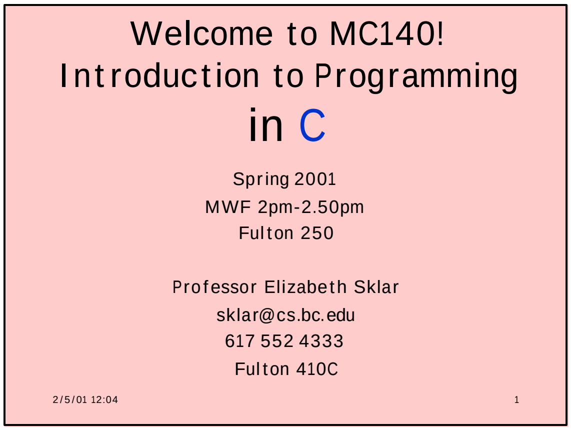
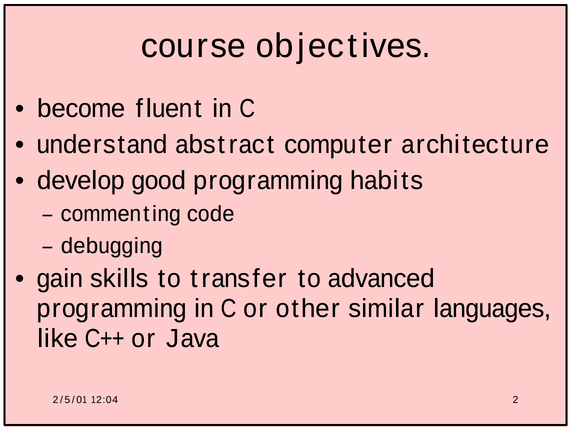


Welcome to MC140! Introduction to Programming in C

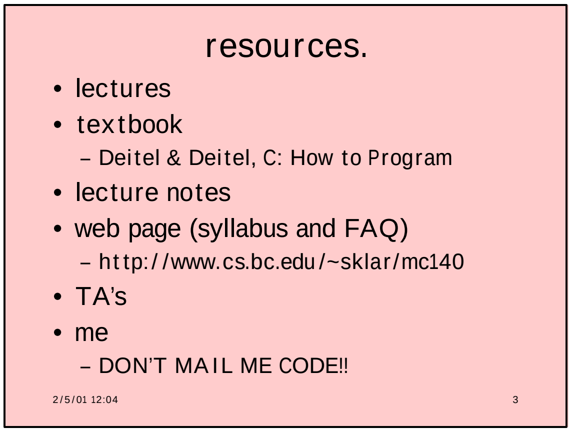
Spring 2001
MWF 2pm-2.50pm
Fulton 250

Professor Elizabeth Sklar
sklar@cs.bc.edu
617 552 4333
Fulton 410C



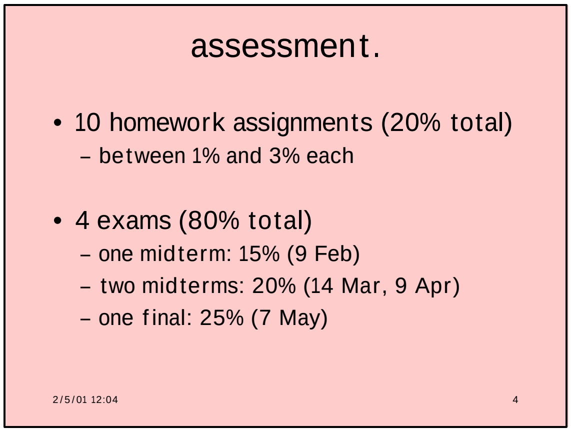
course objectives.

- become fluent in C
- understand abstract computer architecture
- develop good programming habits
  - commenting code
  - debugging
- gain skills to transfer to advanced programming in C or other similar languages, like C++ or Java



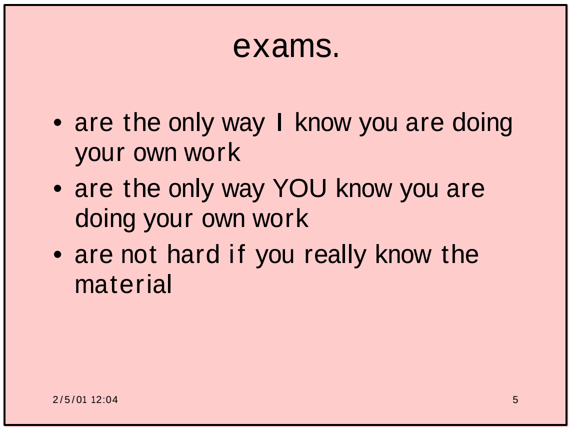
resources.

- lectures
- textbook
  - Deitel & Deitel, C: How to Program
- lecture notes
- web page (syllabus and FAQ)
  - http://www.cs.bc.edu/~sklar/mc140
- TA's
- me
  - DON'T MAIL ME CODE!!



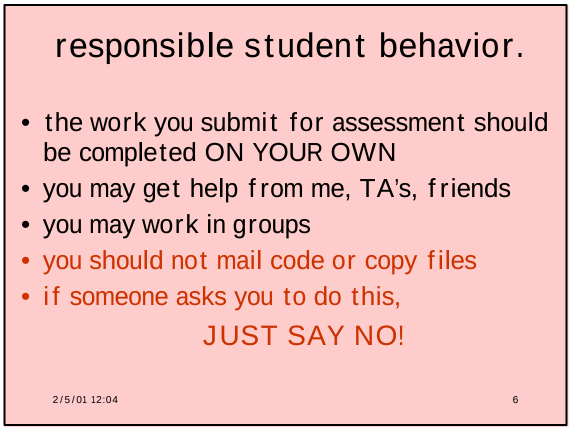
assessment.

- 10 homework assignments (20% total)
  - between 1% and 3% each
- 4 exams (80% total)
  - one midterm: 15% (9 Feb)
  - two midterms: 20% (14 Mar, 9 Apr)
  - one final: 25% (7 May)



exams.

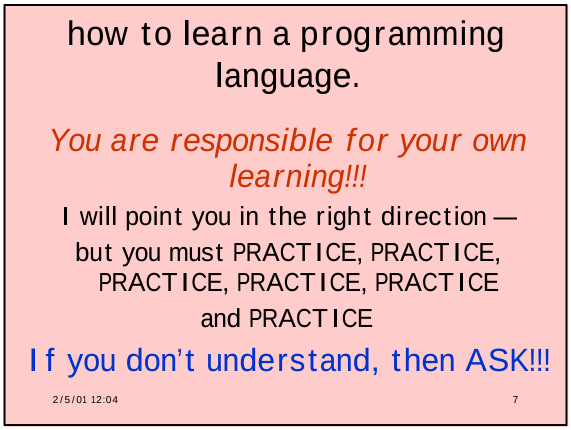
- are the only way I know you are doing your own work
- are the only way YOU know you are doing your own work
- are not hard if you really know the material



responsible student behavior.

- the work you submit for assessment should be completed ON YOUR OWN
- you may get help from me, TA's, friends
- you may work in groups
- you should not mail code or copy files
- if someone asks you to do this,

JUST SAY NO!

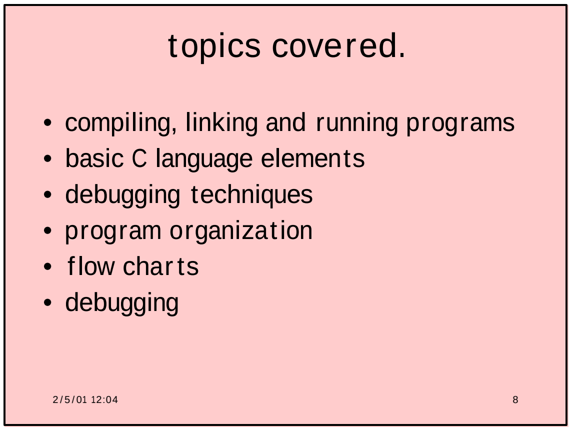


how to learn a programming language.

You are responsible for your own learning!!!

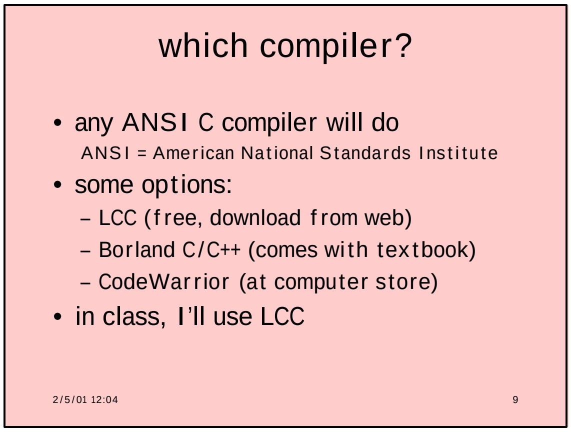
I will point you in the right direction — but you must PRACTICE, PRACTICE, PRACTICE, PRACTICE, PRACTICE and PRACTICE

If you don't understand, then ASK!!!



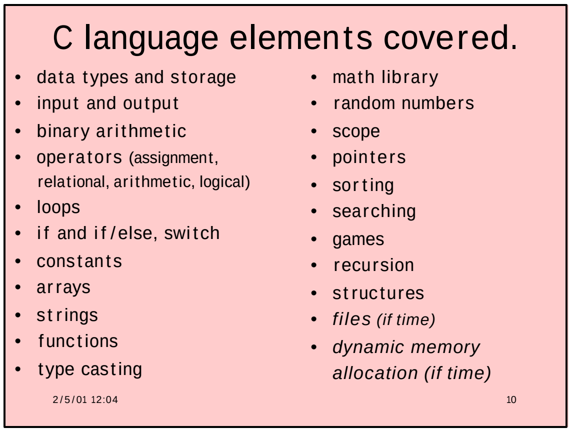
topics covered.

- compiling, linking and running programs
- basic C language elements
- debugging techniques
- program organization
- flow charts
- debugging



which compiler?

- any ANSI C compiler will do
  ANSI = American National Standards Institute
- some options:
  - LCC (free, download from web)
  - Borland C/C++ (comes with textbook)
  - CodeWarrior (at computer store)
- in class, I'll use LCC



C language elements covered.

- data types and storage
- input and output
- binary arithmetic
- operators (assignment, relational, arithmetic, logical)
- loops
- if and if/else, switch
- constants
- arrays
- strings
- functions
- type casting
- math library
- random numbers
- scope
- pointers
- sorting
- searching
- games
- recursion
- structures
- *files (if time)*
- *dynamic memory allocation (if time)*