

# University of Utah School of Computing

CS 3500/5010

Handout 8

November 4, 2010

---

## Problem Set Six

In this problem set, you will be creating a Visual Studio solution that solves the problem described below. Your solution must be called `ps6`, and must consist of two projects, one file, and two optional projects:

- A class library project called `LineSocket`.
- A unit test project called `LineSocketTest`.
- A PDF file called `report.pdf`.
- An optional supplementary credit console project called `ChatServer`.
- An optional supplementary credit forms project called `ChatClient`.

Use C# as your implementation language.

**You must work on this project with a partner. Please find one right away. You must also employ pair programming on this assignment.**

Either you or your partner must attend a lab section on Wednesday, November 10. You or your partner will need to

- Tell the TA your name and the name of your partner
- Tell the TA the login name of the partner in whose repository the solution will be committed.

We will retrieve your solution for grading by running the Linux command

```
svn --username cs3500 --password ##### checkout  
    svn://lenny.eng.utah.edu/home/XXXXXXX/cs3500/ps6/trunk ps6
```

(put it all on one line) where `#####` is the grading password `XXXXXXX` is the CADE login name of the partner who in whose repository the solution resides . We will run the command, which will give us the most recently committed version of your solution, sometime on the morning of Friday, November 12. This is the version that we will grade.

## LineSocket Project

One problem that many of you had during the Boggle project was that whereas the Boggle protocol was defined in terms of complete lines of text, the sockets you were using dealt with bytes. A line-oriented socket would have been much easier for you to use.

I have specified a LineSocket class, which is a line-oriented socket. You can retrieve that specification by getting the LineSocket solution from the course repository.

You and your partner are to use pair programming techniques to implement the LineSocket class.

## LineSocketTests Project

You and your partner are to use pair programming techniques to implement unit tests for the LineSocket class.

## Written Report

You and your partner are to prepare a written report reflecting on your experience with pair programming. Discuss your experience with pair programming, how it compared with the style of cooperation that you used in PS 5, and whether you think that pair programming is a help or a hindrance.

When your report is completed, add it to your solution by right clicking on the solution in the Solution Explorer and adding the report as an existing item.

## ChatServer and ChatClient Projects

These two projects are optional. If you and your partner do them, you can each earn up to 100 points of supplementary credit. Any credit that you earn will be used to increase your overall problem set score, *but only up to a maximum of the total number of points available on the required portions of the problem sets*. (In you are already maxed out on your total problem set score, this optional part of the assignment will not be worth your while from a grading standpoint.)

You and your partner are to use pair programming techniques to create a client/server based chat program. It should support multiple chat rooms, and it should allow any number of clients to join a room.

Your implementation must be based on your LineSocket class. You should define your own protocol.