

INSEGNAMENTO ANNUALE DI
INGEGNERIA DEL SOFTWARE (9 CFU)
PER ALLIEVI IN INGEGNERIA INFORMATICA
a.a. 2013-2014

Docenti

Marina Zanella Orario ricevimento (S1): giovedì 16.30-18.30

Dip. Ingegneria dell'Informazione

Studio 28, Tel. 030 3715.596

e-mail: marina.zanella@ing.unibs.it

Testi adottati

PRIMO SEMESTRE

M. FOWLER, UML DISTILLED, ADDISON-WESLEY ITALIA, 4/E, 2010

C. GHEZZI, M. JAZAYERI, D. MANDRIOLI, INGEGNERIA DEL SOFTWARE –
FONDAMENTI E PRINCIPI, 2/E, PEARSON EDUCATION ITALIA, 2004

(Copie dei lucidi disponibili al sito <http://www.prenhall.com/ghezzi>)

SECONDO SEMESTRE

E. GAMMA, R. HELM, R. JOHNSON, J. VLISSIDES, DESIGN PATTERNS: ELEMENTI PER IL
RIUSO DI SOFTWARE A OGGETTI, ADDISON – WESLEY, PRIMA EDIZIONE ITALIANA, 2002
(ANCHE SU CD IN VERSIONE INGLESE DEL 1998)

Altre fonti

Copie dei **lucidi** del docente saranno a breve disponibili sul sito dell'insegnamento nel portale di E-learning di Ateneo (Moodle). Attualmente i lucidi delle prime lezioni si possono trovare sul sito

<http://eracle.ing.unibs.it/se>

Testi per la consultazione

L. BARESI, L. LAVAZZA, M. PINCIAMORE, DALL'IDEA AL CODICE CON UML 2, PEARSON EDUCATION ITALIA, 2006

A. BINATO, A. FUGGETTA, L. SFARDINI, INGEGNERIA DEL SOFTWARE – CREATIVITÀ E METODO, PEARSON EDUCATION ITALIA, 2006
(WebBook disponibile al sito <http://webbook.cefriel.it>)

B. BRUEGGE, A. H. DUTOIT, OBJECT-ORIENTED SOFTWARE ENGINEERING USING UML, PATTERNS, AND JAVA, 2/E, PEARSON EDUCATION, 2004

C. S. HORSTMANN, PROGETTAZIONE DEL SOFTWARE E DESIGN PATTERN IN JAVA, APOGEO, 2004

C. LARMAN, APPLICARE UML E I PATTERN, 3/E, PEARSON EDUCATION ITALIA, 2005

L. A. MACIASZEK, B. L. LIONG, PRACTICAL SOFTWARE ENGINEERING: A CASE STUDY APPROACH, ADDISON-WESLEY, 2005

S. L. PFLEEGER, SOFTWARE ENGINEERING – THEORY AND PRACTICE, 2/E, PRENTICE HALL, 2001

R. S. PRESSMAN, PRINCIPI DI INGEGNERIA DEL SOFTWARE, 5/E, MCGRAW-HILL, 2008

E. STILLER, C. LEBLANC, PROJECT-BASED SOFTWARE ENGINEERING: AN OBJECT-ORIENTED APPROACH, ADDISON WESLEY, 2002

I. SOMMERVILLE, INGEGNERIA DEL SOFTWARE, 7/E, PEARSON EDUCATION ITALIA, 2005

I. SOMMERVILLE, SOFTWARE ENGINEERING, 9/E, ADDISON-WESLEY, 2010
(COPYRIGHT 2011)

W. ZUSER, S. BIFFL, T. GRECHENIG, M. KÖHLE, INGEGNERIA DEL SOFTWARE CON UML E UNIFIED PROCESS, MCGRAW-HILL, 2004

ESAME

- 1) Prova **scritta** (con data e ora assegnate per la “Prova scritta” nella *pagina di appelli e prove intermedie del portale di Ateneo* e iscrizione obbligatoria)
Dura 90' e consiste in domande sia a risposta chiusa sia a risposta aperta. È da superare necessariamente con votazione $\geq 18/30$. Lo studente può scegliere di sostenere ripetutamente questa prova, iscrivendosi ogni volta alla stessa nella *pagina di appelli e prove intermedie del portale di Ateneo*: il voto valido è quello dell'ultima prova consegnata. La mancata consegna (anche reiterata) non comporta alcuna conseguenza purché sia stata segnalata al docente

2) Prova **orale** (con data e ora assegnate per la “Prova orale” nella *pagina di appelli e prove intermedie del portale di Ateneo* e iscrizione obbligatoria)

Presentazione e discussione di un **progetto**, proposto dal docente e realizzato in gruppo (massimo tre persone). Tutti gli appartenenti al gruppo devono sostenere la prova orale insieme, previa iscrizione di ciascuno di loro a suddetta prova nella *pagina di appelli e prove intermedie del portale di Ateneo*.

Linguaggio di programmazione: Java

Il materiale relativo al progetto deve essere stato consegnato in precedenza rispetto alla prova orale, entro la data indicata nelle note della *pagina di appelli e prove intermedie del portale di Ateneo*. La consegna deve avvenire depositando il materiale (relazione cartacea + CD contenente copia elettronica della relazione cartacea, codice dell'applicazione software richiesta, copia elettronica di ogni documento ritenuto utile) nella casella di posta ordinaria del docente situata all'ingresso del Dipartimento di Ingegneria dell'Informazione. Ogni altra forma di consegna sarà ritenuta non valida.

Il contenuto della relazione richiesta e le due versioni del codice che devono essere consegnate sono precisati nei documenti di specifica del progetto che saranno illustrati durante il corso.

La prova orale si sostiene una sola volta, a meno di mancato superamento della stessa.

L'ordine temporale relativo con cui uno studente sostiene le due prove è indifferente.

Lo studente che si iscrive alla prova orale deve indicare, mediante una nota della *pagina di appelli e prove intermedie del portale di Ateneo*, i nomi degli altri componenti del gruppo o precisare che ha svolto il progetto da solo.

Ogni richiesta di informazione (tramite messaggio email inviato al docente) circa le modalità d'esame che trovi risposta nelle indicazioni qui pubblicate sarà ignorata.

PIANIFICAZIONE DEL PROGETTO

PRIMA PARTE

Il tema del progetto sarà presentato dal docente a lezione verso la fine del primo semestre. Da quel momento gli studenti possono svolgere il progetto stesso, completando la realizzazione dell'applicazione software richiesta e della documentazione relativa entro l'inizio del secondo semestre.

SECONDA PARTE

Dopo un primo nucleo di lezioni del secondo semestre, gli studenti potranno sottoporre il progetto già realizzato a una serie di attività che saranno chiarite all'inizio di tale periodo didattico e produrre la documentazione relativa alle attività stesse.

N.B. Sia il codice sia la documentazione da consegnare ai fini del superamento della prova orale sono suddivisi nelle due parti sopra indicate.

PROGETTO: PRIMA PARTE

Si veda il file denominato Elaborato2013_14, che indica anche la relativa documentazione richiesta.

Si veda anche il file denominato esempi_elaborato2013_14, che fornisce alcuni esempi circa gli ingressi, le elaborazioni e le uscite dell'applicazione da sviluppare nonché circa gli algoritmi da utilizzare

PROGETTO: SECONDA PARTE

Consiste in testing + manutenzione (correttiva e adattiva, se è il caso) + refactoring + testing di regressione del programma prodotto come prima parte del progetto

Documentazione richiesta:

- una relazione che descriva le attività svolte
- il codice (sorgente ed eseguibile) del sistema considerato, sia nella versione precedente sia nella versione successiva agli interventi effettuati
- l'eventuale codice realizzato per l'attività di testing

CALENDARIO

Le lezioni sono sospese nelle prima settimana di Ottobre a causa di impegni scientifici del docente.