159302 (Artificial Intelligence)
Assignment #2

Please accomplish the following check list in order to allow for accurate marking of your assignment.

Check list:

	Item	**your assignment details**		**Comments**
1	Names and ID numbers of Group Members	Cai Gwatkin 15146508		(maximum of 3 members in a group)
2	Operating System used for testing your codes	Windows 10		(e.g. Windows 8.1) Note that the start-up codes only work on Windows.
3	Compiler used	gcc 5.1.0		(e.g. gcc 5.1.0)
4	IDE used	SublimeText 3		(e.g. SublimeText 3, ScITE)
5	Complete source codes (cpp, h files), makefile	**Yes**		you are required to submit the complete source codes, including the makefile, or project file (if using codeblocks, etc.)
6	Algorithm components	Fuzzy rules	**full**/partial	Indicate '**full**', if you have completed the implementation of a component of the algorithm, or 'partial', if you are only submitting a partial implementation.
		Fuzzy membership functions	**full**/partial	
		Defuzzification	**full**/partial	
6	Specify the maximum time your fuzzy controller can successfully balance the pendulum, and at what initial pole angle and cart position.	Initial angle (in degrees)	n/a	Note: the bigger the initial angle that your fuzzy controller can handle, the better.
		Initial cart position (in meters)	n/a	
		Maximum balancing time (in minutes)	n/a	
7	Experiment Results (Control surface) in Excel Worksheet	**Yes**/No		indicate '**Yes**' or '**No**'
8	Extra work (Bonus): Enhancements/ Optimisations included	**Yes**/No. If Yes, list down enhancements you have added.		(e.g. successful implementation and calibration of Yamakawa's design)

Components	Details	Intructions
Inputs	x/x_dot and theta/theta_dot were combined using Yamakawa's technique as follows. Input X: X = 0.7*x + 0.3*x_dot Input Y: Y = 0.8*theta + 0.2*theta_dot	Specify all the inputs, including coefficients if using any.
Fuzzy Rules	There is one FAMM with 13 rules in the system. X is a measure of the "emergency in the cart's position" and Y is a measure of the "emergency in the rod's angle." If X is NL and Y is NL then output is NL. If X is NL and Y is ZE then output is NM. If X is NL and Y is PL then output is PS. If X is NS and Y is NS then output is NM. If X is NS and Y is PS then output is PS. If X is ZE and Y is NL then output is NL. If X is ZE and Y is ZE then output is ZE. If X is ZE and Y is PL then output is PL. If X is PS and Y is NS then output is NS. If X is PS and Y is PS then output is PM. If X is PL and Y is NL then output is NS. If X is PL and Y is ZE then output is PM. If X is PL and Y is PL then output is PL.	Specify all the fuzzy rules in the system. Indicate how many FAMMs are you using. Group the rules according to FAMMs.
Fuzzy Membership functions	Input: X Type: left trapezoid Name: in_nl a = 0, b = 0, c = -2.6, d = -1.3 Input: X Type: regular trapezoid Name: in_ns a = -2.6, b = -1.3, c = -1.3, d = 0 Input: X Type: regular trapezoid Name: in_ze a = -1.3, b = 0, c = 0, d = 1.3 Input: X Type: regular trapezoid Name: in_ps a = 0, b = 1.3, c = 1.3, d = 2.6 Input: X Type: right trapezoid Name: in_pl	Specify all the parameters of all membership functions used for the all inputs. (e.g. input, type, name, a=?,b=?,c=?,d=?)

	a = 1.3, b = 2.6, c = 0, d = 0 Input: Y Type: left trapezoid Name: in_nl a = 0, b = 0, c = -0.42, d = -0.21 Input: Y Type: regular trapezoid Name: in_ns a = -0.42, b = -0.21, c = -0.21, d = 0 Input: Y Type: regular trapezoid Name: in_ze a = -0.21, b = 0, c = 0, d = 0.21 Input: Y Type: regular trapezoid Name: in_ps a = 0, b = 0.21, c = 0.21, d = 0.42 Input: Y Type: right trapezoid Name: in_pl a = 0.21, b = 0.42, c = 0, d = 0	
Defuzzification Method	Centroid defuzzification method.	Specify method used.
If using multiple FAMMs, specify integration method.	n/a	Specify details of integration method.

Due to bugs that I believe were arising from float values becoming too small I was unable to completely balance the rod on the cart. I have included two files with this assignment submission that show debug outputs from the controller demonstrating the bugs I was running into. I was unable to find a solution to these bug in adequate time so I have submitted a complete system that I believe should work (my fuzzy logic is reasonably sound, needed some tweaking but I could not do that due to the bugs) but does not.

In short, the value of one of the inputs or of the output force became such small float values that it caused a drastic change in force applied. This caused the force to fluctuate from greatly negative to greatly positive as the controller attempted to keep the rod balanced and ended up more or less breaking the system.

One other confusing bug to occur was that when the cart with a balanced (angle close to zero) rod moved from positive zero to negative zero on the x axis there would be a sudden addition of negative force in the system. However, going from negative to positive with all other inputs similarly inverted this did not occur. This was a very difficult bug to debug

considering the FAMM, rules, and fuzzy membership functions were all symmetric so it should have done the same in both directions. In an attempt to fix this bug I forced the input X and Y values to zero if the were below a certain threshold – abs(input) < 1e-4 – but this did not entirely work. I have given up on attempting to fix this bug for this assignment.