Project #1b: Behavior Trees (due week 5).

 Required: Have at least 4 NPCs doing something interesting.

(40%) Implementing Behavior Tree Nodes in C++

* (10%) Implement a new selector node in C++
* (15%) Implement 3 unique new decorator nodes In C++
  + (5%) Per unique node
* (15%) Implement at least 10 unique nodes total (includes previous 4) in C++
* Debugging nodes do not count towards this total

(30%) Using the GUI, create 3 unique behavior trees

* (10%) per tree implemented
  + (5%) At least 8 nodes per tree
  + (5%) At least 3 nodes deep at some point in the tree
* Required: use all 10 of the nodes you created

(20%) Using the messaging system, implement agent communication in at least 2 of your behavior tree nodes

* (10%) Per tree
* List the nodes, line numbers, and trees you use messaging in
* Each tree must send and/or receive in at least 2 places
* At least 1 instance of communication between two different trees must occur

(10%) Project post-mortem readme.txt.

* Required: Include file and line number to find agent communication code and what the behavior trees do.
* Write one paragraph explaining what you liked about the project and framework.
* Write one paragraph explaining what you disliked about the project and framework.
* Mention any difficulties you experienced while doing the project.
* Describe any attempted extra credit.

 (Extra Credit 5%, no more than 15% per student) First to find a particular significant bug in the Behavior Tree framework (not the entire project). E-mail me at[steve.rabin@gmail.com](mailto:steve.rabin@gmail.com).

 (Extra Credit 5%) Implement Formations (must move various directions as a formation and not single-file queueing).

 (Extra Credit 5%) Implement 2D Flocking.

 (Extra Credit 10%) Implement 3D Flocking with pointing towards direction of travel (can't get 2D flocking extra credit) 

 Turn in: Remove Project1A directory and HW1.

 Turn in: Place in directory "Project1B" All code, resource files, and exe (-5% if I can't double click exe and run). (-5% for large unnecessary files: ".ncb" or ".sdf" file, hidden subversion directories/files, anything from the Debug/Release dir that isn't the exe)

 Turn in: Readme.txt (include file and line number to find agent communication code. Also include what the behavior trees do.).

 Turn in: Late projects should be e-mailed to [steve.rabin@gmail.com](mailto:steve.rabin@gmail.com). Rename the .exe to .ex\_ or else gmail won't deliver the attachment. Turn in late hardcopy UML diagrams at the following class.

**Create Behavior Trees**

1. To create behavior trees. First create node files. (there are example files in “BTResources” folder as reference)
2. Then include node headers in BehaviorTreeNodes.h.
3. Then run BehaviorTreeGenerator.bat to create your behavior trees. Currently this program does not support editing existing trees.

**Note**

1. The framework supports VS2012, VS2013 and VS2015.
2. By default the framework uses VS2013. You can change the setting via:

Project->Property->Congiguration Properties->General->Platform Toolset

1. After you finish the project. Please change “vs\_version” variable in Build.bat file for the version of Visual Studio you use. (2012 for VS2012, 2013 for VS2013, 2015 for VS2015)
2. Double click Build.bat to build the project and place executable in “BehaviorTrees” folder. Then test your executable to make sure it runs and does not crash.
3. Change your executable file name to the name of your project. (So when TA build your project the executable does not get overwritten)
4. Before submission, double click Clean.bat to remove all unnecessary files.
5. Do not change “BehaviorTrees” folder name.

**Checklist**

1. Executable is in **release** mode.
2. Fill out readme.txt.
3. Submitted on the network (“Submit”) folder BEFORE 6:00PM.
4. **DO NOT ZIP THE PROJECT.**
5. All code and resource files were in “Project1B” folder.
6. Removed all other files and folders in your submission folder.
7. Changed “vs\_version” variable in Build.bat file for the version of Visual Studio you used.
8. Did you test the exe runnable without the debugger? Test it by double clicking the exe to run (NOT IN THE DEBUGGER).
9. Did you remove all unnecessary files? Include but not limit to: the debug folder, release folder, ipch folder, ncb file, sdf file, and all subversion files. (Use Clean.bat)