

Logica

0: Logica per l'Informatica

Claudio Sacerdoti Coen

`<sacerdot@cs.unibo.it>`

Università di Bologna

29/09/2017

Outline

- 1 Il docente
- 2 Cosa studia la logica? (cenni)
- 3 Perchè la studiamo a informatica?
- 4 Organizzazione del corso e modalità di esame
- 5 Decalogo dello studente furbo

Recapiti

- Prof. Claudio Sacerdoti Coen
via Malaguti 1 (scala D)
051 2094973
claudio.sacerdoticoen@unibo.it
<http://www.cs.unibo.it/~sacerdot>
- Pagina Web del corso:
<http://www.cs.unibo.it/~sacerdot/logica>
- Orario di ricevimento (**USATELO!!!**):
??? ??:??



Cosa facciamo?

Docente di

- **Logica per l'Informatica**
(triennale informatica, mutuato filosofia)
- Fondamenti Logici dell'Informatica
(magistrale informatica)
- Paradigmi emergenti di programmazione
(magistrale informatica)
- Algorithms and Data Structures for Computational Biology
(magistrale bioinformatica)

Cosa facciamo?

Attività di ricerca:

- Teoria dei tipi e **dimostrazione assistita**
(software per verificare la correttezza di dimostrazioni, per esempio di correttezza del codice di programmi)
- Linguaggi di programmazione logici, di ordine superiori e a vincoli
- Mathematical Knowledge Management
(software per la gestione e sfruttamento di **conoscenze matematiche**)

Ma lei programma?

Paradigmi di programmazione preferito:

- **funzionale** (OCaml, Haskell, Lisp, ...)
- **logico** (λ Prolog, ...)

Linguaggio di programmazione preferito:

- OCaml (\approx 100,000 righe di codice)
- λ Prolog (\approx 5,000 righe di codice)

Sistema operativo preferito:

- **Linux**, what else?

Outline

- 1 Il docente
- 2 Cosa studia la logica? (cenni)**
- 3 Perchè la studiamo a informatica?
- 4 Organizzazione del corso e modalità di esame
- 5 Decalogo dello studente furbo

Cos'è la logica?

Una disciplina molto ampia con contributi e applicazioni

- 1 In filosofia
- 2 In matematica
- 3 In informatica

Alcune domande chiave:

- **cos'è** la correttezza di un ragionamento?
- **quali** ragionamenti sono corretti?
- ci sono fatti **non deducibili** tramite un ragionamento?

L'informatica come studio di ciò che si può calcolare è figlia della logica (\approx 1930).

*“Il computer sta all'informatica
come il microscopio sta alla biologia.”*

La seguente dimostrazione è corretta?

Definition 1. Let H be a subgroup of a group G . A *left coset* of H in G is a subset of G that is of the form xH , where $x \in G$ and $xH = \{xh : h \in H\}$. Similarly a *right coset* of H in G is a subset of G that is of the form Hx , where $Hx = \{hx : h \in H\}$

Note that a subgroup H of a group G is itself a left coset of H in G .

Lemma 1. Let H be a subgroup of a group G , and let x and y be elements of G . Suppose that $xH \cap yH$ is non-empty. Then $xH = yH$.

Proof. Let z be some element of $xH \cap yH$. Then $z = xa$ for some $a \in H$, and $z = yb$ for some $b \in H$. If h is any element of H then $ah \in H$ and $a^{-1}h \in H$, since H is a subgroup of G . But $zh = x(ah)$ and $xh = z(a^{-1}h)$ for all $h \in H$. Therefore $zH \subset xH$ and $xH \subset zH$, and thus $xH = zH$. Similarly $yH = zH$, and thus $xH = yH$, as required. \square

Lemma 2. Let H be a finite subgroup of a group G . Then each left coset of H in G has the same number of elements as H .

Proof. Let $H = \{h_1, h_2, \dots, h_m\}$, where h_1, h_2, \dots, h_m are distinct, and let x be an element of G . Then the left coset xH consists of the elements xh_i for

Cos'è una dimostrazione?

Dimostrazione = sequenza di frasi che convincono il lettore che l'enunciato (tesi) valga.

Quando sono i singoli passi **corretti**?

- Individueremo **linguaggi artificiali** per scrivere i passi
Un **computer** potrà dire se sono corretti
- Corretto = **se valgono le premesse allora vale la conclusione**

Ma cosa vuol dire valere?

La logica studia la verità?

A vale quando A è vera?

- In un libro fantasy possono esserci ragionamenti sugli unicorni rosa volanti.
- Il ragionamento può essere corretto o sbagliato.
- Ma di sicuro non ci parla di verità

Ma allora **di cosa ci parla la logica?**

Correttezza e completezza

Supponiamo di aver capito cosa significa “vale”.

- 1 **Correttezza:** tutto ciò che è dimostrabile vale?
Ovvero, le nostre regole sono corrette?
- 2 **Completezza delle regole:** tutto ciò che vale è dimostrabile?
Ovvero, abbiamo messo abbastanza regole?
Se risposta negativa: cosa si può dimostrare e cosa no?
- 3 **Completezza delle ipotesi:** aggiungendo ipotesi posso dimostrare tutto ciò che vale in una determinata situazione?
Risposta negativa: **cosa non potremo mai sapere e perchè?**

Logica e automazione

Può un **programma** dirci se una dimostrazione è corretta?

Può un programma fare dimostrazioni automaticamente?

Se no, perchè?

Se sì, con quale complessità computazionale? (di quanto spazio e tempo ha bisogno)?

Una, dieci, cento logiche?

Il senso che diamo a “valere” determina le regole della logica e delle dimostrazioni in tale logica

Tante interpretazioni di valere \Rightarrow tante logiche

- 1 verità \Rightarrow logica classica
- 2 evidenza/programmabilità \Rightarrow logica intuizionista
- 3 accade \Rightarrow logica temporale
- 4 conoscenza \Rightarrow logica epistemica
- 5 possesso \Rightarrow logica lineare
- 6 ...

Esempio: se X è vero/è noto/accade/è ritenuto vero/è posseduto e quando X lo è lo è anche Y , allora Y lo è?

Outline

- 1 Il docente
- 2 Cosa studia la logica? (cenni)
- 3 Perchè la studiamo a informatica?**
- 4 Organizzazione del corso e modalità di esame
- 5 Decalogo dello studente furbo

Concetti e tecniche ricorrenti in informatica

- uso di **linguaggi artificiali** (di prova vs di programmazione)
- **tecniche** comuni per descriverne la sintassi/semantica
- tecniche comuni per descrivere derivazioni
dimostrazioni \approx tipaggio di programmi
- manipolazione di formule logiche \approx **manipolazione di codice**
- programmeremo tutte le definizioni del corso
 \Rightarrow introduzione di **tecniche di programmazione**
- dimostrazioni sulle definizioni programmate =
DIMOSTRAZIONI DI PROPRIETÀ DI UN PROGRAMMA

Strumenti utilissimi nel proseguio degli studi, ma siete troppo acerbi per capirlo: **FIDATEVI!**

Dimostrazioni in informatica

Quando un software è corretto?

- **specifica di un software** =
descrizione di cosa il software deve fare
- **semantica** di un linguaggio di programmazione =
descrizione di cosa fa un programma
- **descrizione = formula logica!**
- **software corretto** =
il codice fa quello che dice la specifica
il testing è sufficiente?
- software corretto se e solamente se c'è una
DIMOSTRAZIONE CORRETTA che quello che il codice fa
è quello che dovrebbe fare

Perchè circola software non corretto?

Programmare è l'attività più complessa mai ideata dall'uomo

Che succede se viene giù un ponte?

E se un software va in crash?

- *Programmare \approx dimostrare*
vedi corso magistrale Fondamenti Logici dell'Informatica
Cosa c'è di più complesso?
- *Certificazione **automatica solo parziale***
I computer privi di intuizioni **non possono** trovare
dimostrazioni

Perchè non si fanno dimostrazioni manuali/assistite allora?

Perchè circola software non corretto?

La certificazione **manuale costa!**

- dati Intel: \approx **10x mesi uomo** per dimostrazione interattiva assistita rispetto a implementazione + testing
- usata solo per codice critico

Codice critico = tale per cui un bug mette a rischio ingenti quantità di soldi (non vite umane?)

Esempi: controllo aereo, centrali nucleari, smartcard, software medicale, aritmetica floating point dei microprocessori, ...

Outline

- 1 Il docente
- 2 Cosa studia la logica? (cenni)
- 3 Perchè la studiamo a informatica?
- 4 Organizzazione del corso e modalità di esame**
- 5 Decalogo dello studente furbo

Orari delle lezioni

- Lezioni frontali:
 - lun + mer, 16:00(+15)–18:00(-15?), aula M1
- Laboratorio:
 - non tutte le settimane; si inizia il **??/??** in Ranzani
 - **AL**: 14:00-16:00(-15), **MZ**: 16:00–18:00
 - a volte in lab **Ranzani**, a volte in lab **Ercolani**
 - **gruppi da 2** persone
 - in Ranzani: uso del dimostratore interattivo di teoremi **Matita, versione 0.5.9** in ambiente **Linux**
 - in Ercolani: esercitazioni cartacee
 - possibilità di arrotondare il punteggio terminando “a casa”

Modalità di esame

Prova scritta ove vengono richieste:

- **definizioni, enunciati**, dimostrazioni
- esercizi di dimostrazione e costruzione di modelli

**Voto finale =
media pesata prova scritta +
punteggio conseguito in laboratorio**

Orale integrativo

- obbligatorio per chi non frequenta il laboratorio
- possibile per chi consegue 16-17 punti allo scritto
- possibile per chi vuole migliorare il voto

Materiale didattico

Slides: sul sito <http://cs.unibo.it/sacerdot/logica/>
messe a disposizione durante il corso

Libro di testo: “Logica a Informatica”, Asperti, Ciabattoni
le lezioni non seguiranno l’ordine del libro

Il docente

Cosa studia la logica? (cenni)

Perchè la studiamo a informatica?

Organizzazione del corso e modalità di esame

Decalogo dello studente furbo

Outline

- 1 Il docente
- 2 Cosa studia la logica? (cenni)
- 3 Perchè la studiamo a informatica?
- 4 Organizzazione del corso e modalità di esame
- 5 Decalogo dello studente furbo

Decalogo dello studente furbo

1. evitare accuratamente di prendere appunti a lezione!

Il docente è pagato per dire cose inutili e non interessanti.
Si rischia inoltre di farsi venire dubbi che rallentano lo studio.

Decalogo dello studente furbo

2. non ripassare mai durante l'anno!

Riguardare i lucidi prima della lezione dopo è una perdita di tempo, anche se ogni lezione riusa le definizioni e le notazioni delle precedenti, e non averli memorizzati significa non capire nulla.

Decalogo dello studente furbo

3. non andare mai a ricevimento durante l'anno!

Andare a ricevimento serve per rimettersi in pari su quello che non si è capito. Basta fregarsene di capire durante l'anno per evitare il ricevimento.

Decalogo dello studente furbo

4. non fate mai domande in aula!

Se farete domande il docente potrebbe capire che non avete capito o che ha spiegato male, per poi punirvi rispiegando. Peggio ancora se doveste imparare qualcosa che poi non vi serva all'esame. Per scrupolo, evitate anche accuratamente di rispondere quando il docente fa domande o sondaggi.

Decalogo dello studente furbo

5. non chiedetevi mai il perchè!

L'importante è imparare tutto a memoria senza sapere a cosa servono le definizioni e perchè i teoremi sono importanti. In ogni caso dimenticherete tutte le dimostrazioni dopo il corso (e le potete ritrovare sul Web), quindi perchè mai capire il senso dei concetti che vi servirebbe in futuro per sapere come ritrovarli e come applicarli?

Decalogo dello studente furbo

6. imparate a fare gli esercizi, non la teoria!

Agli esami vi chiedono di fare esercizi meccanici, no? Tanto per fare un esercizio meccanico non c'è bisogno di sapere cosa si sta facendo. La teoria è per quelli stupidi che poi cercano di risolvere problemi nuovi. E chi se ne frega se poi verrete rimpiazzati in azienda da un robot che fa le cose meccaniche più velocemente e per giunta correttamente.

Decalogo dello studente furbo

7. iniziate a studiare sette giorni prima dell'esame!

Perchè andare con calma e rischiare di assimilare le lezioni bruciando neuroni utili quando potete fare una tirata intensiva solo per passare l'esame, con la certezza di dimenticare tutto subito dopo?

Decalogo dello studente furbo

8. provate e riprovate gli esami senza studiare!

Studiate in Italia, l'unico posto meraviglioso che offre ben sei appelli l'anno per materia. Se ci sono, vuol dire che dovete usarli tutti, no? Continuate a provare l'esame senza studiare fra un appello e l'altro, tanto prima o poi passerete. E se non dovesse succedere, tutti i docenti a un certo punto cedono per sfinimento e un 18 ve lo danno! Ok, magari un giovane laureato trentenne con la media del 18 non è il massimo, ma perchè strafare?

Decalogo dello studente furbo

9. *copiate!*

Imparate a copiare prima che potete: magari sarà l'unica abilità che conseguirete all'università, ma vi tornerà utile di sicuro. Fatelo soprattutto quando siete così persi da non capire nemmeno che state copiando da uno che sta sparando le risposte a caso. E poi si sà, i docenti sono sempre di buon umore quando correggono per la trentesima volta la stessa soluzione sbagliata. Per non parlare dei datori di lavoro che assumono due informatici per fare il lavoro di mezzo.

Decalogo dello studente furbo

10. date prima gli esami facili!

Più esami facili darete per primi, più vi sembrerà di fare progressi in fretta. Non vi preoccupate di quando arriverete ad avere solo gli esami super-difficili e lunghi da dare.