P2 (second team project) = 100 pts

+-----------------------------------------------------+

| There will be two reports: one covering steps 1-3 |

| (due on the step 3 due date) and the other covering |

| the completed project (due on the step 5 due date). |

| No submissions are required for steps 1, 2, and 4. |

+-----------------------------------------------------+

P2 is expected to be a more "polished" game than P1.

(No use of "alerts", for example.) Furthermore, the

clarity of the design and code is important. (See

step 2 instructions.)

There are five steps in six weeks. (Step 4 is two weeks

long because Apr 14 & 17 are spring recess.)

Step 1 (due Mon, Mar 27): Specifications

Step 2 (due Mon, Apr 03): Design and Plan

Step 3 (due Mon, Apr 10): Build (1st iteration)

Step 4 (due Mon, Apr 24): Build (2nd and 3rd iterations)

Step 5 (due Mon, May 01): Final Report

Project demonstrations are MWF, May 1, 3, and 5.

STEP 1

======

Settle on your team and produce a game specification. To

properly guide the later steps, this should be well thought

out. Refer to the "list of things to consider" given in the

instructions for P1 step 1. End with a comprehensive list

of "features". Features will be the starting point for the

design (step 2). If you are building on the first project,

distinguish between old and new features.

STEP 2

======

The implementation should be easy to understand and easy

to maintain (fix bugs and add new features). Organize the

program as a hierarchy of high-to low-level units, where

each unit can be readily understood (and implemented)

without having to know details about other units. High

level units are constructed of multiple lower-level units.

The lowest level units are implemented with code and form

the elemental components of the game. Include diagrams

to illustrate relationships among the units.

STEP 3

======

Divide the design and implementation work into phases.

Do the first phase. Turn in your "mid-way" report.

STEP 4

======

Update the plan and do the 2nd and 3rd phases. Identify

and complete any needs for refactoring the code. Keep

documentation and implementation in sync.

STEP 5

======

Finalize the code. Write report on your experiences. Give

credit where it is due. Prepare for demonstration. Turn

in your final report and an executable copy of the game.