1. **Number and title of course**: EECS 61C, Machine Structures

2. **Course objectives**: CS 61C brings students through a series of abstractions from high-level programming through machine architecture to logic design. The C programming language, MIPS assembly language, and schematic diagrams are used to introduce the abstractions. The course closely follows the Patterson and Hennessy textbook, supplemented by material on the C programming language and notes on the design of synchronous digital systems. A sample week-by-week outline follows.

3. **Topics covered:**
   - Week 1: Introduction; Number representation (P&H 1, 4.1-4.2)
   - Week 2: C programming language (K&R 1-4)
   - Week 3: Memory management (K&R 5.1-5.10)
   - Week 4: Assembly language operations and control flow (P&H 3.1-2, 3.5)
   - Week 5: Assembly language pointers, procedures (P&H 3.8-3.14)
   - Week 6: Machine language representation (P&H 3.4, 3.8)
   - Week 7: Floating point representation/operations (P&H 4.8, 4.10, 4.12-13)
   - Week 8: Input/output and interrupts (P&H 8.3, 8.5, 8.9)
   - Week 9: Disk storage; Networks
   - Week 10: Cache memory systems (P&H 7.1-3)
   - Week 11: Virtual memory (P&H 7.4-7)
   - Week 12: Logic elements and circuits (P&H B, P&H 4.5)
   - Week 13: Logic elements and circuits (P&H B, P&H 4.5)
   - Week 14: CPU organization (P&H 5.1-3)
   - Week 15: Pipelining (P&H 6.1-2)

4. **Relationship of course to program objectives**: Biweekly independent projects reinforce students’ programming skills while reinforcing their understanding of machine architecture and implementation concepts. Sample projects include: a machine language interpreter written in C; a cache simulator written in C; and a CPU hardware design and simulation.

5. **Prepared by**: John Wawrzynek (3/26/06)