

CS 350, slide set 9a

M. Overstreet
Old Dominion University
Spring 2005

Outline

- What to read?
- What to implement?
- What to stub?
- What should instructor provide?

Background reading

- Read pp. 1-17
- Particularly, make sure you understand:
 - Rotation of History Variables (p. 15)
 - In general, you are required to keep copies of previous values of several variables just in case
 - Anytime you compute a new value, throw away the oldest one
 - Exception Handling (p. 16)

A Little FORTRAN

- REAL*8 means double
 - Look in .h files to see all C++ equivalents
- If an exception occurs the spec says to print:
 - What the exception is (see spec)
 - The name of the module,
 - The name of the function within that module
 - The current value of frame counter
- The The FORTRAN statement
FORMAT(x, a6, x, a32, x, i4)
means to print an ASCII string (module name), another
ASCII string (function name) and an integer (frame
counter)

Driver?

- Calls other modules
- What's tricky?
 - Calling order (page 43)
 - Termination
 - What about GCS_SIM_RENDEZVOUS?
 - Stubbed

AECLP (p. 47)

- 5 pages
- Controls big engines (axial)
- Similar to RECLP
- Has some "messy" math
 - Integration
 - Solve differential equation for te_limit

ARSP (p. 53)

- 2 pages
- Use radar to find height
- Math:
 - Need to fit third-order polynomial to existing data
- Some prior knowledge

ASP (p. 55)

- 4 pages
- Measure acceleration (in all 3 directions)
- Math:
 - Matrix multiply
- Error handling

CP (p. 59)

- 4 pages
- Prepare data packet for transmission
- Coding
 - Packing different data types on byte boundaries
- A little prior knowledge

CRCP (p. 63)

- 1 page
- Release parachute

GP (p. 65)

- 7 pages
- Fly the space craft
- Math
 - Integration
 - Matrix multiplies
- Complex!

GSP (pg. 74)

- 2 pages
- Measure rotation rates using gyroscopes

RECLP (p. 75)

- 2 pages
- Control roll engines

TDLRSP (p. 77)

- 5 pages
- Measure velocity with 4 radar beams
- Issues
 - Error handling?

TDSP (p. 83)

- 1 page

TSP (p. 85)

- 3 pages
- Math
 - Find equations of quadratic

Assume 4 member groups

- Which modules to implement
- What can be stubbed?
- What routines should be provided?
 - Due to lack of math prerequisites for class

Required modules

- Driver
- Stubbed routines for any not written
 - E.g. GP, GCS-sim-rendezvous
- Modules:
 - ARSP
 - ASP
 - CP
 - GSP
- Extra credit for extra modules
 - How much depends on difficulty of module; please inquire

Simplified Task Planning Template (p. 82)

- Rows: include **only**:
 - Launch & strategy
 - Plan-tasks and schedule
 - System test plan
 - Only one line** per module (but see next slide)
 - Postmortem
 - Totals
- Columns: include **only**:

| | |
|----------------------|------------------|
| Task Name | Planned Value |
| Plan hours by person | Cumulative PV |
| Total team hours | Actual hours |
| Cumulative hours | Cumulative hours |
| Size | Actual Week |
| Planned Week | |

Task Plan/Report per Module

| Task | Size | Who | Plan-hrs | PV | Act-hrs | Plan com. date | Act. com. date | EV |
|-------------|------|-----|----------|----|---------|----------------|----------------|----|
| Design | | | | | | | | |
| Design rev | | | | | | | | |
| Code | | | | | | | | |
| Code rev. | | | | | | | | |
| Compile | | | | | | | | |
| Code insp. | | | | | | | | |
| Test data | | | | | | | | |
| Test driver | | | | | | | | |
| Test rev. | | | | | | | | |
| Test insp. | | | | | | | | |
| Testing | | | | | | | | |

Another omission

- Given the limited time:
 - Log where errors are **found** only
 - Omit noting where errors are inserted
- You must report errors found for each task
 - Some tasks may be unlikely to discover errors, but also include them in your error report.

Minutes due by Sunday!

- For this meeting, do **not** use week form
 - Will be used after this week
- Include:
 - Team Name
 - Chosen regular team meeting time
 - Tentative decision on who will implement what
 - Team questions
