

CSCI450: Compiler Design

1.	Course Prefix and number	CSCI 450
2.	Number of semester hours	3
3.	Pre-requisites	CSCI 321 (soon CSCI 250) and CSCI 330 as a corequisite.
4.	Location of classroom and time course meets	WS 118 9:30-10:45 Tue, Thu
5.	Intstructor Web Page	Dr. Warren MacEvoy http://www.coloradomesa.edu/~wmacevoy
6.	Office Hours	See my web page.
7.	Course Description	Structures and techniques used in compiler writing are discussed with emphasis on scanners, symbol tables, parsers and code generation. The front end of a recursive descent parser is written for the semester project. Error analysis and code optimization are discussed as time permits.
8.	General Education Objectives	This is not a general education class.
9.	Textbook and Materials	<u>Compiler: Principles, Techniques and Tools</u> , by Aho, Lam, Sethi, & Ullman, 2th edition.
10.	Course Objectives	Learning the theory and practice of accomplishing what is in the course description [:o)]
11.	Policy on absences and tardies	Make up exams will only be given with advanced notification or very unusual circumstances. You are responsible for material given in class, including changes of schedule for exams.
12.	Policy on late	Late projects cost 2 letter grades per day.

	work	
13.	Policy on academic dishonesty	You may be given a failing grade on a project or the course overall for academic dishonesty.
14.	Disability Arrangements	In coordination with Educational Access Services, reasonable accommodations will be provided for qualified students with disabilities. Please meet with the instructor the first week of class to make arrangements. Nancy Conklin, the Coordinator of Educational Access Services, can be contacted at 248-1826, or in person in Houston Hall, Suite 108.
15.	Evaluation and grading	One-third of the grade will reflect performance on assigned individual projects, one-third of the grade will be performance on steps of the group project, and one-third will be based on the final project. Projects will be graded on an A, B, or F basis: • A: Well written and documented, with more bells and whistles than required for the project. Testing procedures must be included with the project. • B: Works correctly. Reasonably written. Reasonably documented. Reasonably tested. • F: Fails to work. Poorly written, documented, or tested.
16.	Major assignments	The most important assignment will be the development of a compiler and/or interpreter.
17.	Course Topics	See below
18.	Success	General

SPRING 2011	MWF				
WEEK	MON	TUE	WED	THU	FRI
22-Aug-11		1		2	
29-Aug-11		3		4	
5-Sep-11		5		6	

12-Sep-11		7		8	
19-Sep-11		9		10	
26-Sep-11		11		12	
3-Oct-11		13		14	
10-Oct-11		15		16	
17-Oct-11	FALL	FALL		17	
24-Oct-11		18		19	
31-Oct-11		20		21	
7-Nov-11		22		23	
14-Nov-11		24		25	
21-Nov-11		26	THANKS	THANKS	THANKS
28-Nov-11		27		28	
5-Dec-11		29		30	
12-Dec-11	FINALS	FINALS	FINALS	* 8AM *	

DAY	TOPIC	Section
1	intro	1
2	syntax and translation	2.2-2.3
3	parsing and example	2.4-2.5
4	lexing and symbol table	2.6-2.7
5	intermediate code	2.8
6	test	

7	lexical analysis and buffering	3.1-3.2
8	tokens	3.3-3.4
9	lex	3.5
10	automata	3.6
11	re -> fa	3.7
12	test	
13	syntax and grammars	4.1-4.2
14	grammars and top-down parsing	4.3-4.4
15	bottom-up parsing	4.5
16	ambiguity	4.8
17	generators	4.9
18	syntax directed translation	5.1-5.2
19	syntax directed application and schemes	5.3-5.4
20	test	
21	intermediate code	6.1-6.2
22	types	6.3
23	expressions	6.4
24	type check	6.5
25	flow control/backpatch	6.6-6.7
26	procedures	6.9

27	test	
28	storage	7.1-7.2
29	non-local access	7.3
30	heap and gc	7.4-7.5