correctly implemented for each word including how common each word is. (1 mark)