

Comments on System Development: From Problem to Program

SFWR ENG 2B03

2003

Robert L. Baber

Goal

To conceive, plan, design, implement, document, etc. a system to satisfy stated requirements

- effectively
- efficiently
- i.e. with a reasonable expenditure of time
- i.e. without sleepless nights

The goal is to develop and document a system,
not write a program

Nongoals

The goal is not

- to see how much time we can spend on the project
- to take as much time away from other courses as possible
- to spend as much time as possible in the lab
- to spend as much time as possible playing with our beloved computers

KISS

Keep It

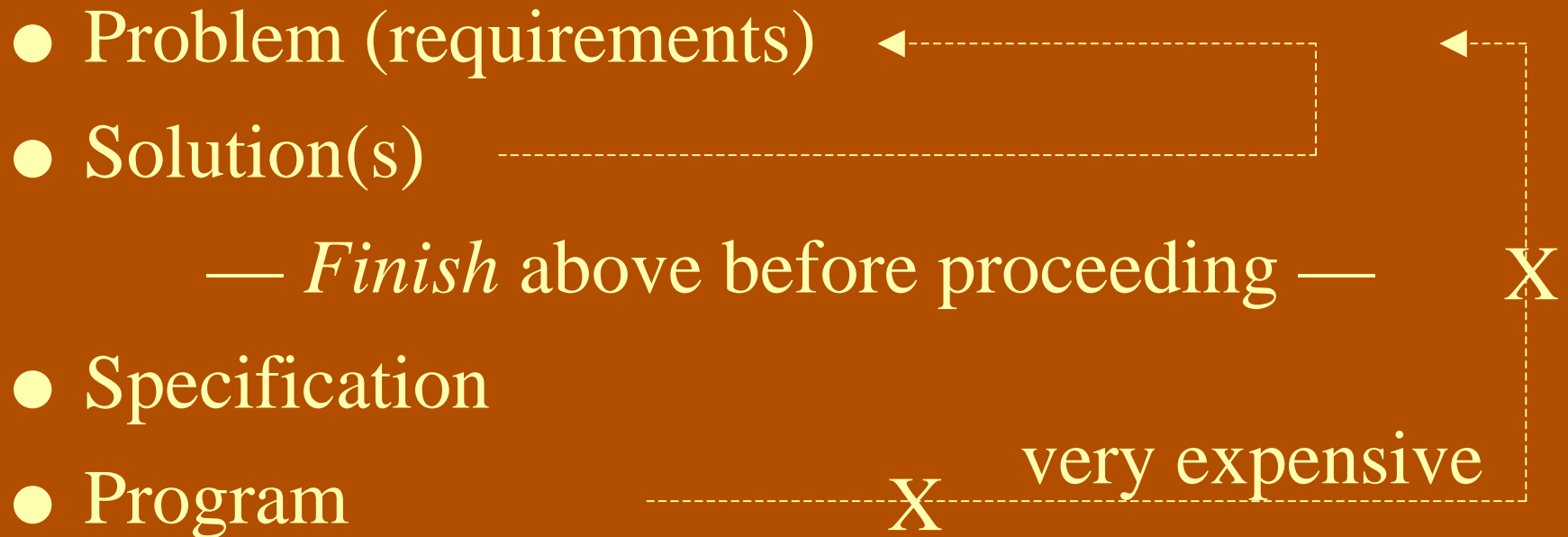
- Simple
- Small
- Systematic

Avoid unnecessary complexity

Thorough, complete analysis early pays off later

Think work early, mechanistic work later

Development phases



Start with a *thorough* analysis and *finish* it

Don't start coding too early; it will almost always turn out to be a waste of time

Subdividing a system into components

Subdivide a system into subcomponents so that

- cohesion within a subcomponent strong
- coupling between subcomponents weak

i.e.

- parts within a subcomponent closely related
- much data shared within subcomponent
- little data or few categories of data transmitted between subcomponents

Subdividing a system into components

Identify data needed throughout the entire system

Identify data needed only in parts of the system,
for only some of the system's functions

- such data often suggests subcomponents

I.e. subdivide functionally and with regard to
data needs (usually closely related)

Subdividing a system into components

In my experience useful question to pose:

If I could have any programming language
but with only about 5 commands,
which ones would I want?

They are my subcomponents.

Examples for class discussion

- Game playing program
- Student registration system
- Nuclear reactor control system

Goals, inputs, outputs, state data?