Course Outline: Math 6/8980-012, Summer 04

Algorithms are deterministic recipes to compute or decide something. In this course we will explore what happens when you leaven the recipe with chance. We are not going to develop the general theory of algorithms. Instead we will survey some interesting examples where probability and algorithm intersect.

We will not be following any particular text. I will draw the examples and supporting materials from a variety of sources. However, for reference I highly recommend *Introduction to Algorithms*, by Cormen, Leiserson, and Rivest.

Grades

If you attend every day, arrive on time, and pay attention throughout class, then you will earn at least a B. Every day you miss class, or are late, or leave early, or drift off to sleep during the lecture your grade drops one step. For example if you miss once and are late once then your grade drops to C+. You can increase your grade in one of two ways.

Programming projects: I will assign at least four programming projects. Some of these will ask you to write code which implements an algorithm, while others will give you code and ask you to study its performance. You can choose to do any, all, or none of the projects.

If you choose to tackle a programming project then bear in mind the following important stipulations.

- A project must be completed by the due date. I will be absolutely rigid on deadlines.
- Your work must be neat, complete, and well-organized. I will not attempt to fill in details for you, nor will I attempt to read something that is messy or disorganized.
- The work you turn in must be your own. If your work duplicates that of a classmate then you will both risk an F in the course. Copying is cheating, and cheating is a serious offense.
- You must use the programming language Python. I will devote one class meeting to learning Python. That is all it takes to get started. After that there are many printed and on-line references for learning the language.

I will not give numerical scores nor letter grades for each project. Each project will be marked either 'satisfactory' or 'unsatisfactory'. Each satisfactory project will raise your grade one step. Unsatisfactory projects will have no effect on your final grade.

Final exam: If, at the end of the course, you are not happy with your grade you can take a written final exam. If you take the final exam then your final grade will be based solely on your score, ignoring attendance and programming projects.