Unit Testing

This document was intended to track all u	nit test cases carried out on										
our code.											
Incomplete Test	A test which isn't implement The code currently passses	ed correctly									
Passed Test Falled Test	A failed test needs to be ad	dressed									
Redundant Test	A test for a removed peice of	of code									
Test ID Test Case	Context	Test Description Testing the get function for the position	Test Data	Expected Result	Actual Result	Notes					
1.01 Retreive aircract position	Aircraft.position	attribute	testAircraft	[0, 0, 0]	[0, 0, 0]						
1.02 Retreive aircraft name	Aircraft.name	Testing the get function for the name attribute	testAircraft	testAircraft	testAircraft						
1.03 Retreive aircraft origin	Aircraft.originName	Testing the get function for the name of the origin attribute of the aircraft	testAircraft	Berlin	Berlin						
1.04 Retreive aircraft destination	Aircraft.destinationName	Testing the get function for the name of the destination attribute of the aircraft Testing the get funtion for the variable	testAircraft	Dublin	Dublin						
		Testing the get funtion for the variable which reads TRUE if the aircraft has									
1.05 Read if the plane has finished	Aircraft.isFinished	finished	testAircraft	FALSE	FALSE						
Read if the plane is being manually 1.06 controlled	Aircraft. isManuallyControlled	Testing the get function for the variable that reads TRUE if the aircraft is in manual control mode	testAircraft	FALSE	FALSE						
1.07 Retreive the speed of the aircraft	Aircraft.speed	Testing the get function for the speed of the aircraft	testAircraft	20							
		Calculating if the aircraft is acsending,	testAircraft	20	. 20						
	Aircraft.altitudeState	descending or stationary Calculating if the aircraft is out of the	1	<u> </u>	1						
1.09 Test if the aircraft is out of bounds	Aircraft.outOfBounds	visible airspace Setting the current altitude state of the	testAircraft testAircraft	TRUE	TRUE						
1.10 Setting the altitude state of the aircraft	Aircraft.setAltitudeState	aircraft	1		1 1						
		Creating a new vector object by setting the		vector.x = 1.0 vector.y = 1.1	vector.x = 1.0 vector.y = 1.1						
2.01 Store co-ordinates as a vector	Vector	x, y and z co-ordinates	[1.0, 1.1, 1.2]	vector.z = 1.2 vector.x = 1.0	vector.z = 1.2 vector.x = 1.0						
2.02 Retreive co-ordinates from a vector	Vector.X, Vector.Y, Vector.	Retreving the x, y and z co-ordinates from a vector object	[1.0, 1.1, 1.2]	vector.y = 1.1 vector.z = 1.2	vector.y = 1.1 vector.z = 1.2						
		Calculation the magnitude of a vector from		vector.2 = 1.2	veClULZ = 1.2						
2.03 Calculate magnitude of a vector	Vector.magnitude	the x, y and z values	[1.0, 1.1, 1.2] [12.0, 16.0, 21.0]	29	3 3 9 29						
2.04 Test for vector equality	Vector.equals	Test equality of the stored vector and an input vector	[1.9, 2.2, 7.4]	TRUE	TRUE						
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,			[9, 4.2, 5.1] [9, 4.2, 5.0]	FALSE	FALSE						
2.05 Calculate magnitude squared of a vector	Vector magnitude Courses	Calculation the enugre of the magnitude of	[1.0, 2.0, 2.0]								
			[12.0, 16.0, 21.0]	841	1 841						
2.06 Normalise vector	Vector.normalise	Converting the vector to a normalised form	[1.0, 4.0, 8.0]	[1/3, 2/3, 2/3] [1/9, 4/9, 8/9]	[1/3, 2/3, 2/3] [1/9, 4/9, 8/9]						
2.07 Scale a vector	Vector.scaleBy	Scaling the vector object by an input value	[1.0, 2.0, 3.0]	[1.0, 2.0, 3.0]	[1.0, 2.0, 3.0]						
			[1.0, 2.0, 3.0]	[-2.0, -4.0, -6.0]	[-2.0, -4.0, -6.0]						
2.08 Vector addition	Vector.add	Adding an input vector to the vector object	[2.0, 2.0, 4.0]	[3.0, 4.0, 6.0]	[3.0, 4.0, 6.0]						
2.2.7000 00000			[6.0, 8.1, 16.0] [1.0, 2.0, 3.0]	[7.0, 10.1, 19.0]	[7.0, 10.1, 19.0]						
		Subtracting an input vector from the vector	[2.0, 3.0, 4.0]								
2.09 Vector subtraction	Vector.sub	object	[1.0, 1.0, 2.0]	[1.0, 2.0, 2.0]	[1.0, 2.0, 2.0]						
		Calculating the angle between the object	[1.0, 6.0, 0.0] [1, 0, 0] [0, 1, 0]	[13.0, 6.0. 0.0]	[13.0, 6.0. 0.0]						
2.10 Angle between vectors	Vector.angleBetween	vector and an input bector		π/2	π/2						
				score.timePlayed = 0	score.timePlayed = 0						
		Creation a new errors object with default		score.manualTime = 0	0 score.manualTime = 0	Many of these attributes will not be in					
3.01 Initialise score	Score Score	Creating a new score object with default score values		score.gameOvers = 0	score.gameOvers = 0	the release version					
	Score Score.timePlayed Score.flightsSuccessful Score.timeViolated			score.timePlayed = 0 score.flights = 0	score.gameOvers = 0 score.timePlayed = 0 score.flights = 0 0 score.manualTime = 0						
Retreive time played and successful	Socre.umewanuai			score.timeviolated = t	u score.timeviolateu = u	many of these attributes will not be in					
3.02 flights	Score.gameOvers	Retrieving the score data from the object Adding time played to the current length of	6	score.gameOvers = 0	score.gameOvers = 0						
3.03 Adding time played	Score.addTime	session	4	9		Test consit of adding time in sequence					
				1127	7 1127	Test consit of adding time in sequence					
		Adding the leavest and to the succession	127								
3.04 Adding time in manual control	Score.addTimeManual	Adding time in manual mode to the current total manual mode length		12		Function was not used in final release					
3.04 Adding time in manual control	Score.addTimeManual	total manual mode length			2 12	Function was not used in final release Function was not used in final release					
3.04 Adding time in manual control 3.05 Adding seperation violation time	Score.addTimeManual Score.addTimeViolated	Adding time in manual mode to the current total manual mode length Adding seperation volated time to the current value		12	2 12 2 102						
		Adding seperation volated time to the		102	2 12 2 102 6 26	Function was not used in final release Function was not used in final release					
3.05 Adding seperation violation time	Score.addTimeViolated	Adding seperation volated time to the	8 4 14 88 10 16 0	12 102 26	2 12 2 102 6 26 1 1	Function was not used in final release Function was not used in final release Function was not used in final release					
		total manual mode length Adding seperation volated time to the current value	8 4 14 88 10 16 0 1 1 looped 10 times looped 13 times	102	2 12 2 102 6 26 1 1 0 10	Function was not used in final release Function was not used in final release					
3.05 Adding seperation violation time 3.06 Adding a successful flight	Score.addTimeViolated Score.addFlight	Adding seperation volated time to the current value Incrementing the number of successful flights	8 4 14 88 10 16 0 1 1 looped 10 times looped 13 times	12 102 26 1	2 12 2 102 6 26 1 1 0 10	Function was not used in final release					
3.05 Adding seperation violation time 3.06 Adding a successful flight	Score.addTimeViolated Score.addFlight	total manual mode length Adding separation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs	8 4 14 88 10 16 0 1 1 looped 10 times looped 13 times	12 102 26 1	2 12 2 102 6 26 1 1 0 10	Function was not used in final release Function was not used in final release					
3.05 Adding seperation violation time 3.06 Adding a successful flight	Score.addTimeViolated Score.addFlight Score.addGameOver	Adding seperation volated time to the current value Incrementing the number of successful flights	8 4 14 18 88 10 10 10 10 10 10 10 10 10 10 10 10 10	12 102 26 1	2 12 2 102 6 26 1 1 1 0 10 3 13 4 4	Function was not used in final release Function was not used in final release					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over	Score.addTimeViolated Score.addFlight	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the	8 4 14 18 88 10 10 10 10 10 10 10 10 10 10 10 10 10	10 100 26 1 11 11 4	2 12 2 102 6 26 1 1 1 0 10 3 13 4 4	Function was not used in final release Function was not used in final release					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over	Score.addTimeViolated Score.addFlight Score.addGameOver	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the	8 4 14 18 88 10 10 10 10 10 10 10 10 10 10 10 10 10	12 100 20 1 11 11 4	2 12 2 102 6 26 1 1 1 0 10 3 13 4 4 9 -3119	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over	Score.addTimeViolated Score.addFlight Score.addGameOver	total manual mode length. Adding seperation volated time to the current value. Incrementing the number of successful flights. Incrementing the number of game overs. Calculating the game score from the stored score values.	8 4 14 88 10 16 0 1 1 looped 10 times looped 13 times	10 100 26 1 11 11 4	2 12 2 102 6 26 1 1 1 0 10 3 13 4 4 9 -3119	Function was not used in final release Function was not used in final release					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score	Score.addTimeViolated Score.addFlight Score.addGameOver Score.adcJameOver	total manual mode length. Adding seperation volated time to the current value. Incrementing the number of successful flights. Incrementing the number of game overs. Calculating the game score from the stored score values.	8 44 48 88 88 10 10 10 11 10 10 10 11 10 10 10 10 10	112 100 24 1 1 1 1 1 2 3 10000	2 12 2 102 6 26 1 1 0 10 3 13 4 4 4 9 -3119	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint from the value.	8 4 14 14 16 18 18 18 18 18 18 18 18 18 18 18 19 19 11 10 16 10 1 1 10 10 11 10 10 11 10 10 11 10 10	12 100 26 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 10	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score	Score.addTimeViolated Score.addFlight Score.addGameOver Score.adcJameOver	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint from the value.	8 4 4 4 8 8 8 8 8 10 10 10 1 1 10 10 1 1 10 10 10 10 10 10 10 10 10 10 10 10 10	112 100 22 113 114 115 115 115 115 115 115 115 115 115	2 12 12 2 102 6 26 6 26 1 1 1 1 0 103 13 13 4 4 4 4 4 93119 0 10000 110.10.0]	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.05 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Refreive waypoint position 4.02 is mouse over waypoint	Score.addTimeViolated Score.addFlight Score.addGameOver Score.calculate Waypoint.position Waypoint.sMouseOver	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Refrieving the position of the waypoint from the object Testing if the cursor is positioned over the waypoint	8 4 14 14 18 18 18 18 18 18 18 18 18 18 18 19 19 11 10 16 10 1 1 10 10 1 1 10 10 10 10 10 10 10 1	12 100 100 110 110 110 110 110 110 110 1	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint from the value.	8 4 4 4 8 8 8 8 8 10 10 10 1 1 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 26 6 26 1 1 1 1 0 103 13 13 4 4 4 4 4 93119 0 10000 110.10.0]	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint	Score addTimeViolated Score addTimeViolated Score addGameOver Score calculate Waypoint.position Waypoint.isMouseOver Waypoint.isEntryOrExit	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint from the object Testing of the cursor is positioned over the waypoint Testing if the waypoint object is an ently leaf point.	8 4 4 4 8 8 8 8 8 8 8 10 10 10 1 1 10 10 1 1 10 10 11 10 10 11 10 10 10 10 10 10 10	12 100 100 110 110 110 110 110 110 110 1	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.05 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Refreive waypoint position 4.02 is mouse over waypoint	Score addTimeViolated Score addTimeViolated Score addGameOver Score calculate Waypoint.position Waypoint.isMouseOver Waypoint.isEntryOrExit	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retreiving the position of the waypoint from the object Testing if the cursor is positioned over the waypoint Testing if the waypoint object is an entitylexit point Testing the calculation of the cost between	8 4 4 4 8 8 8 8 10 10 10 11 10 10 1 1 10 10 1 1 10 10 1	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreve waypoint position 4.02 is mouse over waypoint 4.03 Test if entrylexit point 4.04 Calculate cost	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.sinCouseOver Waypoint.isInTryOrExit Waypoint.getCost	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs. Calculating the game score from the stored score values Retrieving the position of the waypoint from the object Testing if the curror is positioned over the waypoint Testing if the waypoint object is an entitylexit point. Testing the calculation of the cost between the a waypoint and the current waypoint	8 4 4 4 8 8 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entrylexit point 4.04 Calculate cost	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.sinCouseOver Waypoint.isInTryOrExit Waypoint.getCost	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entrylexit point 4.04 Calculate cost	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.sinCouseOver Waypoint.isInTryOrExit Waypoint.getCost	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entrylexit point 4.04 Calculate cost	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.sinCouseOver Waypoint.isInTryOrExit Waypoint.getCost	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entry/exit point 4.04 Calculate cost 4.05 Calculate cost between	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.sinCouseOver Waypoint.isInTryOrExit Waypoint.getCost	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entrylext point 4.04 Calculate cost 4.05 Calculate cost between Test Aircraft Face Aircraft das tests Face Aircraft das tests	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.sinCouseOver Waypoint.isInTryOrExit Waypoint.getCost	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entry/exit point 4.04 Calculate cost 4.05 Calculate cost between Test Aircraft The following is the test data used for many of the Aircraft diss tests	Score.addFlineViolated Score.addFlight Score.addGameOver Score.calculate Weypoint.position Weypoint.position Weypoint.isMouseOver Weypoint.isMouseOver Weypoint.getCost Weypoint.getCostBetween	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entrylexit point 4.04 Calculate cost 4.05 Calculate cost between Test Aircraft The following is the test data used for many of the Aircraft data lests Name Name Gestinations	Score addTimeViolated Score addFlight Score addGameOver Score addGameOver Score calculate Waypoint position Waypoint isMouseOver Waypoint isEntryOrExit Waypoint.getCost Waypoint.getCostBetween	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding separation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entrylexit point 4.04 Calculate cost 4.05 Calculate cost 4.06 Calculate cost between 4.07 Adding is the test data used for many of the Aurorat class tests 4.08 Name 6.07 Adding is the test data used for many of the Aurorat class tests 4.09 Aurorat class tests 4.09 Aurorat class tests 4.09 Adding separation violation time 6.09 Adding a separation violation time 6.09 Adding a successful flight 6.00 Adding a successfu	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.isMouseOver Waypoint.isMouseOver Waypoint.getCost Waypoint.getCost Berlin Dublin Dubl	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 4.01 Refreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entry/exit point 4.04 Calculate cost 4.05 Calculate cost 4.06 Calculate cost 4.07 Adding a game over waypoint 4.08 Test if entry/exit point 4.09 Calculate cost 4.09 Calculate cost 4.00 Calculate cost between	Score addTimeViolated Score addFlight Score addGameOver Score calculate Waypoint.position Waypoint.isMouseOver Waypoint.isMouseOver Waypoint.getCost Waypoint.getCost Berlin Dublin Dubl	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 3.08 Calculate score 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entry/lexit point 4.04 Calculate cost 4.05 Calculate cost 4.06 Calculate cost 4.07 Calculate cost between Test Aircraft The following is the test data used for many of the Aircraft disas tests Name origin/lexit cost Value origin/lexit point 4.05 Calculate cost between	Score addTimeViolated Score addGameOver Score addGameOver Score addGameOver Score calculate Waypoint position Waypoint isMouseOver Waypoint isEntryOrExit Waypoint getCost Waypoint getCost Union Topic on True 100, 100, True 100, 100, True 100, 100, True	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
3.05 Adding seperation violation time 3.06 Adding a successful flight 3.07 Adding a game over 4.01 Retreive waypoint position 4.02 Is mouse over waypoint 4.03 Test if entirylexit point 4.04 Calculate cost 4.05 Calculate cost 4.05 Calculate cost 4.06 Calculate cost 4.07 Retreive waypoint desired for many of the Aircraft class tests Aircraft Test Aircraft Test Aircraft Aircraft Name originName destination-Name de	Score.addTimeViolated Score.addFlight Score.addGameOver Score.calculate Waypoint.position Waypoint.isMouseOver Waypoint.isMouseCver Waypoint.getCostBetween testAircraft Bestin Bestin 100,100,1rue 0,0 True 100,100,True 100,True	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					
1.05 Adding separation violation time 1.06 Adding a successful flight 1.07 Adding a game over 1.08 Calculate score 1.08 Calculate score 1.09 Indicate score 1.00 Indi	Score addTimeViolated Score addGameOver Score addGameOver Score addGameOver Score calculate Waypoint position Waypoint isMouseOver Waypoint isEntryOrExit Waypoint getCost Waypoint getCost Union Topic on True 100, 100, True 100, 100, True 100, 100, True	total manual mode length Adding seperation volated time to the current value Incrementing the number of successful flights Incrementing the number of game overs Calculating the game score from the stored score values Retrieving the position of the waypoint Testing if the cursor is positioned over the waypoint Testing if the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between the a waypoint and the current waypoint. Testing the calculation of the cost between	8 4 4 4 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	100000 21 100000 11 11 11 11 11 11 11 11 11 11 11 11 11	2 12 12 2 102 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	Function was not used in final release Function was not used in final release.					