Administrative stuff

- Weekly reports and minutes
  - See web
- Library web page/section
  - Weekly reports, minutes, deliverables
- Due Thursday (start of class)
  - Preliminary Report
  - Project Web I
The Development Process
Fundamental Components

- Specification
  - Requirements (Thursday)
- Design and Implementation
  - HLD, DD, OOD (Jan 25)
- Validation & Verification
  - Quality assurance, testing, etc. (Feb 1)
- Evolution
Real Benefits

Effort Estimation

Compile & Test Defects
Waterfall Model
Winton Royce, 1970
Problems with the Waterfall

- Assumes a single pass/path
  - Feedback to preceding task only
- Assumes builds all at once
  - Work on pieces, then assemble
Evolutionary Development
For small systems

Outline description
Concurrent activities
Specification
Development
Validation
Concurrent activities

“One to throw away…”
Spiral Development
Boehm, 1988

- Loop with main tasks of
  - Objective setting
  - Risk assessment
  - Development and validation
  - Planning
Scheduling Software Development
Time, time, time...

- Poor techniques
  - undue optimism—"this should do it"
- People and time
  - people & techniques often assume they are interchangeable
- Discipline
  - uncertainty leads to drifting, creeping, etc.
- Observability
  - schedule progress needs to be monitored
- Trouble
  - natural to add people when schedules slip
The Man-Month

- Cost $\equiv$ people $\times$ time $\neq$ progress
- "Man-month" as a unit of measure of job size
  - Normally dangerous and deceptive
  - OK when people/months interchangeable (fig 2.1)
- Not partitionable, result is constant (fig 2.2)
- Partitionable w/ inter-communication (fig 2.3)
- Tasks w/ complex inter-relationships (fig 2.4)
System Test

- Qty & complexity/subtlety of bugs -> time
- Optimism
  - implies few bugs so usually mis-scheduled
- Brooks formula (ROT)
  - 1/3 planning
  - 1/6 coding
  - 1/4 component and early system test
  - 1/4 system test (integrated)
- Few people allow as much testing
- Test problems can be disastrous
  - Too late, costly
Gutless Estimating

- Programmer like chef
  - Patron controls the schedule not the actual
  - Under-cooked or uneven (burned and raw)
- Brooks’ Law [MMM]
  - “Adding manpower to a late software project makes it later.”
- Regenerative schedule disaster
  - (Northrop cartoon)
Brooks’ example

- Task assumptions
  - 12 man-months: 3 people, 4 months
  - Measurable milestones A, B, C, D
- If milestone A is late by one month...
Done on time, A only problem

- 9 MM remain in 2 months
- Need 4.5, so add 2 (5 total)
- (MMM Fig 2.6)
Done on time, all are problems

- 18 MM remain in 2 months
- Need 9, so add 6
- (MMM Fig 2.7)
Reschedule or Trim

- **Reschedule**
  - Allow enough time to complete the work
  - “No small slips” (P. Fagg)

- **Trim the task**
  - Judicious cuts
  - Will happen anyway…
Training New People

- Even smart people need training
- Takes time of new people
- Takes time of one (or more) current
Scheduling Tools
Gantt & PERT Charts

- Visualization tools
- Used by project managers
- Control and administer tasks
Gantt Chart

- Charles Gantt, 1917
- Sequences of tasks (mfgr)
- Time -> horizontal, bars for tasks
- Dependencies represented with arrows
- Resources also listed
PERT Chart

- Program Evaluation and Review Technique
- Developed in 1950’s by US Navy
- Dependencies -> critical path
- MS Project can make
AB  Define class interfaces
BC  Write class "String"
BD  Write class "Search Engine"
BE  Write class "Image"
BF  Write class "Web Page"
CG DG EG FG - Integration
GH Testing