

CMPUT 401

Software Process and Product Management

Ildar Akhmetov

ildar@ualberta.ca

Department of Computing Science

University of Alberta

Estimation and Planning

Fall 2020

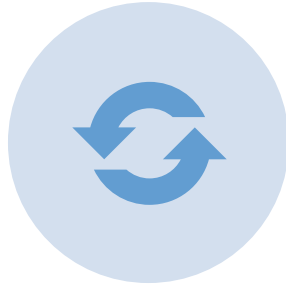
The Project Management Triangle



Types of Planning



RELEASE
PLANNING



ITERATION
PLANNING



RISK
PLANNING

Uncertainty Space

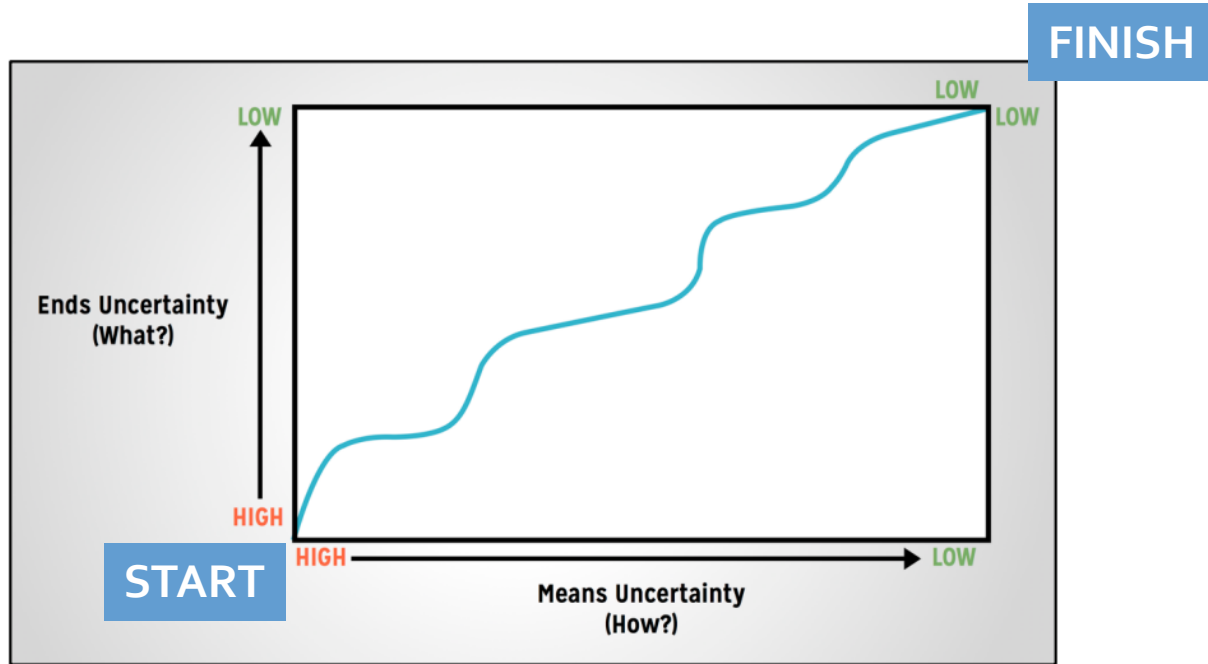
Ends uncertainty

- What are you creating in the product?

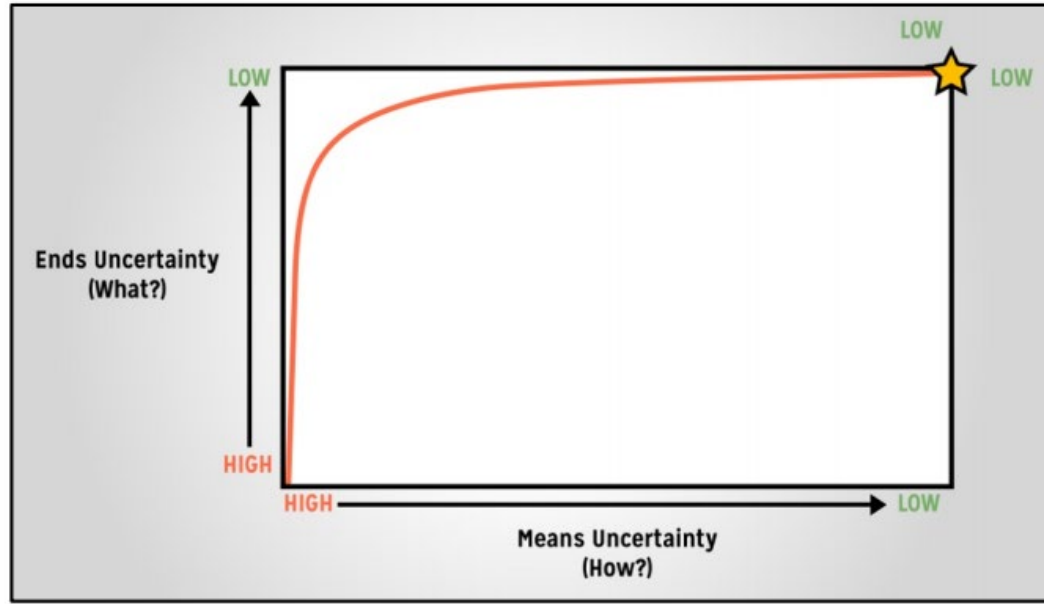
Means uncertainty

- How will the product be developed?

Uncertainty Space Diagram

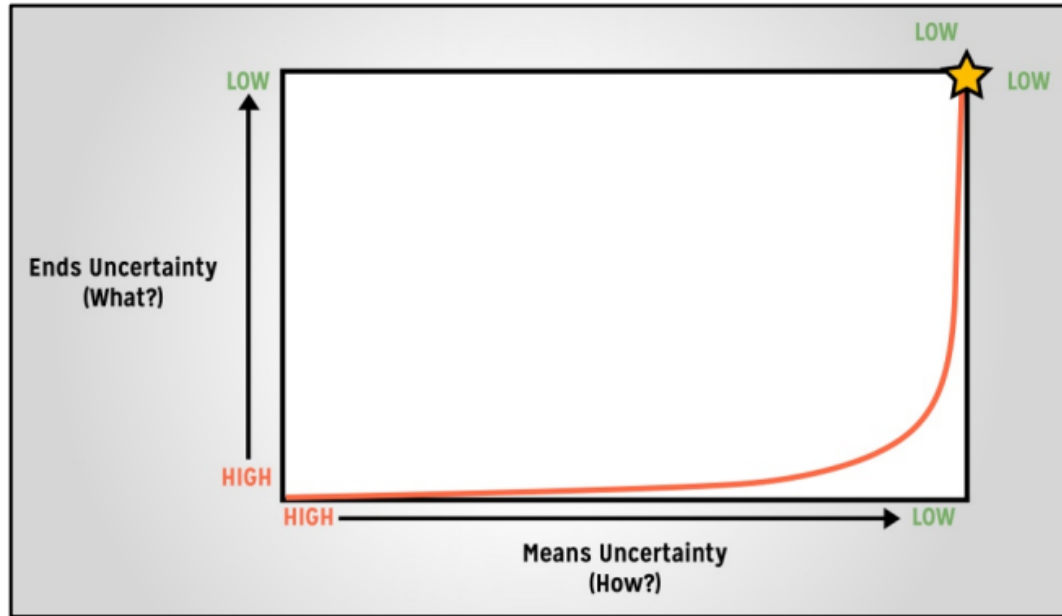


Waterfall Process Example

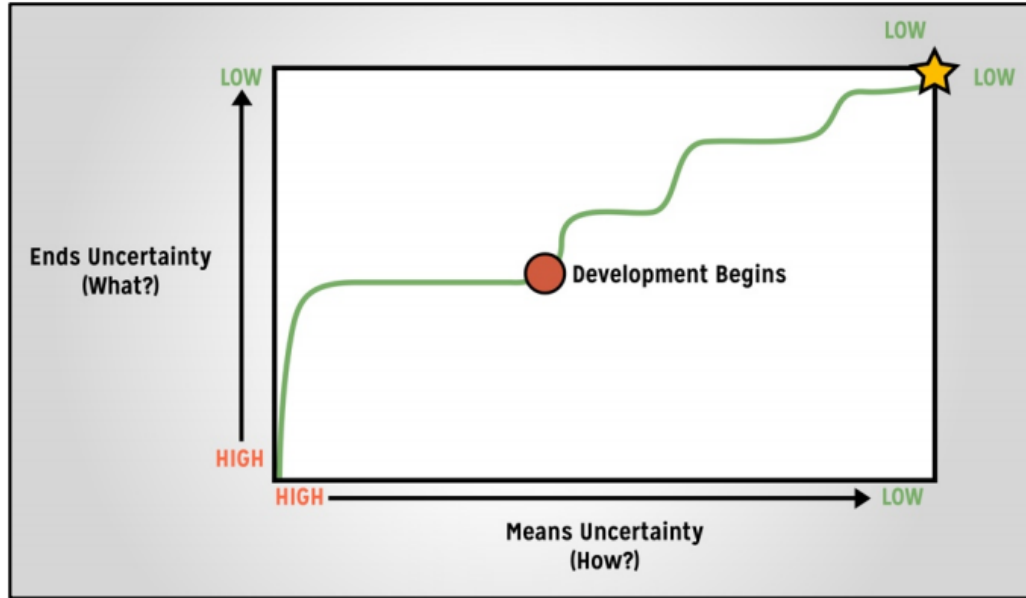


Ad Hoc Development Example

*Ad hoc =
"made up as
you go"*

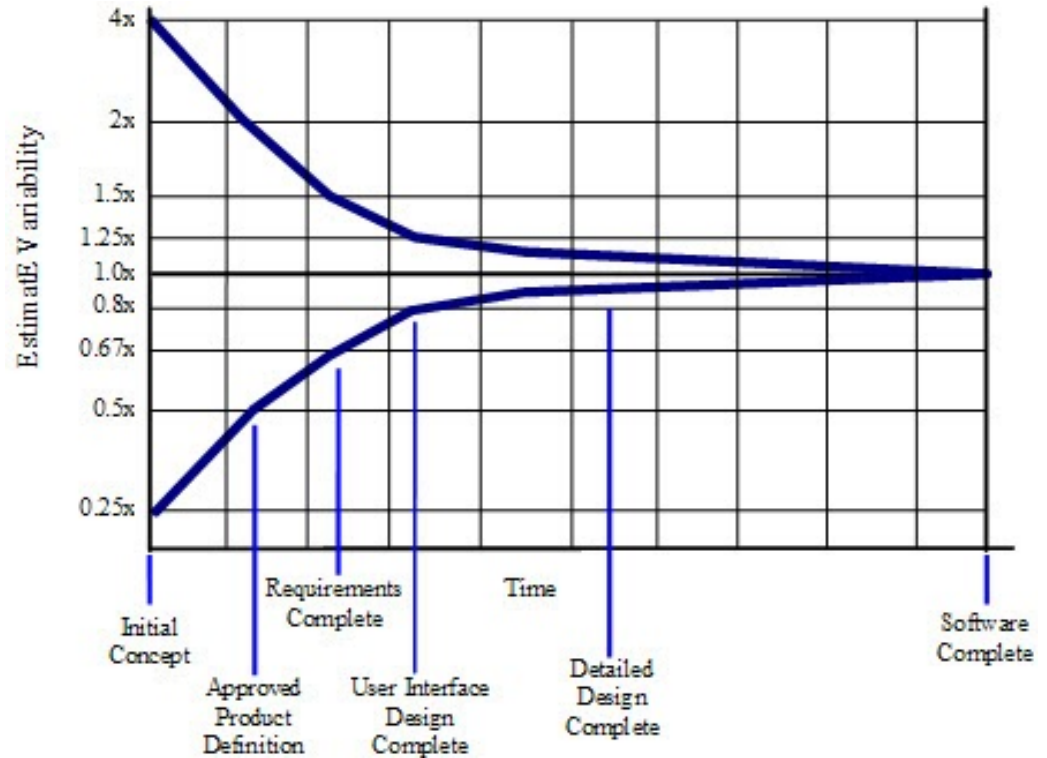


A Real World Case



Cone of Uncertainty

*To underestimate
or to
overestimate?*



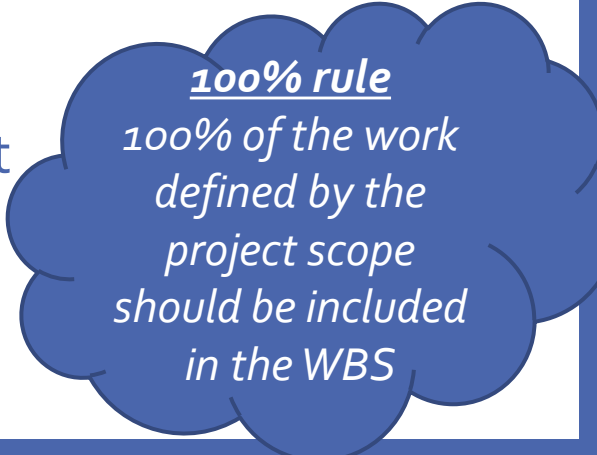
Work Breakdown Structure (WBS)

- **WBS**

- A complete depiction of all tasks necessary to complete a project successfully

- **What is a “task”?**

- Small enough for adequate planning
- Large enough to avoid micromanagement



100% rule
*100% of the work
defined by the
project scope
should be included
in the WBS*

WBS Example

1

• Project initiation

- ...
- ...

2

• Hardware/equipment acquisition

- ...
- Development
- Testing
- Implementation
- Sprint?

1

- define needs
- define deliverables
- define scope
- define measurement methodology
- define process methodology
- define tools and resources
- define risks
- define assumptions and constraints
- define contingencies

2

- **select hardware**
 - define requirements
 - find the vendors
 - request bids
 - look over proposals
 - demo
 - negotiate deal
- **make the purchase**
 - request purchase order for vendor
 - take delivery and get it installed
 - verify that everything is working correctly

Estimate vs Target vs Commitment



Estimate

A guess of time it will take
the team / worker to
complete a task

Usually a range

Not negotiable



Target

An “ideal deadline”

Set externally to the
development team

Not negotiable

Examples: end of a sprint,
release date



Commitment

What you agree to deliver
Negotiable!

Estimate is discussed with
the developed team →
discussed with the client
→ commitment

Example

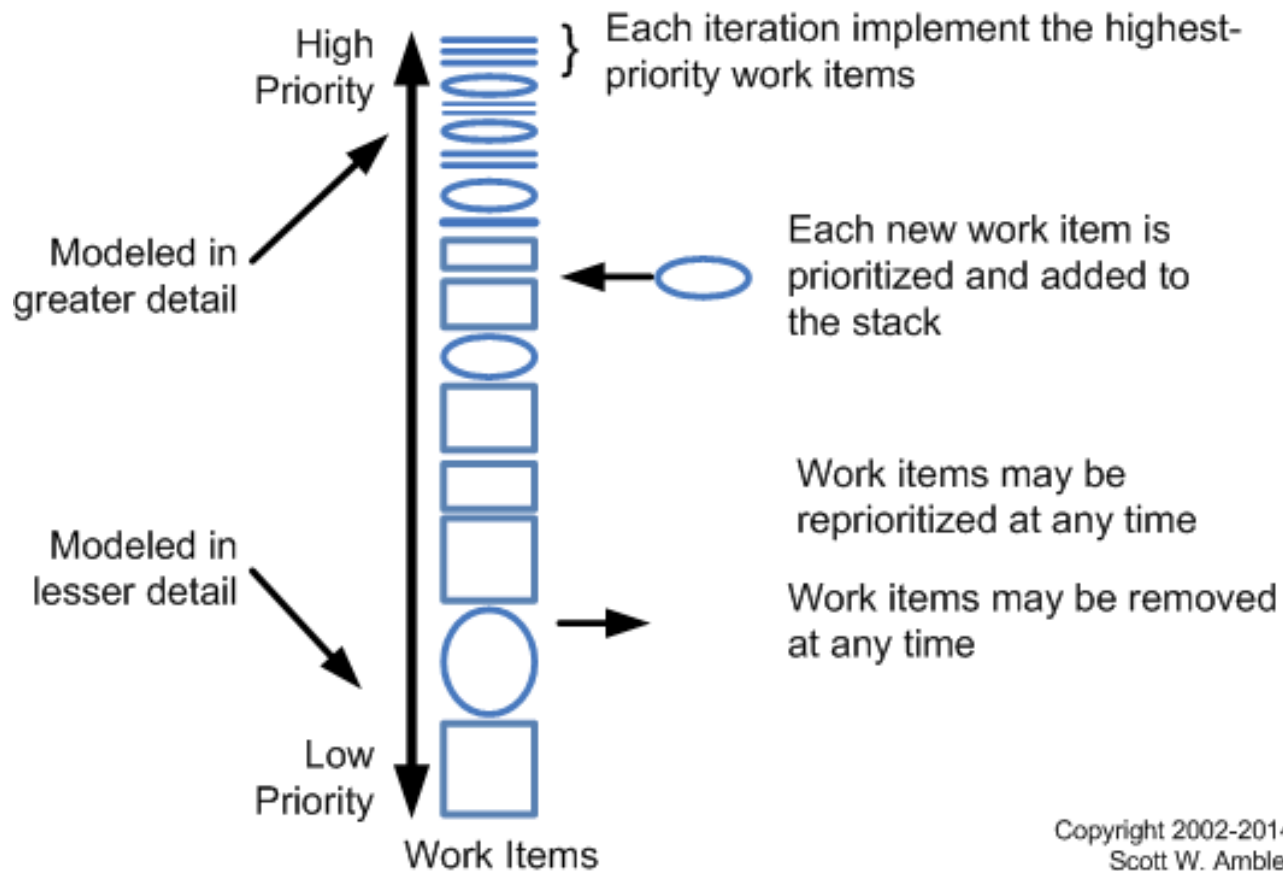
1. Estimate – 1000 hours
2. Client wants it to be done in 600 hours (Target)
3. Negotiations between a client and a development team
4. Decided to adjust the project scope
5. Commitment – 500 hours

BAD PRACTICE:

*Estimate =
Commitment*

Estimation Based on Story Points

- Story points – an alternate system for estimates
- How long it will take to finish a piece of work IN RELATION to other pieces of work?
- **Story points \neq hours!!!**



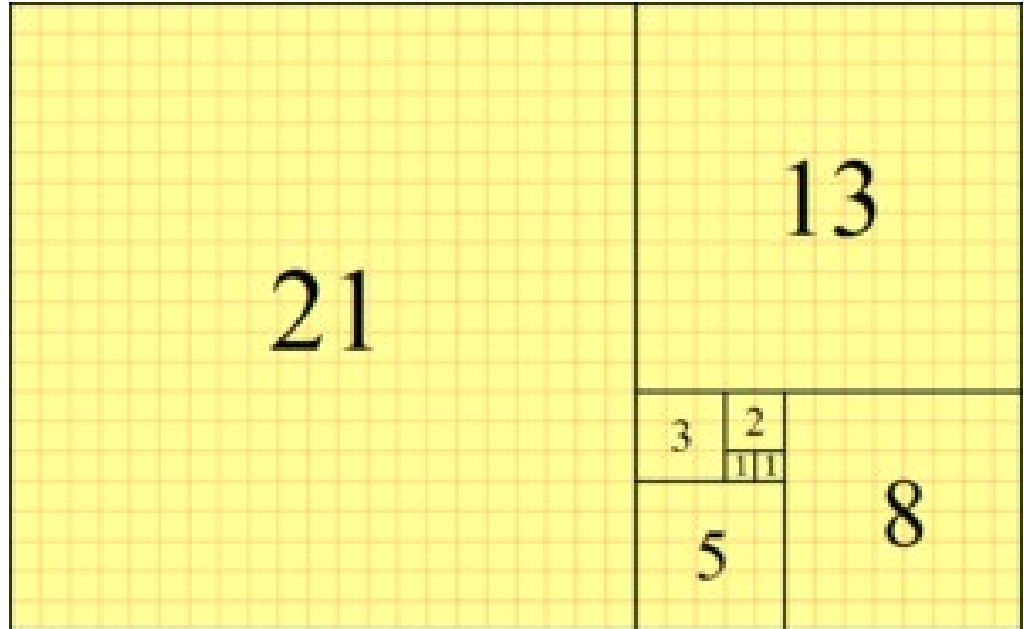
How to Assign Story Points?

Many options out there!

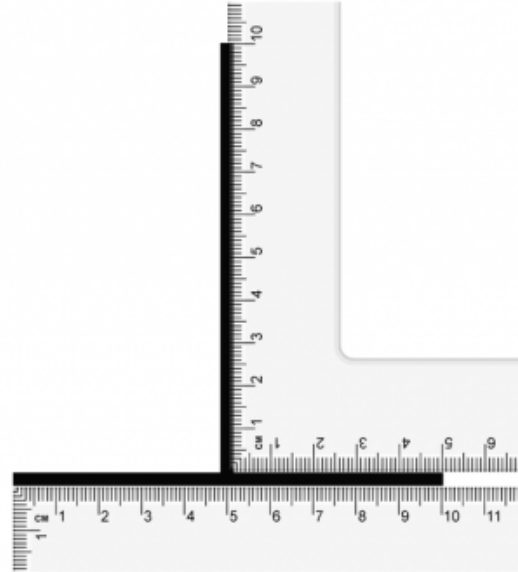
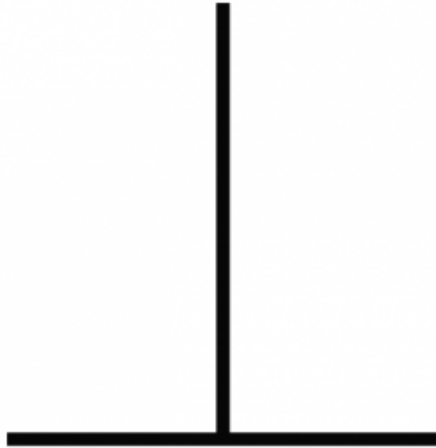
One of them – using the Fibonacci sequence

1. 1. 2. 3. 5. 8. 13. 21. 34. 55. 89, 114, ...

If an estimate falls between – round up!
(e.g., 9 → 13)



BTW, People Are BAD at Estimation...



Velocity

- Number of story points completed over a sprint
 - Measurement of the work that can be taken on and finished over the length of a sprint
 - Partially finished tasks do not count towards velocity!
- **Velocity-driven development**
 - E.g.: if the three previous sprints had a velocity of 15 points per sprint, we can use it to plan the next sprint

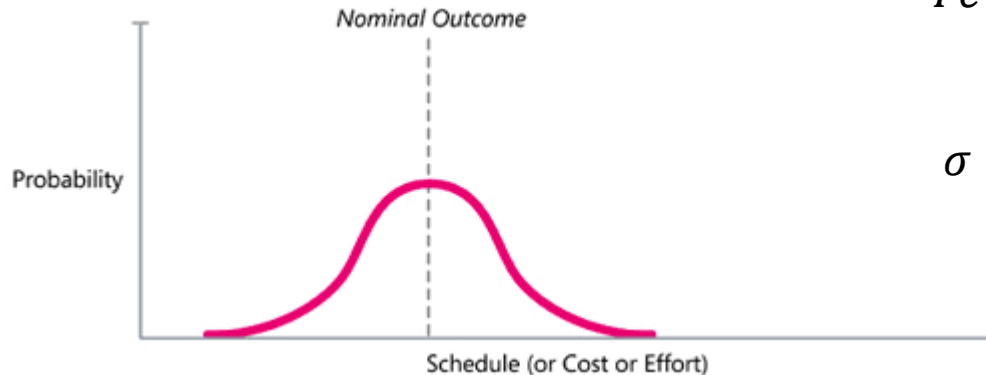
How to Estimate Time?

- Given:
 - **T_m** = most probable time
 - **T_o** = optimistic time
 - **T_p** = pessimistic time

- Calculate
 - **T_e** = expected time
 - **σ** = deviation

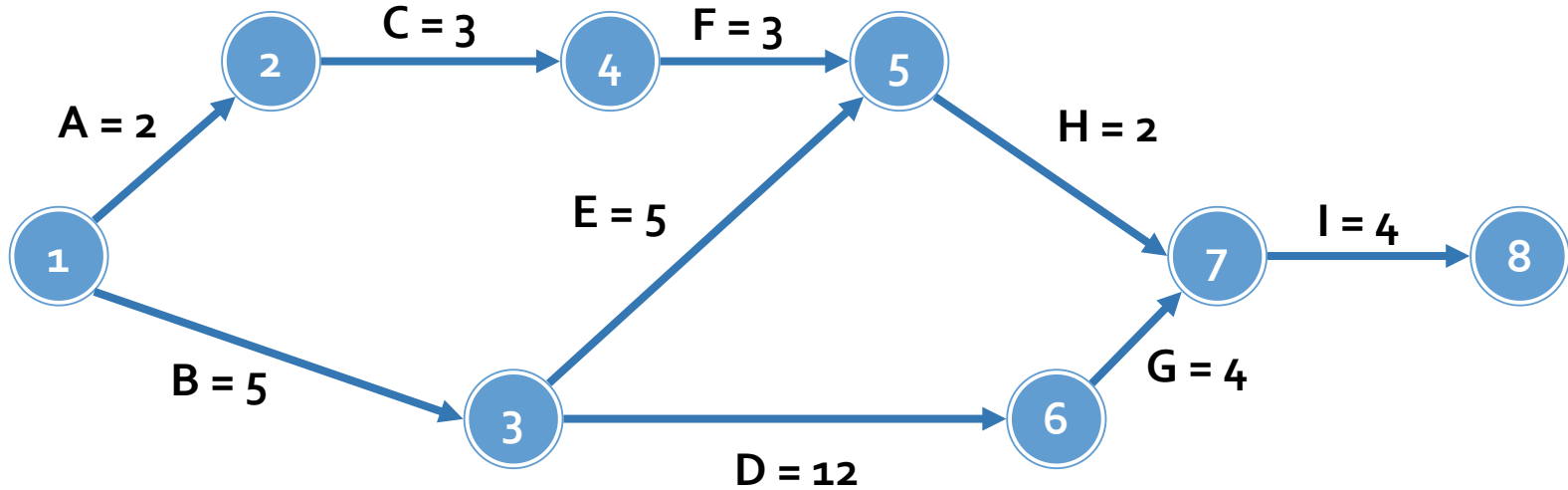
$$T_e = \frac{T_o + 4T_m + T_p}{6}$$

$$\sigma = \frac{T_p - T_o}{6}$$



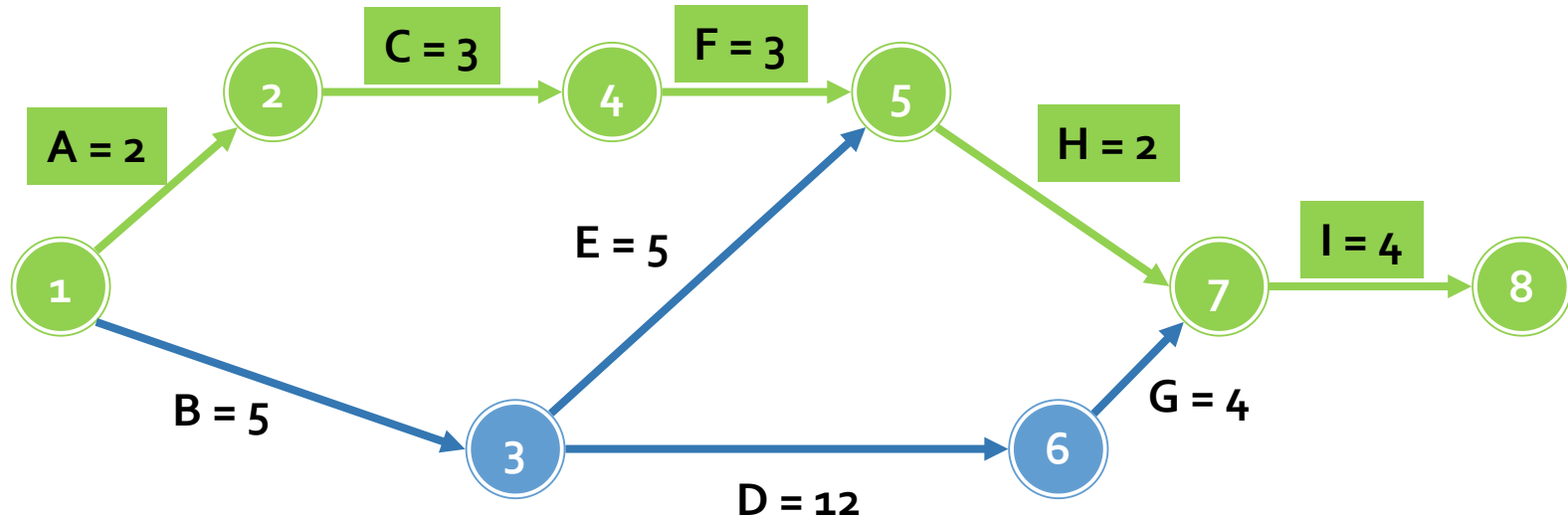
Critical Path

2 Milestones
A = 2 Time estimates



Path 1

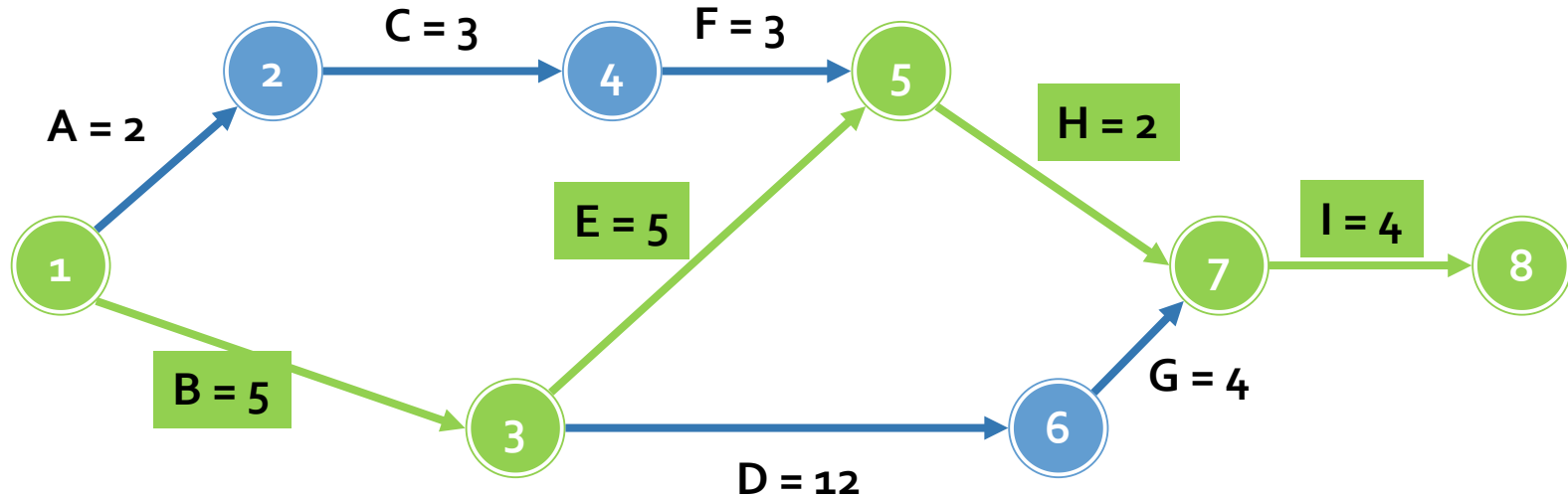
2 Milestones
A = 2 Time estimates



$$A-C-F-H-I = 2+3+3+2+4 = 14 \text{ days}$$

Path 2

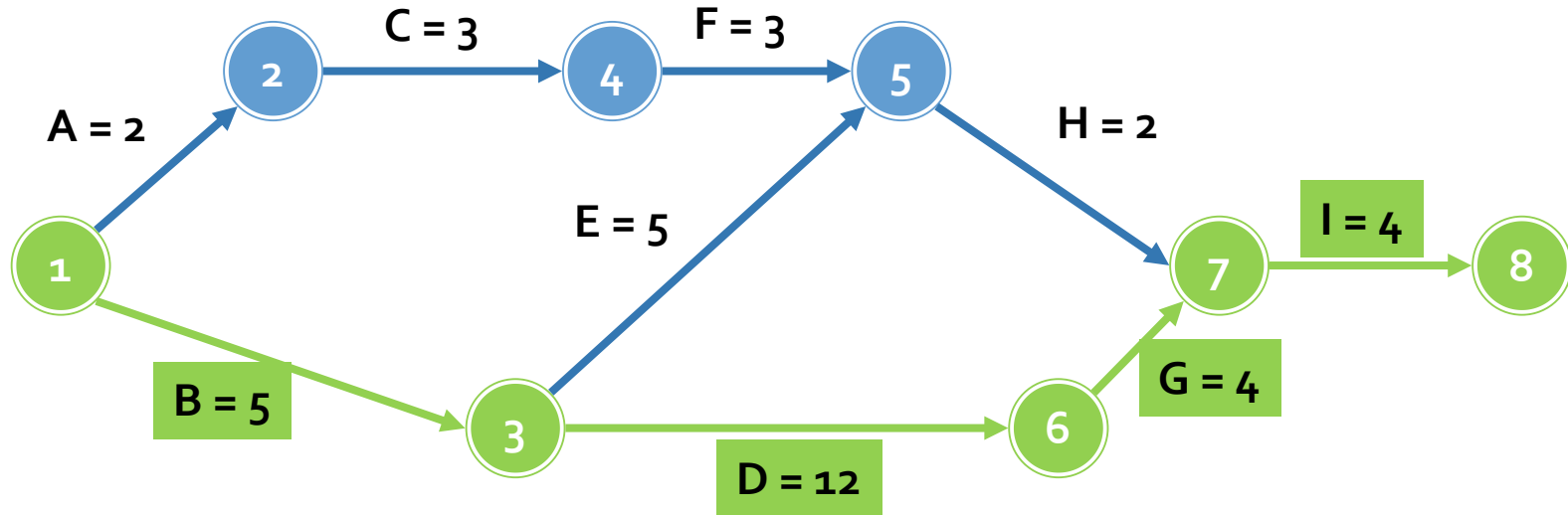
2 Milestones
A = 2 Time estimates



$$B-E-H-I = 5+5+2+4 = 16 \text{ days}$$

2 Milestones
A = 2 Time estimates

Path 3



$$\text{B-D-G-I} = 5 + 12 + 4 + 4 = 25 \text{ days}$$

Critical Path

Path 1: A-C-F-H-I = $2+3+3+2+4 = 14$ days

Path 2: B-E-H-I = $5+5+2+4 = 16$ days

Path 3: B-D-G-I = $5+12+4+4 = 25$ days

Creating an Iteration Plan

1. Create a Sprint Goal
2. Report the project velocity from the last sprint
3. Determine all potential user stories for the sprint, break them down into tasks
4. Developers must create estimates for each task
5. After estimates have been made, revisit chosen user stories
6. The development team must sign up for tasks (*preferably, tasks should be self-assigned*)

Causes of Failures

- Scope risks
- Technology risks
- Customer risks
- Stakeholder risks
- Personnel risks
- Other risks: legal, security, physical...

THINK

How are you going to apply these concepts?

Quiz

- Available on eClass
- Submit until the end of this week (Sunday 11:59 pm)

Individual Assignment

- Cybera + Django
- Oct 2 (Friday), 9:00 – 10:50: Assignment Tutorial
- Oct 9 (Friday), 9:00 – 10:50: Help Session
- Assignment due Oct 16**

Bonus: Anti Patterns

Group Anti Patterns

- Analysis paralysis
- Gold plating
- Cart before the horse
- Viewgraph engineering
- Groupthink
- Fire drill
- Silos
- Heroics
- Vendor lock-in
- Death march
- Over-engineering

Individual Anti Patterns

- Micromanagement
- Seagull management
- Email as the primary means of communication
- Loose cannons
- Intellectual violence