COS550 – Fall 2010
Theoretical Computer Science

George Markowsky
Department of Computer Science
University of Maine

GOALS

1. To provide a comprehensive survey of the key important areas in theoretical computer science.

2. To highlight the implications of theoretical computer science for practical computer science.

3. To increase your ability to handle formal reasoning and notation.

PREREQUISITES

1. A good understanding of basic mathematics and computer science.

2. A willingness to learn something new and to work.

3. An ability to handle theoretical material at the graduate level.

GRADING

Your final average will be computed using the harmonic mean of your homework grade and your exam grade as described below. This method places equal emphasis on homework and exams. If you get a poor grade in either homework or exams it will seriously drag your average down.

There will be a midterm and a final in this course, together with regular homeworks.

1. I will use +/- grading in the class. The grades will be assigned on the basis of your final class average based on the following ranges:

A	90 or above	C	70 to 72
A-	85 to 89	C-	65 to 69
B+	83 or 84	D+	63 or 64
B	80 to 82	D	60 to 62
B-	75 to 79	D-	55 to 59
C+	73 or 74	F	54 or below

2. To encourage improvement, I will replace earlier test scores by later test scores if they are

better. For example, if you get 50 on the midterm and 85 on the final, both scores will be counted as 85. Thus, a poor midterm grade will not hurt you as long as you improve.

3. The homework average is just the arithmetic average of the homework grades.

4. The exam average is computed as follows:

   a. ExamAvg = AdjustedMidterm/ 3 + 2*Final/3

   b. AdjustedMidterm = max(Midterm,Final).

   c. FinalAvg = HarmonicMean(HwAvg,ExamAvg);

5. All numbers are rounded and the letter grades are assigned according to the scale mentioned in bullet 1 above.

THE HARMONIC MEAN APPROACH TO GRADING

In this course, both the homework and exams are very important. The arithmetical mean that is often used to average course grades allows a good performance in one area to offset a poor performance in another. Since I want to emphasize good performance in both homework and exams, I will use the harmonic mean of your homework grade and your exam grade as your final average.

The harmonic mean is very sensitive to extremes in performance. In other words, if your homework and exam grades are close to one another, their harmonic mean is essentially the same as the usual arithmetic mean. If they are far apart, the harmonic mean is pulled sharply down toward the lower grade. Thus, students who copy their homework from others cannot use the good grades they obtain in this manner to offset their poor exam grades. The following table gives you sample values.

Final Grade				**Exam Grade**										
		40	**45**	**50**	**55**	**60**	**65**	**70**	**75**	**80**	**85**	**90**	**95**	**100**
	40	40	42	44	46	48	50	51	52	53	54	55	56	57
	45	42	45	47	50	51	53	55	56	58	59	60	61	62
	50	44	47	50	52	55	57	58	60	62	63	64	66	67
HW Grade	**55**	46	50	52	55	57	60	62	63	65	67	68	70	71
	60	48	51	55	57	60	62	65	67	69	70	72	74	75
	65	50	53	57	60	62	65	67	70	72	74	75	77	79
	70	51	55	58	62	65	67	70	72	75	77	79	81	82
	80	53	58	62	65	69	72	75	77	80	82	85	87	89
	85	54	59	63	67	70	74	77	80	82	85	87	90	92
	90	55	60	64	68	72	75	79	82	85	87	90	92	95
	95	56	61	66	70	74	77	81	84	87	90	92	95	97
	100	57	62	67	71	75	79	82	86	89	92	95	97	100

The exact formula for the harmonic mean is

$$F = 2*E*H/(E+H)$$

Here E is the exam grade and H is the homework grade. Your final grade F will be computed by using this formula exactly and rounding off to the nearest whole number.

ADDITIONAL NOTES

1. Exams are demanding, so you must know the material thoroughly! You are expected to know the definitions of all important concepts since you might be asked to write them down on the exams.

2. The midterm is listed on the schedule below. The final exam time will be announced once a time is selected by the registrar.

3. The midterm and final are closed book and in class.

4. The midterm and final are cumulative and cover all the material up to the time that they are given.

5. To ensure that you have adequate time on the midterm and final, they will be scheduled at times when ample time will be available. The details on this will be presented in class.

6. Partial credit will be given on exam problems.

7. I want people to work on the homework individually. You can talk to each other and give help, but this help should not take the form of letting other people copy your work. It is important that you understand how to do all the problems on your own. Otherwise you cannot do well on the exams. The harmonic mean grading scheme requires a good performance on both the homework and exams in order to get a good grade.

8. You will lose credit even if minor errors are spotted. Errors are the source of much mischief, so it is important to reduce them as much as possible. Even minor arithmetical mistakes will result in points being lost, so do your work carefully.

9. If a problem asks you to write a program, a function or a procedure, always submit a printed listing and output, even if the problem does not explicitly ask for these. Handwritten code is not acceptable.

10. If your programs have bugs, I expect you to make a reasonable effort to find the bug on your own. I will be happy to help you find problems in your programs, but you must come with evidence that you have tried to find the problem on your own and the program I see should have evidence of your efforts to debug it.

11. If you run out of time and must turn in a program that doesn't run, submit output showing the crash and the error message as well as a listing.

12. Be sure that your listings include comments that explain what you are doing if it is not completely obvious. It is up to you to explain what you are doing.

13. If you do not understand a problem get a clarification from me. Do not waste a lot of time working on something that you don't understand.

14. I do not accept late homework except in special circumstances. You must get permission in order to turn your homework in late. Such permission is the exception rather than the rule. Homework turned in late without permission will lose points.

15. Since this is a graduate course, I expect very high quality work from you. Points will be deducted for sloppy or disorganized work.

16. Programs must be integrated into the manuscript. In other words, don't just hand in a program listing stapled to the homework. Place it close to the text giving the answer to that particular problem. It must be clearly labeled and relevant parts highlighted. In particular, I don't want little short paragraphs that say "see PROGRAM..." with a pile of printout attached. The pages of the homework should be consecutively numbered.

17. Any program that you submit must include sample output that adequately tests it. This sample run should not be copied from the screen by hand and should not be a screen dump attached separately. If you want to use a screen dump, put it close to the relevant problem in the homework and give adequate indications of where the output can be found. Alternatively, have the output sent to a file and include the file in your manuscript. You should think carefully about what constitutes an adequate test for each program that you write. You will lose points for inadequate testing.

18. When you write programs, pay attention to the human interface. The requests for data should be reasonable. Ridiculous interfaces will lose points just for being ridiculous.

19. Be sure that you answer the question. Don't answer questions that are "almost" like the questions you are asked.

20. I expect your algorithms to be reasonably efficient. Just simply whipping something together that gets the job done might not be enough. Also, if you make modifications to algorithms, you will lose points if you make the algorithm perform significantly less efficiently from what was presented.

21. Do not scatter your work all over the place. I will not read through listings to find an analysis of an algorithm. This analysis should be in the text, not in the listing somewhere.

22. Submit all necessary pieces. I don't want to guess what data types you used, etc.

23. You will lose points for submitting poorly organized and unreadable material. I expect your homework to be stapled together not paper clipped together.

24. You will lose points for poor programming style. I do not want to see hoards of global variables in your programs. You have been taught how to do things correctly and I want to see you do it.

25. I will post current grades on my door encoded by keywords. This process will be explained in class and your keywords assigned. You should check this listing regularly to make sure that your grades have been recorded correctly.

TEXT

Computability, Complexity and Languages: Fundamentals of Theoretical Computer Science, by Martin Davis, Ron Sigal and Elaine Weyuker, Academic Press, 2nd edition, 1994. ISBN: 0122063821.

OFFICE HOURS

Office	236 Neville Hall
Office Hours	TuTh 9:30-10:45 am and by appointment Check with Wendy Maupin
Phone	581-3940
e-mail	markov on any UNET or CS Dept. Network and also on First Class

If you plan to come for office hours it is best to make sure that no unplanned event is keeping me away. You can check by either calling my number or the Computer Science Department (581-3941). If you have problems with this course and need help come in to see me immediately. Don't fool around until the end of the semester and then try to learn all the material in a week.

CLASS MEETINGS

The class will meet every Tuesday and Thursday from 8:00 until 9:15. I will have to reschedule classes when conflicts with other events will occur.

Tuesday, October 12 is FALL BREAK.

Thursday, November 25 is THANKSGIVING.

DISABILITIES

If you have a legitimate disability that is interfering with your performance in class, please speak to me about it. I will try to accommodate it as much as possible.

SYLLABUS

I plan to follow the text closely. My goal is to cover Chapters 1-4, 6 and 9-10 completely, and along with a small number of additional topics. I will be traveling some this semester so we might need to have some makeup classes or to use a video link.

Week of 08/30	Chapter 1.Preliminaries. Induction. Infinite Sets.
Week of 09/06	Chapter 2. Programs and Computable Functions.
Week of 09/13	Chapter 3. Primitive Recursive Functions.
Week of 09/20	Chapter 3. Primitive Recursive Functions.
Week of 09/27	Chapter 4. A Universal Program.
Week of 10/04	Chapter 4. A Universal Program.
Week of 10/11	Fall Break; Chapter 4. A Universal Program.
Week of 10/18	Midterm and Midterm Review.
Week of 10/25	Chapter 9. Regular Languages. Finite State Automata.
Week of 11/01	Chapter 9. Regular Languages. Finite State Automata
Week of 11/08	Chapter 10. Context-Free Languages. Push-down Automata.
Week of 11/15	Chapter 10. Context-Free Languages. Push-down Automata.
Week of 11/22	Chapter 6. Turing Machines. Thanksgiving.
Week of 11/29	Chapter 6. Turing Machines.
Week of 12/06	Quantum Computing and Other Models of Computation.

There will be a midterm and a final in this course. The midterm is tentatively scheduled for the week of 10/18. The time and date of the final will be selected by the Registrar and announced in class. The final will take place during the week of December 13.

There will be frequent homeworks. I reserve the right to ask questions about the homework to ensure that the person submitting the work actually did it.