Coursework 1

Tom Cresswell 1903451

Question 1

1. Let **M** be the set of all states in the Simple State Machine
   Let **C** be the set **{T, Premium, not Premium}**
   S $\vDash_p$ Premium, T
   S $\vDash_F$ Not Premium, T

2.
   a. Not a run - There is no transition from the idle state to the Playing Song state.
   b. Not a run - There is no ‘Stop’ action available from the Idle state
   c. Not a run - There is no ‘Stop’ action available from the Idle state
   d. Not a run - There is no transition from the idle state to the Playing Song state.
3.
4. For Premium Users

ID	Input Set	Expected State	Actual State	Pass/Fail
T1	{Idle, (Choose playlist, T, Playlist chosen)}	Idle	-	-
T2	{Idle, (“Play”, T, Start playing)}	Playing Song	-	-
T3	{Idle, (“Play”, T, Start playing), (“Stop”, T, Stop playing)}	Idle	-	-
T4	{Idle, (“Play”, T, Start playing), (“Next”, T, play next}	Playing Song	-	-
T5	{Idle, (“Play”, T, Start playing), (Song finished, T, Play next)}	Playing Song	-	-

For Non-Premium Users

ID	Input Set	Expected State	Actual State	Pass/Fail
T1	{Idle, (Choose playlist, T, Playlist chosen)}	Idle	-	-
T2	{Idle, (“Play”, T, Start playing)}	Playing Song	-	-
T3	{Idle, (“Play”, T, Start playing), (“Stop”, T, Stop playing, start playing ad)}	Playing Ad	-	-
T4	{Idle, (“Play”, T, Start playing), (“Stop”, T, Stop playing, start playing ad), ad over}	Idle		
T4	{Idle, (“Play”, T, Start playing), (“Next”, T, stop playing song, start playing ad}	Playing Ad	-	-
T5	{Idle, (“Play”, T, Start playing), (Song finished, T, stop playing song, start playing ad)}	Playing Ad	-	-
T6	{Idle, (“Play”, T, Start playing), (“Next”, T, stop playing song, start playing ad, (ad over, T, play next)}	Playing Song	-	-

Question 2

1.

Test ID	a	b	C	Expected Outcome	Actual Outcome	Pass/Fail
T1	0	5	5	1	-	-
T2	1	5	5	2	-	-
T3	9	5	5	0	-	-
T4	10	5	5	0	-	-
T5	5	0	5	0	-	-
T6	5	1	5	0	-	-
T7	5	9	5	0	-	-
T8	5	10	5	1	-	-
T9	5	5	0	2	-	-
T10	5	5	1	2	-	-
T11	5	5	9	0	-	-
T12	5	5	10	0	-	-

2.

Added test cases for robustness testing

Test ID	a	b	c	Expected Outcome	Actual Outcome	Pass/Fail
T0	-1	5	5	Out of Range	-	
T1	11	5	5	Out of Range	-	
T2	5	-1	5	Out of Range	-	
T3	5	11	5	Out of Range	-	
T4	5	5	-1	Out of Range	-	
T5	5	5	11	Out of Range	-	

3. In my opinion, the above test suite, using Boundary Value Analysis along with additional tests to allow for Robustness Testing, is a high-quality test suite and covers both outputs – that is, the Out-of-Range outcome, and the normal distinct real numbers found from the solution. It should be noted, however, that the problem does not consider complex numbers, which cannot be shown with the current specification. Due to this, the test suite would find any fault in the current spec, and therefore is a good fit, however, it would be impossible to show the complex solutions.
4. Due to the current specification being vague around complex numbers, there would be no need to increase the number of tests to allow for Worst Case and Robust Worst Case testing, as Boundary Value and Robust case testing already covers both outputs. From this, in my opinion, the most appropriate method for testing is Robust Boundary Value testing, as this covers every available output, and would find any fault in the computational problem, using Worst Case testing would increase the number of test cases with no additional benefit, and using just Boundary Value Analysis would be ineffective due to poor coverage – only one output would be covered.