Programming

- Why is programming fun?
  - Fourth is the joy of always learning, which springs from the non-repeating nature of the task. In one way or another the problem is ever new, and its solver learns something: sometimes practical, sometimes theoretical, and sometimes both.

Source: Frederick P. Brooks, Jr. *The Mythical Man-Month: Essays on Software Engineering.*

Announcements

- Project 2
  - Take a story
    - Public domain or you wrote it
  - Take user input from a form
  - Replace words in the story with words supplied by the user

Announcements

- Project 2
  - Read all the instructions, including the rubrics at the end, before you begin!
    - Don't just start blazing away!
  - The section just before the rubrics lists the deliverables for Project 2A and for Project 2B

Whole Picture

Solving large problems is tough -- but approach them logically and you will succeed

Problem Solving

Large problems share many properties:

- They are daunting -- there's so much to do!
- We don't know where to begin
- Not sure we know all of the tasks that must be done to produce a solution
- Not sure we know *how* to do all of the parts -- new knowledge may be required
- Not sure it is within our capability -- maybe an expert is needed

Assume you will succeed; not trying concedes defeat

Problem Decomposition

"Divide and conquer" is a political strategy, military strategy, & IT strategy

Top-level Plan--(Project 2A.2)

1. Describe (in any language) a series of steps that produce a solution
2. For each step, solve it or decompose further
3. For steps needing decomposition, repeat 2
4. Assemble solutions and test correctness
5. When solution fully assembled, evaluate

More Specifics

We will step through the process, using Project 2 as an example:

- Problem decomposition is mostly common sense
- Process is not algorithmic
- Problem decomposition is to help you, so apply it as needed

1. Give Steps to a Solution

Specify (in any language) a series of steps that produce a solution

- For a huge problem the steps may at first be vague, but they can be (& must be) made more precise as the whole picture emerges
- The goal is an algorithm(s), so …
- List & describe the inputs
- List & describe the outputs
- Be guided in figuring out the steps by the need to transform the inputs into the outputs
  - Correct answers, student's choices, total score

You will be naming things

What Are Steps for Quiz?

Steps

- Student as Teacher—Creating an Online Quiz (150 points)
  - 2A: Creating the GUI in HTML (25 points)
  - 2B: Scoring the Quiz (125 points)

Project 2A

- 2A.1 Creating the GUI
  - Write questions and answers
    - Choose a subject you know well
  - Create the GUI in HTML
    - Eight fill-in-the-blank questions
    - Add mouseover effects (rollover) to an image
- 2A.2
  - Write a planning document
    - Plan your coding strategy
    - Write in narrative form what your coding will do for the entire project

Project 2B

- Part 2B: Scoring the Quiz
  - Score eight fill-in-the-blanks from 2A
  - Write and score two multiple-choice questions
    - One with one answer
    - One with several answers
  - Score the quiz with JavaScript
  - Print the total score to the page
  - Depending on score, a new page opens (Study more! or Good work!)
  - Write a reflection paper on the project

What Are Steps for Quiz?

Project 2A

- Build basic GUI
  - With 8 textboxes for each answer
  - Add questions to each textbox
  - Add a submit button
  - Add an image with a rollover (mouseover event)
  - Add any instructions needed by the user
  - Primp design & make cool looking
- Write planning document
  - Decompose the coding for Project 2B
  - Write a narrative explaining your coding strategy

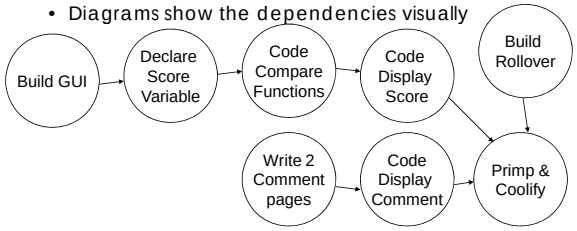
Steps for Quiz

- Part 2B: Scoring the Quiz
  - Create an array of correct answers
  - Create a variable to hold the student's score
  - Write a function to compare the student's answer with the correct answer.
  - Create multiple-choice questions
    - Radio buttons for one answer
    - Checkboxes for several answers
  - Create 2 HTML pages:
    - Study More!
    - Good Work!

PERT

PERT is Program Evaluation & Review Technique … developed in 1950s

- Diagrams show the dependencies visually



2&3. Solve or Decompose

For each step, solve it or decompose it further, i.e. apply same technique

- Most "top level" steps can't be brained out, and need further decomposition
- "Top level" steps often seem huge, too
- The technique allows one to concentrate on only one problem at a time
- As before, focus on inputs, outputs, process to transform inputs into outputs

Often, "last" decomposition done during solution

Inputs & Outputs

- Inputs
  - Array of quiz answers
  - User input from form
  - Click event on submit button
  - Mouseover on rollover image
- Outputs
  - Final score
  - Comment pages
    - Good job!
    - Study More!
  - Change bgcolor based on score

2&3. Solve or Decompose

"Code compare functions"

- Build onSubmit event handler
- Access student answers from form inputs
- Compare correct answers in array with student answers from form

Need to learn about
- accessing elements in array
- accessing student answers from form inputs

4. Assemble Parts

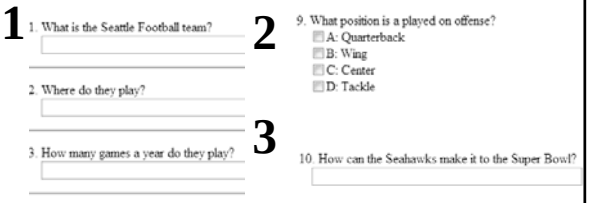
Assemble Solutions &Test Correctness

- Putting solutions together can be tough because of different assumptions made while solving the parts -- it *always* happens
- When working alone it is common to combine parts along the way and to test continuously
- Because of the need to test, pick a good order to solve the problems

Getting something working quickly is best

4. Assemble Parts

Project 2 solves & assembles parts together in a 'good' order



4. Assemble Parts

Project 2 solves & assembles parts together in a 'good' order

- Most parts of Project 2 use the developing solution for testing -- that's 'good'
- Notice adding steps to test a solution may be wise
- Parts mismatch is common problem, but not in Project 2

Summary

Large problems can be solved by the 'divide and conquer' technique

- The process is "top down" -- get a top level solution even if it is vague, imprecise
- Whenever you cannot produce a solution to a step directly, reapply the technique
- The start and first several steps will be daunting … but the process works!
- Get part of solution working quickly if possible

Reflection Paper

- Write for ten minutes on this topic:
  - Compare and contrast the use of HTML and JavaScript for Web publishing