

Programmazione Orientata agli Oggetti

A.A. 2018-2019

Presentazione del Corso

Informazioni Generali

- **Docente:** Valter Crescenzi
crescenzi@dia.uniroma3.it

<http://crescenzi.dia.uniroma3.it>

- **Sito del corso:**
<https://sites.google.com/site/roma3poo>

- Iscrivetevi (subito!) al gruppo di discussione del corso per ricevere gli avvisi

<https://sites.google.com/site/roma3poo/avvisi-e-discussioni>

Il Corso

- CFU: **9** (a meno di importanti debiti formativi)
- Obiettivi
 - Il corso mira ad esporre principi, tecniche e metodologie per la programmazione orientata agli oggetti
- Prerequisiti Formali (nei prossimi anni accademici)
 - **Fondamenti di Informatica**. In particolare è richiesto che lo studente:
 - abbia confidenza con i costrutti della programmazione procedurale (variabili, strutture di controllo, sottoprogrammi, paramentri, array) e con i concetti di base della programmazione ad oggetti (classi e oggetti)
 - sappia scrivere algoritmi non banali e codificarli in Java
 - sappia compilare, eseguire e valutare la correttezza di programmi Java
- Prerequisiti Sostanziali
 - **Algoritmi e Strutture Dati**. In particolare è necessario che lo studente:
 - sappia progettare senza eccessiva difficoltà algoritmi polinomiali di media complessità
 - abbia una visione pertinente di come siano rappresentate le principali strutture dati (in particolare collezioni) utilizzate
- Superato il corso, lo studente sarà in grado di:
 - apprezzare e valutare la qualità del codice
 - realizzare applicazioni OO di complessità non banale
 - partecipare alla realizzazione di applicazioni OO di media complessità
 - comprendere alcune delle problematiche della programmazione concorrente

Carico di Lavoro

- 9 CFU -> 9×25 h -> 225 h di studio
 - ~80h di lezioni frontali in ~14 settimane
 - ~5 h verifiche (vedi dopo >>)
 - ~Almeno $225 - 85 = 140$ ore di lavoro individuale, ovvero ~10h a settimana, ovvero ~2h al giorno (festivi esclusi) **oltre** la lezione
 - Ma usate i festivi per recuperare quanto non avete fatto negli altri giorni...
- ✓ Il corso è stato progettato, e raffinato per molti anni consecutivi, **anche** con lo scopo di allinearsi al corretto carico di lavoro

Strumenti

- Linguaggio di programmazione: Java
 - Implementa in maniera chiara e lineare i principi della POO
 - E' molto popolare e diffuso (anche e soprattutto nell'industria)
- Strumenti di sviluppo
 - Eclipse, un IDE professionale, molto usato anche in ambito industriale

<https://www.eclipse.org/downloads/>

Contenuti

- [Introduzione al linguaggio Java](#)
 - Il paradigma di Programmazione Orientato agli Oggetti
 - Classi e oggetti
 - Gestione della memoria
 - Riferimenti e tipi primitivi, costruttori, metodi, parametri, information hiding, overloading
 - Qualità del codice e Testing
 - Documentazione
 - Accoppiamento e coesione
 - Polimorfismo ed Estensione
 - Interfacce
 - Supertipi e sottotipi, polimorfismo, upcasting, downcasting
 - Estensione: concetti base
 - Collezioni e Generics
 - Generics, List, Set, Map, utilità
 - Riutilizzo del codice
 - Estensione: approfondimenti
 - Classi astratte
 - [Tipi enumerati](#)
 - [Classi nidificate](#)
 - I/O Stream, Gestione delle eccezioni, riflessione, annotazioni
 - [Introduzione alla Programmazione Concorrente](#)
 - [Java Thread: creazione, terminazione, sincronizzazione](#)
 - [Interferenza](#)
 - [Speed-up per semplici Problemi di Decomposizione Parallela](#)
 - [Confinamento per Thread](#)
 - [Introduzione alla Programmazione ad Eventi](#)
 - [Modello di Concorrenza sottostante GUI moderne](#)
 - [JavaFX](#)
 - [Librerie](#)
-

Impostazione Didattica

- Programmazione Orientata agli Oggetti
 - programmazione "in the large"
 - manutenzione e riuso
 - Questi aspetti si comprendono solo con una esperienza su un progetto di medie dimensioni in continua evoluzione
 - Si propone pertanto uno studio di caso, che evolve durante tutto il corso
-

Lo Studio di Caso: un Gioco di Ruolo

Classi e Oggetti

DiaDia
Stanza
Comando
Attrezzo
Partita

Qualità del codice

DiaDia
Stanza
Comando
Attrezzo
Partita
Labirinto

Polimorfismo

DiaDia
Stanza
Partita
Comando
ComandoVai
ComandoGuarda
Giocatore

Collezioni

DiaDia
Stanza
Comando
ComandoVai
ComandoGuarda
Giocatore
Attrezzo
ComandoRaccogli
ComandoPosa
ComandoPosa

Riuso del codice Ereditarietà e composizione

DiaDia
Stanza
StanzaMagica
Comando
ComandoVai
ComandoGuarda
Giocatore
Attrezzo
ComandoRaccogli
ComandoPosa
Personaggio*
Mago
Matto

Stream, Eccezioni

DiaDia
Stanza
StanzaMagica
Comando
ComandoVai
ComandoGuarda
Giocatore
Attrezzo
ComandoRaccogli
ComandoPosa
Personaggio*
Mago
Matto
LettoreLabirinto
LabirintoNonValidoException

GUI

DiaDia
Stanza
StanzaMagica
Comando
ComandoVai
ComandoGuarda
Giocatore
Attrezzo
ComandoRaccogli
ComandoPosa
Personaggio*
Mago
Matto
LettoreLabirinto
LabirintoNonValidoException
SimpleGUI
ConsoleUI

...
???

Materiale Didattico

Lucidi, codice, esercizi codice a cura del docente
(disponibili sul sito web del corso)

Libri: Un testo da usare come **manuale** Java di riferimento può risultare utile (anche se non indispensabile). Ad esempio:

- Ken Arnold, James Gosling, David Holmes
"Il linguaggio Java: Manuale Ufficiale (IV Edizione)" - Addison Wesley
E' il manuale "ufficiale" del linguaggio.
- Cay Horstmann
"Concetti di informatica e fondamenti di Java" - APOGEO
Un testo con una forte caratterizzazione didattica
- Cay Horstmann, Gary Cornell
"Core Java2 Vol I: Fondamenti (VII ed. Italiana)" - Prentice Hall
"Core Java2 Vol II: Tecniche avanzate (VII ed. Italiana)" - Prentice Hall
Testi molto tecnici e approfonditi (coprono anche molti concetti non affrontati nel corso; gli argomenti del corso sono distribuiti su entrambi i volumi)

Ma in rete c'e' moltissima documentazione e la stessa documentazione di JDK, accompagnata da una buona dose di *perseveranza*, può essere sufficiente

Preparazione all'Esame

- **Preparazione**

- Durante il corso saranno proposti esercizi da svolgere autonomamente, utili per verificare la preparazione ai fini dell'esame
 - Due tipologie di esercizi:
 - **Verifiche:** micro-programmi Java (max 3 classi) mirano a valutare la comprensione di aspetti tecnici puntuali
 - **Progetti:** ristrutturazioni di piccoli progetti mirano a sviluppare capacità metodologiche
 - Per alcuni esercizi viene proposta una soluzione in aula
 - Talvolta il testo d'esame è basato su progetti già presentati a lezione
 - Per completare la preparazione è utile svolgere tutti i compiti d'esame degli ultimi anni **scrivendo non solo la soluzione, ma anche il codice per verificarne la correttezza**
 - Attenzione: una preparazione sovraspecializzata solo sui testi di esame non è efficace!
-

Esame

- L'esame si compone di due prove, e per superarlo è necessario prendere la sufficienza in entrambe le prove (quindi chi non supera la prima prova non è ammesso alla seconda)
 - Prima Prova (peso: 12/31; durata circa 45 minuti):
 - ~10 quiz a risposta chiusa
 - Mirano a verificare la conoscenza di aspetti *puntuali* della programmazione orientata agli oggetti
 - Valutazione:
 - risposta esatta: *punteggio positivo (+2)*
 - risposta errata: *punteggio negativo (-1)*
 - risposta nulla: 0
 - ~5 Esercizi di programmazione corretti automaticamente
 - Mirano a verificare la capacità di utilizzare librerie standard essenziali
 - Valutazione:
 - risposta esatta: *punteggio positivo (+3)*
 - risposta errata o nulla: 0
 - Sufficienza: poco meno di 6/10 del punteggio max
 - Seconda Prova (peso: 19/31; durata circa 70 minuti):
 - Esercizio di programmazione **al calcolatore**
 - Di solito, si tratta di ristrutturare una piccola applicazione, al fine di migliorarla applicando correttamente i principi della POO
 - Le linee secondo le quali la applicazione deve essere ristrutturata sono indicate nel testo
 - L'obiettivo è quello di verificare se lo studente è in grado di comprendere e applicare i principi della POO

Esame

- I testi degli esercizi della Seconda Prova d'Esame degli appelli passati sono disponibili sul sito Web del corso
- **I quiz e gli esercizi proposti nella Prima Prova d'Esame sono disponibili sul sistema di quiz raggiungibile dal sito Web del corso (cambiano solo i nomi delle variabili) e sono sempre gli stessi da diversi anni!**

http://dotto.dia.uniroma3.it:8080/quiz/student_start.do



Ripeto ...

I quiz e gli esercizi proposti nella prima prova d'esame sono disponibili sul sistema di quiz raggiungibile dal sito Web del corso (cambiano solo i nomi delle variabili) e sono

SEMPRE GLI STESSI DA DIVERSI ANNI!!!

Filosofia del Corso (1)

- Un ingegnere edile progetta “edifici”
 - Un ingegnere informatico ?
-

Filosofia del Corso (2)

*"per me non è importante che il codice "funzioni",
perché io do per scontato che lo faccia.*

*Le cose che mi interessano sono altre, in
primissimo luogo la manutenibilità del codice
a distanza di anni e da parte di persone diverse"*

[un imprenditore IT]

Filosofia del Corso (3)

- Ingegnere edile: “Guarda la casa che ho progettato. Sta in piedi !”
 - Ingegnere informatico: “Guarda il programma che ho scritto. Funziona !”
 - ...
-

Sindrome del Calcolatore

- Per comprendere la programmazione OO è indispensabile fare esperienza **al calcolatore**
 - Questo inizialmente comporta sicuramente un investimento di tempo, ma è l'unico modo per ottenere risultati
 - Può essere abbastanza semplice comprendere i singoli dettagli tecnici; la difficoltà consiste nel mettere insieme i concetti e nell'applicarli su vasta scala
 - Solo svolgendo esercizi **al calcolatore** si può avere una verifica obiettiva della propria comprensione degli argomenti svolti
 - Solo attraverso l'esercizio **al calcolatore** si può raggiungere un sufficiente livello di approfondimento
 - **L'esame è al calcolatore...**
-

Strumenti

- JDK (Standard Edition)
 - Scaricare ed installare dal sito della ORACLE l'ultima versione disponibile di Java 8
<https://java.com/it/download/>
 - Recommended Version 8 Update **201**
 - Documentazione JDK
 - Scaricare ed installare dal sito della ORACLE
 - Strumenti di sviluppo
 - Un IDE professionale: Eclipse (4.X)
 - Scaricare **Eclipse IDE for Java Developers** da
<https://www.eclipse.org/downloads/eclipse-packages/>
(è la versione usata anche in sede d'esame)
 - Configurare Eclipse per compilare **Java 7** >>
-

Versione di Java di Riferimento

- Il corso farà sistematicamente riferimento a **Java 7**
 - anche se non è l'ultima disponibile
 - Anzi... oramai è disponibile Java 11.0.2!
 - marzo 2019: Java SE Development Kit 8u201
 - Java 8: una notevole evoluzione rispetto a Java 7
 - che introduce novità ispirate alla *Programmazione Funzionale*
 - ✓ E' una scelta didattica
 - ✓ Dal punto di vista didattico, per un **primo** corso sulla Programmazione ad Oggetti è tuttavia fortemente consigliabile limitarci a Java 7
 - ✓ Semplice: Meno scelte possibili → meno errori
 - Tornate su Java 8+ successivamente
 - ✓ ad es. dopo avere frequentato un corso sulla *Programmazione Funzionale*>>
-

Strumenti

- Diamo per scontato che lo studente sappia:
 - Usare il Sistema Operativo (variabili di ambiente incluse)
 - Documentarsi
 - Perseverare
 - Perseverare e documentarsi
 - Spendere le proprie energie per documentarsi!
 - Chiedere!
 - E che nel giro di un paio di giorni:
 - abbia installato Eclipse
 - sappia compilare ed eseguire programmi Java

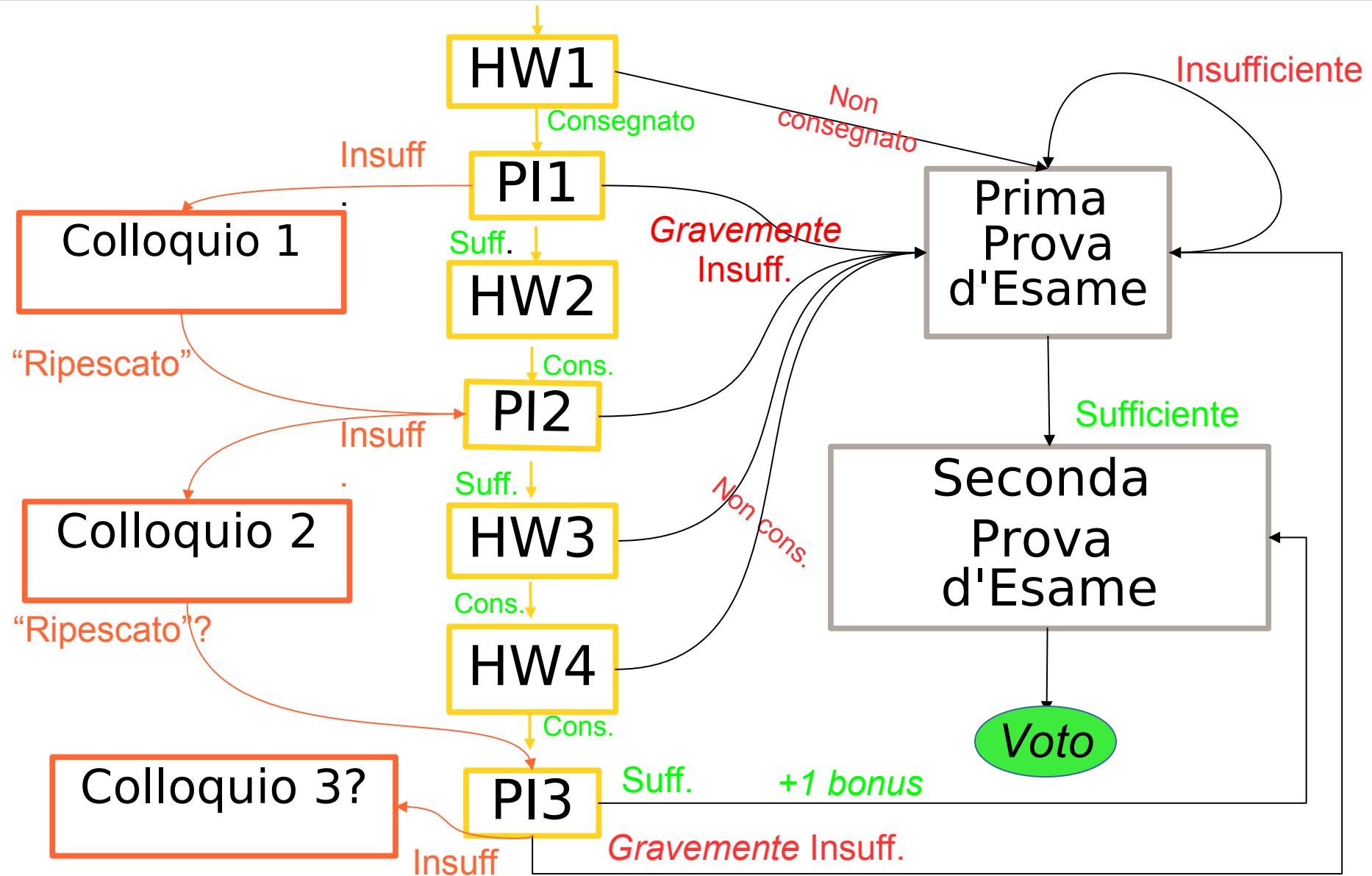
Percorso HQ (Homework & Quiz)

- A chi frequenta questo percorso, durante il corso saranno proposti **homework** e **prove in itinere** di verifica
 - *Homework*: realizzazione di piccoli progetti svolti in **gruppi di due persone** e consegnati singolarmente nei tempi e nelle modalità indicati dal docente
 - *Prove in Itinere*: quiz svolti con le stesse modalità della Prima Prova d'Esame (singolarmente, senza materiale, nel laboratorio CampusOne, in un tempo fissato, usando il **SISTEMA DI QUIZ**) secondo un calendario stabilito dal docente (e comunicato per tempo agli studenti).
 - Non possono essere gestite e/o recuperate assenze
- Gli studenti che ottengono la sufficienza nelle tre prove in itinere sono esonerati dalla Prima Prova d'Esame e il punteggio assegnato sarà pari al punteggio medio ottenuto complessivamente con le Prove in Itinere **+1 bonus**
 - Punteggio di esonero alla Prima Prova al completamento del Percorso HQ:
 $(\text{punteggio totale} / \text{punteggio max ottenibile}) \times 12 + 1$

Percorso HQ (Colloqui)

- Gli studenti che risultati *gravemente* insufficienti ad una prova in itinere non possono continuare il Percorso HQ
 - Sosterranno anche la Prima Prova d'Esame
 - Gli altri studenti che non hanno conseguito voto sufficiente possono continuare il Percorso HQ solo se hanno consegnato i precedenti homework e si presentano per un colloquio con il Tutor Didattico
 - I colloqui *NON* sono un momento di valutazione, ma di supporto allo studente (che ha provato a svolgere gli hw)
-

Percorso HQ (Sintesi)



Risorse Limitate...

- ✓ Si cerca di spendere le poche e limitate risorse per gli studenti che non abbiano mai usufruito delle stesse:
 - i «ripescaggi» saranno sempre di meno mano a mano che avanziamo con le Prove in Itinere
 - ✓ sino a scomparire del tutto (PI3)
 - uno stesso studente può essere ripescato una sola volta
 - non possono accedere al Percorso HQ gli studenti che abbiano già consegnato un HW negli anni precedenti
 - ✓ Sarà fatto un controllo automatizzato sulla base delle matricole che hanno partecipato a PI1 negli ultimi **9** anni
 - Chi consegna quest'anno non potrà partecipare al percorso dei prossimi
 - Semplice e lineare *consiglio*: *Se cominciate...finite!*
 - ✓ Altrimenti meglio non cominciare proprio
-

Quiz “Magico”

- *In realtà la Prima Prova consisteva non di 9 quiz ma di 10!*
 - Il primo dei dieci quiz verrà impartito direttamente oggi, subito!
 - E' un quiz “magico”:
 - Chi indovina la risposta **triplica** la probabilità di superare questo esame al primo appello
 - Chi sbaglia la risposta **triplica** la probabilità di *non* superare questo esame al primo appello
-


Quiz “Magico”

Quando studiare?

- Risposte possibili:

- 1) Da subito e per tutta la durata delle lezioni
 - 2) Appena finiscono le lezioni, prima dell'appello
 - 3) Nessuna delle precedenti
-

Suggerimento...

- 
- 1/3 – Inizio Lezioni
 - HW1
 - ~12/4 – PI1
 - 9/4 – Colloquio su HW1
 - HW2
 - ~17/5 – PI2
 - 8/5 – Colloquio su HW2
 - HW3
 - HW4
 - ~14/6 – PI3
 - 14/6 – Fine delle Lezioni
 - 17/6 – Primo giorno utile dopo la fine delle lezioni
 - 17/6 ✓ Appello d'Esame
-

Statistiche Precedente A.A. (1)

Totale “Avvistati” = **315**;

Totale “Apparsi” = **239**;

✓ Def. di “Avvistato”: studente che da una qualche forma di interazione, anche solo elettronica ma tracciabile, si intuisce esistere;

✓ Def. di “Apparo” : *studente che si presenta ad una qualsiasi prova*;

Percorso HQ

	PI1	HW1	Coll1	PI2	HW2	Coll2	HW3	Coll3	HW4	PI3
totale	162	152	18	131	132	11	122	4	123	117

Statistiche Precedente A.A. (2)

Totale "Avvistati" = 315;

Totale "Apparsi" = 239;

I Appello di Recupero (Settembre)

Pres.			
69	Primo Appello di Recupero		
	Superano Prima Prova	55	Superano Esame
	di quelli che hanno superato la PP		21
			38.18%
	degli "apparsi" durante a.a.		8.79%
	rifiutano:		0

Appello (Giugno)

Pres.				
128	con PI		Primo Appello	
99	Superano Prima Prova	171	Superano Esame	116
	senza aver superato PI3		38	32.76%
	AVENDO superato PI3		78	67.24%
	degli "apparsi" durante a.a.			48.54%
	rifiutano:			6

II Appello di Recupero (Febbraio)

Pres.				
4	Secondo Appello di Recupero			
	Superano Prima Prova	27	Superano Esame	14
	che hanno superato la PP			51.85%
	degli "apparsi" durante a.a.			5.86%
	rifiutano:			0

- * Dei 6 che hanno rifiutato il voto a giugno per ripresentarsi subito a settembre
- 1 ha migliorato il voto
 - 4 lo hanno confermato
 - 1 ha peggiorato il voto

Statistiche Precedente A.A. (3)

Qualche indicatore per stimolare il ragionamento sull'uso efficiente delle risorse comuni

- Se cancellassimo gli appelli di recupero (settembre e febbraio), avremmo comunque il 75% delle verbalizzazioni a fine anno accademico
 - ✓ $[116 - ((21-6) + 14)] / 116 = 75\%$
- La Prima Prova sarebbe svolta solo dal 49% degli studenti a cui viene attualmente erogata (Appello di giugno senza HQ)
 - ✓ $99 / (69 + 34 + 99) \approx 49\%$;
- In definitiva il 25% delle verbalizzazioni costa circa la metà del lavoro svolto per gli esami durante l'intero a.a.
- ✓ E se dedicassimo una parte di queste risorse a quelli che invece cercano di ottenere il massimo da questo corso, da subito???

Domande

Da parte vostra?

Da parte mia:

- ✓ Cortesemente compilate (in forma anonima) il questionario *informale* che trovate a questo indirizzo:

<https://goo.gl/pRXvJR>

Informazioni di Servizio (1)

Se siete immatricolati nell'anno accademico 2015/2016 o precedenti il vostro PdS già prevede

- Calcolatori Elettronici da 9 CFU
- POO da 6 CFU

Avete la facoltà di chiedere un *PdS personalizzato* con

- *Calcolatori Elettronici da 6 CFU*
- POO da 9 CFU

presentandovi di persona (NON E' POSSIBILE FARLO ONLINE) presso la segreteria didattica del Collegio Didattico di Ingegneria Informatica nei giorni:

Lun 9:30-12:30

Mar 9:30-12:30

Gio 13:00-15:00

Fonte: <http://informatica.ing.uniroma3.it/contatti/>

N.B. contestualmente vi verrà chiesto di scegliere i corsi del III anno, cominciate a farvi un'idea

✓ Il mio *endorsement* è sicuramente per il corso di *Programmazione Funzionale*, per i motivi che saranno chiari tra breve...

Informazioni di Servizio (2)

Per chi deve ancora sostenere l'esame da 6CFU, *nessun* particolare trattamento:

- ✓ se non ha mai seguito il Percorso HQ è libero di farlo quest'anno (o nei successivi)
- ✓ se lo ha già seguito, non può seguirlo

Come per tutti gli altri:

- Prima Prova da 12 pt
- Seconda Prova da 19 pt

- ✓ Nella *Seconda Prova d'Esame* troverà una o più domande finali a cui non deve rispondere
-