CS251 Programming Languages **Handout # 2**
Prof. Lyn Turbak **January 29, 2002**
Wellesley College

Tentative Syllabus

This is a tentative syllabus for the course. It will be updated during the semester to reflect our actual progress in the course.

The readings listed with a lecture cover (some) material in that lecture. You are encouraged to do the reading *before* the associated lecture.

Lec.	**Date**	**Topic**	**Reading**	**Problem Sets**
		Week 1		
1	T 1/29	administrivia; course overview; introduction to Scheme		#0 out
2	W 1/30	more Scheme; substitution Model; standard library/desugaring	*SICP* 1.1–1.2	
3	F 2/01	lists and list recursion	*SICP* 2.1, 2.2–2.2.2, 2.3	#0 due #1 out
		Week 2		
4	T 2/05	more list recursion; tree recursion: compositional programming		
5	W 2/06	higher-order functions 1: first-class functions	*SICP* 1.3	
6	F 2/8	higher-order functions 2: compositional programming	*SICP* 2.2.3-2.2.4	#1 due #2 out
		Week 3		
7	T 2/12	manipulating programs (INTEX)		
8	W 2/13	simple interpretation (INTEX)		
9	F 2/16	simple naming (BINDEX)		#2 due #3 out
		Week 4		
	T 2/19	**Monday Schedule; no lecture**		
10	W 2/13	conditionals and simple data (IBEX)		
11	F 2/15	desugaring (IBEX)		#3 due #4 out
		Week 5		
12	T 2/26	functions and scoping (HOFL)		
13	W 2/27	environment model (HOFL)	*SICP* 3.2–3.2.2; 4–4.2.2	
14	F 3/1	recursive bindings (HOFL)		#4 due #5 out

Lec.	Date	Topic	Reading	Problem Sets
		Week 6		
15	T 3/5	first-order functions (FOFL); block structure (FOBS)		
16	W 3/6	compound data: products, sums, algebraic datatypes		
17	F 3/8	introduction to ML	*MLWP* 2–5	#5 due midterm out
		Week 7		
18	T 3/12	ML examples		
19	W 3/13	type checking 1: types, type rules, and derivations		
20	F 3/15	type checking 2: implementing type checking		midterm due
21	3/16–24	**Spring Break**		
		Week 8		
22	T 3/26	type reconstruction 1: overview, substitution, unification		#6 out
23	W 3/27	type reconstruction 2: rules, examples		
24	F 3/29	type polymorphism		
		Week 9		
25	T 4/01	imperative programming: state, explicit cells (HOILEC)	*SICP* 3.2.3	
26	W 4/03	imperative programming: implicit cells (HOILIC), imperative interpreters		
27	F 4/05	classical imperative programming: C, Pascal		#6 due #7 out
		Week 10		
28	T 4/09	memory management		
29	W 4/10	parameter passing/laziness	Hughes paper; *SICP* 3.5, 4.2; *MLWP* 5.12–5.20	
30	F 4/12	introduction to Haskell	*HCFP* 10, 17	#7 due #8 out

Lec.	Date	Topic	Reading	Problem Sets
Week 11				
31	T 4/16	Haskell examples	*HCFP* 18	
32	W 4/17	control 1: non-local exits		
33	F 4/19	control 2: exceptions		#8 due #9 out
Week 12				
34	T 4/23	control 3: continuations		
35	W 4/24	non-deterministic programming	*SICP* 4.3	
36	F 4/26	logic programming 1	*SICP* 4.4	#9 due #10 out
Week 13				
37	T 4/30	logic programming 2		
	W 5/01	**Ruhlman conferene; no lecture**		
38	F 5/03	object-oriented programming 1	*SICP* 2.4, 2.5	
Week 14				
39	T 5/07	object-oriented programming 2		
40	W 5/08	**(Last class)** CS251 Jeopardy!		#10 due