Hi 6170! Today we are presenting chAI, the online Turing test platform. The big idea is to create a website where users can submit chatbots, which then talk to real users. After each conversation, users are asked to gauge whether their counterpart was a human or a bot. To make it more fun, users can also talk to other users and pretend they are bots. By turning AI into a fun and competitive game, we will help advance the field and make it more accessible to the general public!

1 Next Monday's nano quiz will be on this finest British tradition. Do not forget to do the reading, DNJ will take it to heart: http://en.wikipedia.org/wiki/Tea_%28meal%29
The minimum viable product we would like to deliver is a website that supports the following:
- real-time interaction between two humans, or between a human and a bot.
- a way for users to upload bots, which will then be instantiated and talk to other bots, as well as human users.
- when running user code, we must make sure that it cannot do anything nasty to our deployment servers.
- a leaderboard, with rankings of the most human-like bot and the most bot-like human.
There are several risks that we must address in this project. The main three are:

- **Real-time chat**. The standard still does not support any way to reliably push information from the server to the client. HTML5 introduced web sockets, but they are not yet widely supported. If the user's browser does not support web sockets, we will automatically fall back to using long polling.

- **Running user-submitted code** is dangerous. We will address this risk by using the PyPy Sandbox, which allows us to control what modules are called by a python app. We will also limit resources, such as memory and time, available to the process running the user code.

- **Bot API must be future-proof**. Our bot API will certainly be extended in the future to support sending files, "nudging" the other user, changing one's status etc. We are thinking of having separate callbacks for each action, so new functionality can be introduced by adding new callbacks.