Project #1: Behavior Trees (due week 4)

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

(50%) Implementing 10 Behavior Tree Nodes in C++

* (5%) per node implemented
* At least 3 of them have to be control flow nodes or decorator nodes.
* 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 (root node does not count)
* Required: use all 10 of the nodes you created

(10%) 2 instances of agent communication with messaging system

* (5%) per instance
  + At least one communication between different trees
* Required: sender and receiver have to be different

(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

* At least 3 different formations.
* Must move various directions as a formation and not single-file queueing
* Must have button(s) to switch between them.
* Must move to the target when user clicks it with mouse.

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

* Must have sliders or buttons to tweak separation, alignment, and cohesion.

 (Extra Credit 10%) Implement 3D Flocking

* With pointing towards direction of travel
* Can't get 2D flocking extra credit.

 Turn in: Remove “Homework1” directory.

 Turn in: Place in directory "Project1" All code, resource files, and exe (-5% if I can't double click exe and run). (-5% for large unnecessary files: ".ncb", “.db”, or ".sdf" file, hidden subversion directories/files, anything from the Debug/Release dir that isn't the exe)

 Turn in: Maintain original framework structure and make sure \*.bat batch files set to the correct VS version. (-5% if I can't double click Build.bat to build the project)

 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 “Source\ BehaviorTrees\Nodes\” folder as reference)
2. Then register nodes in “BTResources\Nodes.def”.
3. Then run BehaviorTreeEditor.exe to create your behavior trees.
4. In world.cpp, register agent with the tree
5. Read course slides for more info.

**Note**

1. The framework supports VS2013 and VS2015.
2. If you can’t build the project. You may need to install DirectX SDK.
3. 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. (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. Also manually check the size of your folder. If it exceeds 20MB then you may have some unnecessary files or folders.
5. Do not change “BehaviorTrees” folder name.
6. This is a 32-bit project. **DO NOT CHANGE IT TO 64-bit.**

**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 “Project1” 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, db file, sdf file, and all subversion files. (Use Clean.bat)