



Premesse al Corso

IS

Anno accademico 2009/10
Ingegneria del Software mod. A

Tullio Vardanega, tullio.vardanega@math.unipd.it

Corso di Laurea in Informatica, Università di Padova1/22



Premesse al Corso

Ingegneria del software – 1

- ❑ **Disciplina per la realizzazione di sistemi *software* così impegnativi da richiedere lavoro di gruppo**
 - Capacità di produrre "in grande" oltre che "in piccolo"
 - Assicurare la qualità dei prodotti (efficacia)
 - Garantire l'efficienza della produzione
- ❑ **Molti sistemi esistono in versioni multiple**
 - Alcune di esse devono operare per molti anni
- ❑ **Esistono svariate tipologie di prodotti *software***
 - Su commessa → forma, contenuto e funzione fissate dal cliente
 - Pacchetto → forma, contenuto e funzione idonee alla replicazione
 - Componente → forma, contenuto e funzione adatte alla composizione
 - Servizio → forma, contenuto e funzione fissate dal problema

Corso di Laurea in Informatica, Università di Padova2/22



Premesse al Corso

Ingegneria del software – 2

- ❑ **Durante il proprio ciclo di vita molti sistemi vengono sottoposti a diverse forme di manutenzione**
 - Correttiva: per correggere difetti eventualmente rilevati
 - Adattativa: per adattare il sistema alla modifica di requisiti
 - Evolutiva: per aggiungere funzionalità al sistema
- ❑ **La manutenibilità è una qualità essenziale**
 - Come ottenerla, come garantirla?
 - *In the large, in the small*

Corso di Laurea in Informatica, Università di Padova3/22



Premesse al Corso

Ingegneria del software – 3

- ❑ **I progetti *software* sono spesso insoddisfacenti**
 - Difficoltà nelle fasi iniziali : ritardi e maggior costo
 - Cambi di piattaforma e tecnologia inattesi e/o non controllabili
 - Difetti residui nel prodotto finale
- ❑ **A volte falliscono clamorosamente**
 - Per obsolescenza prematura
 - Per incapacità o impossibilità di completare
 - Per esaurimento dei finanziamenti

Corso di Laurea in Informatica, Università di Padova4/22



Premesse al Corso

Ingegneria del software – 4

- ❑ **Riuscire a soddisfare obiettivi prefissati entro limiti certi di tempo e di sforzo**
- ❑ **L'applicazione di principi ingegneristici al *software* è compito arduo**
 - Mancano base matematica solida e parametri tecnici certi
 - Disciplina prevalentemente basata su un ciclo virtuoso "esperienza ↔ sistematizzazione"
 - *Best practice*
- ❑ **Conferenza NATO a Garmisch (D)**
 - 7-11 ottobre 1968: nasce la disciplina *Software Engineering* come risposta alla "*software crisis*"

Corso di Laurea in Informatica, Università di Padova5/22



Premesse al Corso

Definizione 1: secondo IEEE

L'approccio sistematico allo sviluppo, all'operatività, alla manutenzione e al ritiro del *software*

(Glossario IEEE)

- ❑ **Il *software* è un prodotto con un proprio ciclo di vita**
 - La sua manutenzione costa spesso molto più della sua produzione
 - I costi di produzione sono spesso dominati dai costi di verifica
- ❑ **Il suo trattamento richiede un approccio sistematico**

Corso di Laurea in Informatica, Università di Padova6/22



Premesse al Corso

Definizione 2: secondo Fairley

La disciplina tecnologica e gestionale per la produzione sistematica e la manutenzione di prodotti *software* sviluppati e modificati con tempi e costi preventivati
(R. Fairley, 1985)

- **Enfasi sulla disciplina gestionale**
 - Controllo di costi, tempi, risorse
 - Controllo di qualità: costi contenuti e risultati definiti

Corso di Laurea in Informatica, Università di Padova

7/22



Premesse al Corso

Figura professionale – 1

- **Software engineer ≠ programmatore**
- **Il programmatore**
 - Figura professionale dominante negli anni pionieristici dell'informatica ('50-'70)
 - Scrive programmi per se stesso, da solo, sotto la propria responsabilità tecnica → svolge un'attività creativa fortemente personalizzata
 - "Bomba logica" contro il (pre-)pensionamento



Corso di Laurea in Informatica, Università di Padova

8/22



Premesse al Corso

Figura professionale – 2

- **Il software engineer**
 - Realizza parte di un sistema complesso che potrà essere usato, completato e modificato da altri
 - Deve guardare e comprendere il quadro generale nel quale il sistema cui contribuisce si colloca
 - La dimensione "sistema" include ma non si limita al *software*
 - Deve operare compromessi intelligenti e lungimiranti tra visioni e spinte contrapposte
 - Costi – qualità
 - Risorse (tempo, mezzi e competenze) – disponibilità
 - ...

Corso di Laurea in Informatica, Università di Padova

9/22



Premesse al Corso

Un principio cardine – 1

- **Frederick P. Brooks, Jr.**
Computer Magazine, aprile 1987
No Silver Bullet. Essence and Accidents of Software Engineering
 - Distingue tra problematiche essenziali ...
 - Specifica, realizzazione, verifica di prodotti *software*
 - ... e problematiche accidentali
 - Gli strumenti e le tecniche per la rappresentazione e la verifica di accuratezza di rappresentazione delle problematiche essenziali

Corso di Laurea in Informatica, Università di Padova

10/22



Premesse al Corso

Un principio cardine – 2

- L'evoluzione tecnica e tecnologica può rendere sempre più agevole affrontare le problematiche accidentali
- Nessuna soluzione tecnica o tecnologica potrà esonerarci dall'impegno concettuale, di astrazione, di analisi, di rigore (ecc.!) necessario per affrontare le problematiche essenziali
 - Non ci può essere alcuna "soluzione finale"

Corso di Laurea in Informatica, Università di Padova

11/22



Premesse al Corso

Un principio cardine – 3

- Brooks ci invita a guardare con scetticismo a ogni slogan commerciale che attribuisca a strumenti tecnici o tecnologie la capacità intrinseca di risolvere le problematiche essenziali dell'informatica!

Corso di Laurea in Informatica, Università di Padova

12/22



Premesse al Corso

Problematiche essenziali – 1

- ❑ **Sviluppo e realizzazione**
 - Strategie di analisi e di progettazione
 - Tecniche per la comprensione e la soluzione di un problema
 - *Top-down, bottom-up*, progettazione modulare, OO
- ❑ **Linguaggi di specifica, progettazione e analisi**
 - Strumenti formali per la definizione di sistemi software
 - UML (informale), Reti di Petri, Z, B, VDM, ... (formali)
 - Ambienti di sviluppo
 - Strumenti per analisi, progettazione e realizzazione
 - Strumenti tradizionali, CASE, RAD

Corso di Laurea in Informatica, Università di Padova

13/22



Premesse al Corso

Problematiche essenziali – 2

- ❑ **Processo *software***
 - **Organizzazione e gestione dei progetti**
 - Metodi di composizione dei gruppi di lavoro
 - Strumenti di pianificazione, analisi, controllo
 - **Cicli di vita del *software***
 - Definizione e correlazione delle attività necessarie allo sviluppo, all'uso, alla manutenzione
 - Modelli ideali di processi (di sviluppo, di uso, di manutenzione, ...)
 - **Modelli del processo di sviluppo**
 - Norme per la definizione delle attività
 - Strumenti per la definizione dei processi

Corso di Laurea in Informatica, Università di Padova

14/22



Premesse al Corso

Problematiche essenziali – 3

- ❑ **Qualità del *software***
 - **Metodi di verifica e controllo**
 - Metodi di verifica, criteri di progettazione delle prove
 - Controllo della qualità, valutazione del processo di sviluppo
 - **Modelli di qualità**
 - Definizione di caratteristiche della qualità
 - Valutazione dei prodotti
 - **Metriche *software***
 - Unità di misura, scale di riferimento, strumenti
 - Indicatori di qualità

Corso di Laurea in Informatica, Università di Padova

15/22



Premesse al Corso

Relazione con altre discipline

- ❑ **L'ingegneria del *software* ha relazioni strette con svariate discipline sia informatiche che non**
 - **Linguaggi di programmazione**
 - **Architettura degli elaboratori**
 - *Hardware*, sistemi operativi, basi di dati, reti, sicurezza
 - **Scienze gestionali**
 - Del tempo, delle risorse, delle persone
 - **Ingegneria**
 - Dei sistemi, elettronica

Corso di Laurea in Informatica, Università di Padova

16/22



Premesse al Corso

Libro di testo – 1

- ❑ **I libri esistenti si dividono in due categorie**
 - **Teorici** : trattano la materia in modo privo di riflessi di **esperienza concreta**
 - Principi esposti, ma troppo spesso non vissuti
 - **Esperienziali** : espongono l'esperienza degli autori, ma spesso senza relazionarla in modo convincente alla **visione astratta del problema e della disciplina**
 - Eccessiva enfasi sugli aspetti accidentali
- ❑ **Non utilizzeremo direttamente né gli uni né gli altri**

Corso di Laurea in Informatica, Università di Padova

17/22



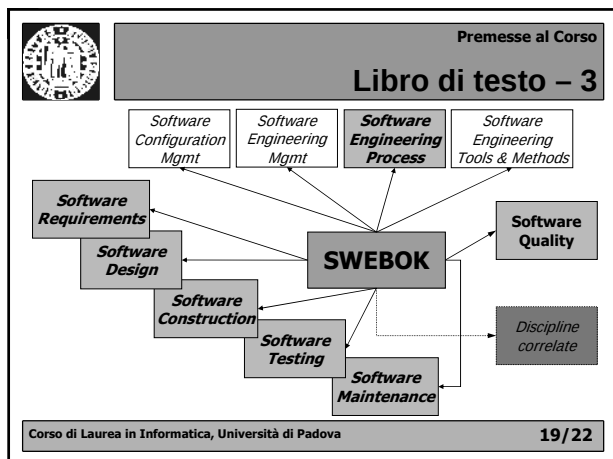
Premesse al Corso

Libro di testo – 2

- ❑ **Faremo riferimento esplicito e implicito a**
 - **Guide to the Software Engineering Body of Knowledge**
IEEE Computer Society
Software Engineering Coordinating Committee
<http://www.swebok.org>
- ❑ **Lo SWEBOK ci aiuterà a familiarizzarci con le 10 aree di conoscenza della disciplina**

Corso di Laurea in Informatica, Università di Padova

18/22



 Premesse al Corso

Libro di testo – 4

- ❑ **Come testo di consultazione useremo**
 - Ian Sommerville
Software Engineering, 8th ed.
Addison Wesley (Pearson Education)
2006
- ❑ **Importante ausilio progettuale**
 - E. Gamma, R. Helm, R. Johnson, J. Vlissides (GoF)
Design Patterns
Addison-Wesley, Pearson Education Italia, ISBN 88-7192-150-X

Corso di Laurea in Informatica, Università di Padova 20/22

 Premesse al Corso

Progetto didattico – 1

- ❑ **Un progetto didattico impegnativo da svolgere in gruppo**
 - Riconduce a unità i 2 moduli del corso IS
 - Aiuta a esplorare in forma esperienziale le 3 dimensioni fondamentali della disciplina IS
- ❑ **6-7 persone per gruppo**
- ❑ **~100 ore di impegno individuale**

Corso di Laurea in Informatica, Università di Padova 21/22

 Premesse al Corso

Progetto didattico – 2

- ❑ **Dimensione 1: attività di gruppo**
 - Ripartita, coordinata, regolata e controllata
- ❑ **Dimensione 2: analisi del problema**
 - Al di là degli aspetti puramente realizzativi
 - Attenzione alla tipologia degli utenti, all'ambito d'uso, alle risorse disponibili, alle evoluzioni future, ...
- ❑ **Dimensione 3: disciplina**
 - Adesione rigorosa a una metodologia di sviluppo e di conduzione del progetto

Corso di Laurea in Informatica, Università di Padova 22/22