Today

- Estimation (cont.)
- SE Ethics

Reminders

- Proj presentations (8 min., Thurs)
- HW 8 due Thurs.
- Proj reports (due at Final)
- Final is Wed, 5/9 at 3:30-5:30
What is being estimated?

Estimation questions ...

- Why worry about accurate estimates?
- Where do estimation errors come from?
- How should estimates be presented?
- What is being estimated?

Q: What factors most influence effort, cost, schedule?

1. Software size is the most significant determinant
2. The kind of software you are developing is second
3. Personnel factors are a close third
Project Size

Projects exhibit “Diseconomies of scale” ...

Larger the system, the more effort needed to build it!

- e.g., 10 times larger system doesn’t imply 10 times more effort
- Q: Why?
  - larger projects require coordination among larger groups of people
  - requires increasingly more communication links (increases exponentially)
  - thus effort scales non-linearly

Estimating system size

- typically in **early and middle stages** of sequential projects (waterfall, etc.)
- iterative approaches estimate by **features** ... and how many per iteration
- examples of **measures** (metrics) for project size
  - lines of code
  - function points
  - GUI components
  - features, user stories, story points
  - web pages
  - DB tables
  - classes
  - ...
Lines of Code (LOC)

- as “cost/measure” or “measure/cost” ... e.g., “LOC/day”
- 10,000 LOC ... size estimate
- 100 LOC/Day ... speed estimate
- $100 \text{ LOC/Day} \times x \text{ Days} = 10,000$ ... schedule estimate ($x = 100 \text{ Days}$)

NOTE: effort per line of code has been found to be roughly constant across most programming languages (... close enough)

Q: What might be positives for using LOC?

- easy to collect for past projects
- can more easily allow for cross-project comparisons (lower-level than stories)
- can make estimating future projects from past ones easier

Q: What might be negatives for using LOC?

- don’t often include upstream activities
- hard to estimate directly (typically based on “proxies”)
- simple models like “LOC / dev day” are error-prone:
  - different coding rates for different kinds of software
  - programmers can differ in productivity
- determining what constitutes a line of code ...
  - all code, or only code included in the release
– how to count reused code from other projects
– blank lines and comments?
– class interfaces, data declarations?
– one logical line of code split across lines? multiple on one line?

LOC is still the **primary way** companies use to estimate software size
Function Points

A synthetic measure of program size

- based on certain “proxies” (inputs, outputs, queries, files, ...)
- Maintained by International Function Point Users Group (www.ifpug.org)

Easier to estimate from requirements than LOC

Number of function points in a program based on number and complexity of proxies

- External inputs: any input to program (user or program), e.g., screens, forms, dialog boxes
- External output: any output of program (user or program), e.g., screens, graphs, reports
- External queries: input/output combinations, e.g., from a DB
- Internal logical files: completely controlled by system
- External interface files: controlled by other systems, information that enters or leaves program
# Estimation using Function Points

Determine the project’s total “Function Point Count”

<table>
<thead>
<tr>
<th>Program Characteristic</th>
<th>Low Complexity</th>
<th>Med Complexity</th>
<th>High Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>External inputs</td>
<td>___ × 3 = ___</td>
<td>___ × 4 = ___</td>
<td>___ × 6 = ___</td>
</tr>
<tr>
<td>External outputs</td>
<td>___ × 4 = ___</td>
<td>___ × 5 = ___</td>
<td>___ × 7 = ___</td>
</tr>
<tr>
<td>External queries</td>
<td>___ × 3 = ___</td>
<td>___ × 4 = ___</td>
<td>___ × 6 = ___</td>
</tr>
<tr>
<td>Internal logical files</td>
<td>___ × 4 = ___</td>
<td>___ × 10 = ___</td>
<td>___ × 15 = ___</td>
</tr>
<tr>
<td>External interface files</td>
<td>___ × 5 = ___</td>
<td>___ × 7 = ___</td>
<td>___ × 10 = ___</td>
</tr>
</tbody>
</table>

**Unadjusted Total:**

**Influence Modifier:**

**Adjusted Total:**

Computing an “**Influence Modifier**”:

- based on the influence of 14 factors on the program
- e.g., data communications, processing complexity, etc.
- multiplier ranges from 0.65 to 1.35
Given an adjusted total can compute **effort** (in dev months)

- often based on type of project and team size

- for example:

**General Project:**

\[ \text{DevMonths} = 0.512 \times \text{FunctionPoints}^{0.392} \times \text{MaxTeamSize}^{0.791} \]

**Enhancement:**

\[ \text{DevMonths} = 0.520 \times \text{FunctionPoints}^{0.385} \times \text{MaxTeamSize}^{0.866} \]

Can also convert Function Points to LOC

<table>
<thead>
<tr>
<th>Language</th>
<th>Min (-1 stddev)</th>
<th>Mode</th>
<th>Max (1 stddev)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java</td>
<td>40</td>
<td>55</td>
<td>80</td>
</tr>
<tr>
<td>C++</td>
<td>40</td>
<td>55</td>
<td>140</td>
</tr>
<tr>
<td>C#</td>
<td>40</td>
<td>55</td>
<td>80</td>
</tr>
<tr>
<td>Assembly</td>
<td>130</td>
<td>213</td>
<td>300</td>
</tr>
<tr>
<td>SQL</td>
<td>7</td>
<td>13</td>
<td>15</td>
</tr>
</tbody>
</table>

- e.g., if estimate is 284 function points, then \(284 \times 55 = 15,675\) LOC

We’ve only scratched the surface ...

- many other metrics, formulas, approaches for estimating projects

What you should remember when estimating:

- Count, compute, then judge
- Beware expert judgement (especially when used alone)
- Always give estimation uncertainty
Exercise: ACM Code of Ethics

Take a few minutes and read over the ACM Code of Ethics

Q: Why have a professional code of ethics?

- state and clarify ethical requirements of profession
- holds the profession accountable to the public ... to yield public trust
- help individual decision making (wrt ethical issues)

Sections:

- 1 are general principles
- 2 are more specific principles (related to computing)
- 3 are for those in leadership roles
Scenario 1

A software company has just produced a new program to compute taxes for small businesses based on recent tax law changes. The company president knows the program has bugs, but believes the first firm to put out this type of program will likely capture the largest future market share. When the company ships the product, they include a disclaimer of responsibility for errors resulting from use. The company expects it will receive a number of complaints and suggestions, which it plans to use in future updates. The president argues this is general industry policy and that anyone who buys version 1.0 of a program knows this and will take proper precautions. However, because of bugs, a number of companies filed incorrect tax returns and were penalized by the IRS.

Q: Which principles in the ACM code of ethics were violated?

- Several ...
- avoid harm to others (1.2)
- did not strive to achieve highest quality work (2.1)
- failed to report bugs in the system (2.5)
- possibly 2.3 (respect existing laws) since a disclaimer can only be used when the company “is in good conscience”
- president violated 3.1 since he did not encourage his staff to accept responsibility to (the welfare of) society
Scenario 2

A small software company is building an integrated inventory control system for a large national electronics manufacturer. The system will gather sales information daily from stores nationwide, which will be used to control various functions within the company. The inventory functions are critical to the smooth operation of the system.

Jane, a quality assurance engineer with the software company, suspects that the inventory functions are not sufficiently tested, although they have passed all their contracted tests. She is being pressured by her employers to sign off on the software. Legally she is only required to perform those tests which had been agreed to in the original contract. However, her experience has led her to be concerned over risks of the system. Her employers say they will go out of business if they do not deliver the software on time. Jane contends that if the inventory subsystem fails, it will significantly harm their client and its employees.

Q: Any ethical issues here? What should they do?

- avoid negative consequences to others (1.2)
- honesty about system, provide full disclosure of limitations and problems (1.3)
- strive to achieve the highest quality (2.1)
- at the least, the client should be informed about Jane’s reservations
Scenario 3

Diane is a consultant hired to design a database app for the personnel office of a medium-sized company. She has involved the client in the design process, informing the CEO, the director of computing, and the director of personnel about the progress of the system. It is now time to make decisions about the kind and degree of security to build into the system. Diane has described several options to the client. Because the system is going to cost more than they planned, the client has decided to opt for a less secure system. She believes the information they will be storing is extremely sensitive. It will include, e.g., performance evaluations, medical records for insurance claims, salaries, etc.

With weak security, employees using the system may be able to figure out ways to get access to this data, not to mention the possibilities for online attacks. Diane feels strongly the system should be much more secure, and has tried to explain the risks to the company. However, the CEO, director of computing, and the personnel director agree that less security will do.

Q: What are the ethical issues, and what should she do?

- respect privacy of others (1.7) ... “it is the responsibility of professionals to maintain the privacy and integrity of data describing individuals.”
- those in decision-making positions should “verify that systems are designed and implemented to protect personal privacy ...” (3.5)
- similarly, organization leaders have obligations to assess and validate the needs of all those affected by the system (3.4)
- Diane has a responsibility to educate on the consequences (2.7)
- Depending on what happens, she may need to choose between her contractual obligation (2.6) and her obligation to honor privacy
Scenario 4

A software consultant is negotiating a contract with a local community to design their traffic control system. He recommends they select the TCS system out of several available systems on the market, which are each developed by the same software company. He fails to mention that he is a major stockholder of the company producing TCS software.

Q: Any ethical issues here?

- Computer professionals must be objective when evaluating, recommending, and presenting system descriptions and alternatives (2.5)

- A computer professional must be honest about “any circumstances that might lead to conflicts of interest” (1.3)