

Automaadid, keeled, translaatorid

Varmo Vene & Vesal Vojdani

Arvutiteaduse Instituut
Tartu Ülikool

Administratiivinfo

- LTAT.03.006 (6 ainepunkti)
- Loengud: 56 tundi
- Praktikumid: 32 tundi
- Iseseisev töö: 68 tundi



Kaugõppe korraldus

- Kolmapäevane loeng täpselt kell 10:15-12:00 Deltas ja Zoomis.
- Esmaspäeva ja teisipäeva praktikumid samamoodi Deltas ja Zoomis.
- Interaktiivne teater, mida ei saa video peale võtta. **Pigem osale aktiivselt!**
- Reedel paneme videod välja!
(Lisaks on reedeti vabatahtlik huviring.)



Automaadid, keeled ja translaatorid 2022/23 kevad



Üldinfo

Õppekorraldus

Reeglid

Töövahendid

Projekt

Ajakava

Üldinfo

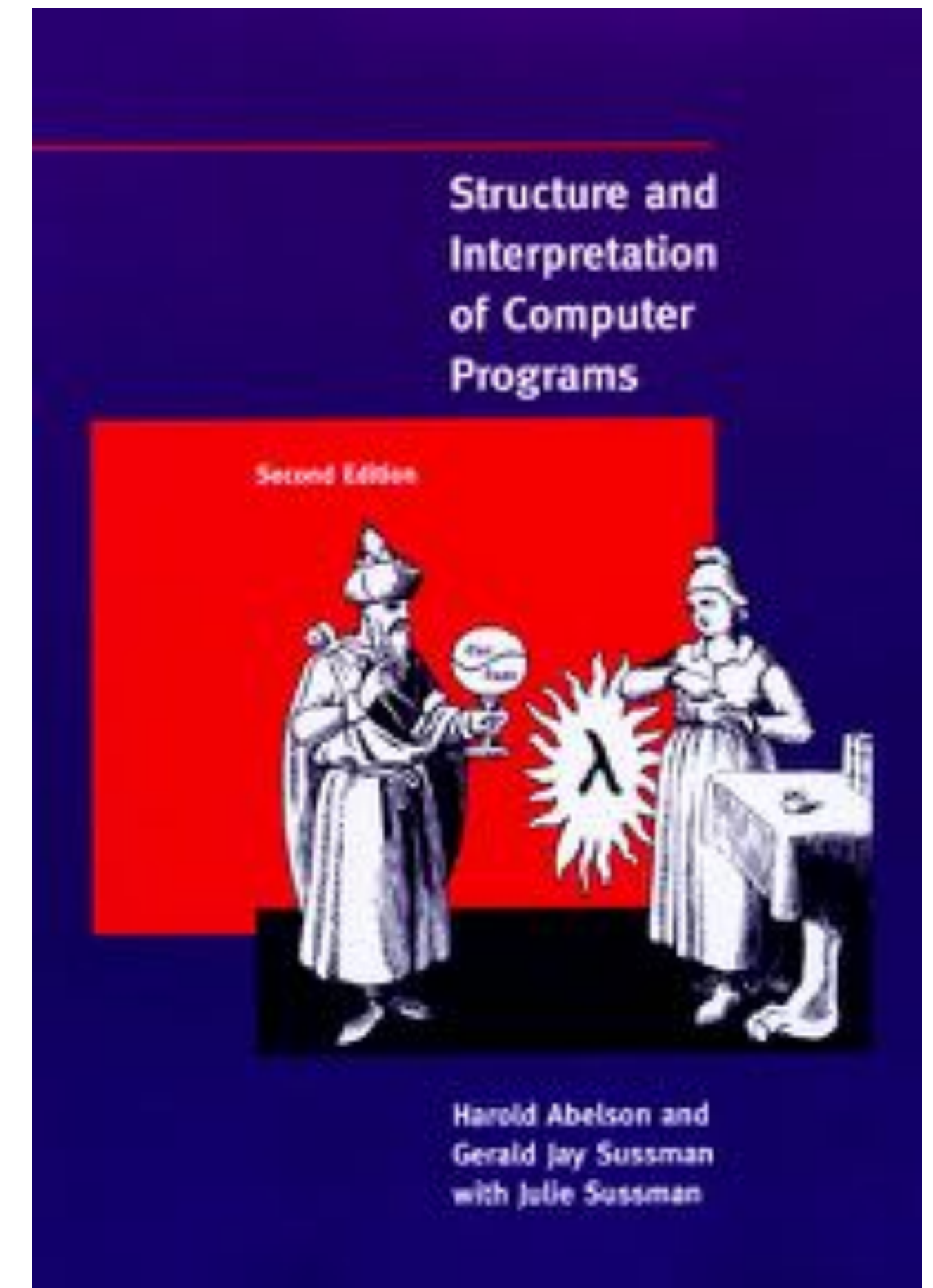
- Ainecode: LTAT.03.006 (6 EAP).
- Loengud:
 - K 10:15, Delta-1021, Vesal Vojdani & Varmo Vene.
 - R 14:15, Delta-1037, Simmo Saan & Vesal Vojdani.
- Praktikumid:
 - E 16:15, Zoom/Delta-1019, Helena Talimaa & Karoline Holter.
 - T 16:15, Delta-1019, Oliver Vainumäe & Karoline Holter.
- Praktikumideks on vaja kaasa võtta sülearvuti, millel on töötav Java arenduskeskkond.
- Õpik: [Introduction to Compiler Design](#); CMa kohta [Compiler Design: Virtual Machines](#).
- Tutvuge kindlasti [õppekorralduse](#) ja kursuste [reeglitega](#).

<https://courses.cs.ut.ee/2023/AKT>

Kursuse veebileht

Kursuse eesmärk

- Kursus süvendab tudengite arusaamist arvutiprogrammide **ülesehitusest ja täitmisest**.
- Selline arusaamine on hädavajalik **usaldusväärsema** tarkvara arendamiseks.
- Kursuse põhieesmärk on vähendada halvast programmeerimisest tingitud **kahju ühiskonnale**
- ... ja suurendada heast programmeerimisest tulenevat **rõõmu teile endile**.



Afektiiivsed väljundid!

- Tudeng **tunnistab**, et halb programmeerimine teeb teistele inimestele haiget.
- Tudeng **hindab** koodi korrektsuse moraalsust ACM kutse-eetika koodeksi valguses.



Kursuse läbinu oskab...

- **Koostada** kontekstivaba grammatika etteantud programmeerimiskeele süntaksi täpseks iseloomustamiseks kõnekeelse kirjelduse ja näidete põhjal.
Rakendada parseri loomise ning puu läbimise meetodeid, et antud keeles kirjutatud programmide abstraktseid süntaksipuid kätte saada.
- **Implementeerida** interpretaatori, mis väärtustab programmi selle süntakspuu põhjal.
- **Luu** translaatori, mis teisendab puu virtuaalmasina käskude jadaks.
(See on tegelikult suhteliselt lihtne: ainult maksimumi saavutamiseks võib olla mõni raskem samm.)

Põhiosa

Alusosa

Lõviosa

Hinde kujunemine

- Kodutööd (30 punkti)
- Eksam (70 punkti)
- Loengute ActiveQuiz (10 punkti)
- Seega on kokku võimalik koguda 110 punkti.

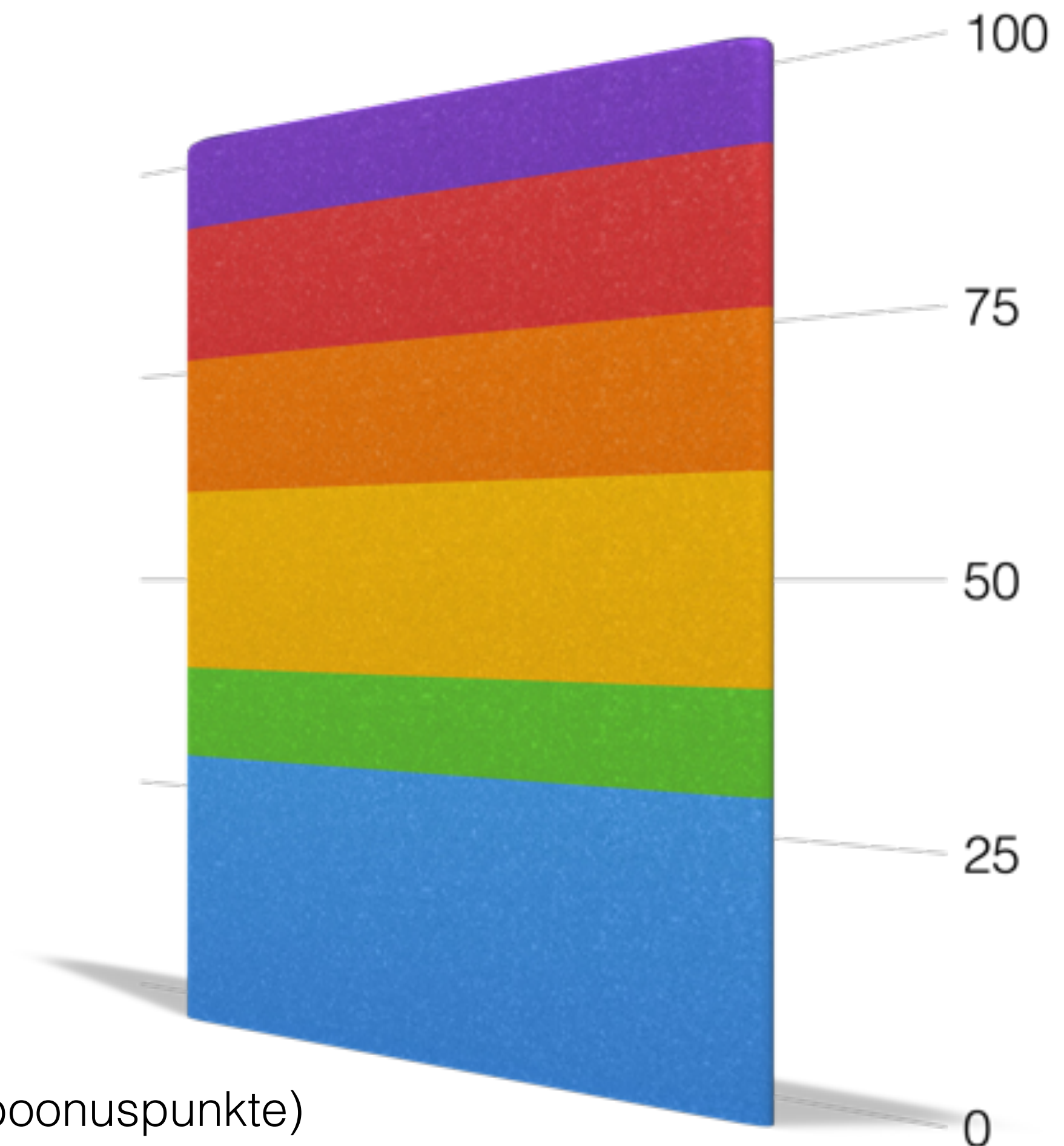
Hinne		Punkte
A	Suurepärase	≥ 91
B	Väga hea	81-90
C	Hea	71-80
D	Rahuldav	61-70
E	Kasin	51-60
F	Puudulik	0-50

Eksam: kirjuta translaator!

- Praktiline osa (40 punkti): lihtsa keele jaoks interpretaator ja translaator valmis kirjutada.
- Teooria osa (30 punkti): teooriatest ja automaatide joonistamine.

- Baitkoodi genereerimine
- Grammatika ja parser
- Avaldise väärtustamine
- Teooriatest (moodle)
- Automaatide joonistamine
- Enne eksamit

(lisaks loengute eest 10 boonuspunkte)

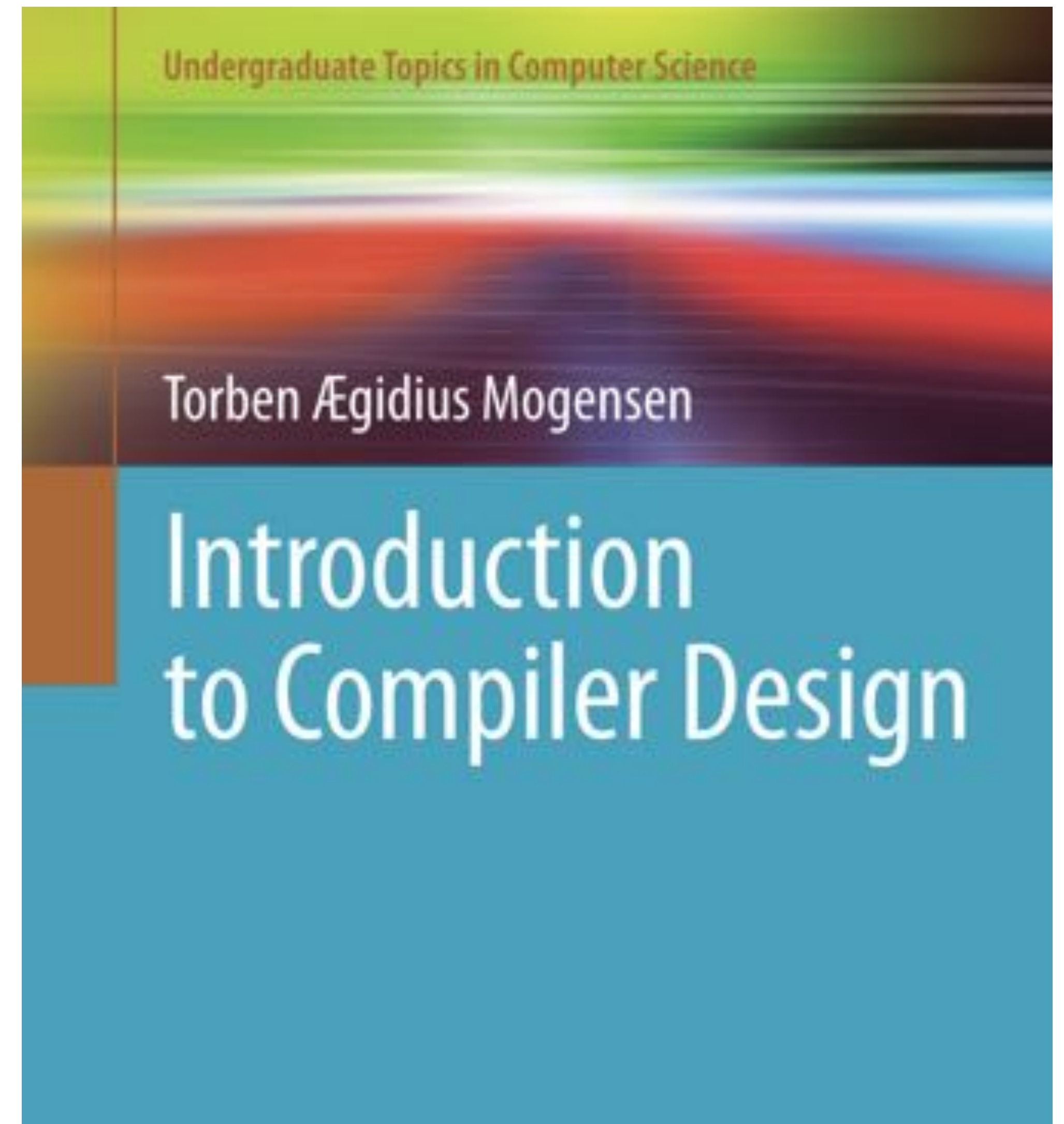


Kodutööd (30 p)

- Aines on 10 põhilist kodutööd ja iga kodutöö annab 1 punkti.
- Kodutööde eest saab punkte esitamise eest! (Vähemalt ühe testi peab läbima ja praktikumides aidatakse selleni jõuda.)
- Ülejäänud 20 punkti saab iganädalaste moodle'i-testide eest.

Kirjandus

- Õpik on ülikooli raamatukogu kaudu kõigile kättesaadav.
- Reedeks paneme järgmise nädala video välja!
- Järgmisel loengul eeldame, et video on vaadatud (ActiveQuiz & **arutelu**).
- Loengu järel oleks mõistlik õpikut lugeda.



Zulip!

- Tagasiside andmiseks kasutame Zulipi keskkonda!
- Sinna saab sisse logida otse UT kontoga.
- **NB! See on ATI ühine server.** Teisest nädalast paneme AKT toad privaatseks, seega palun liituda (*subscribe*) abikeskuse ja rühma kanaliga.
- Individuaalse tagasiside saamiseks peate ise vähemalt ühe väikese sammukese tegema—juhendajaga ühendust võtma.



Üliõpilaste tagasiside
(ja meie lahendused)

Õiged arvamused

- “Super aine selles mõttes, et see arendab sind programmeerijana meelelt, kuid see nõuab vastavalt ka korralikult tööd.”
- “Aine teemad on kohati päris keerulised. Seevastu eksami ülesanded on ette teada ja kui vähegi pingutada, tuleb hea hinne kergelt”
- “Alustage kodutööga enne praktikumi, et praktikumi ajal oleks juba eesmärgi arusaam ja küsimused, mida küsida.”

Peamised kriitikad

- “Käige kohal igal pool, kus saate, **ärge usaldage** seda, kui näiteks öeldakse, et piisab ainult ühes praktikumis kohal käimisest.”
- “Punktisüsteem absoluutselt **ei pane õppima**. Enne eksamit saab väga minimaalse pingutusega maksimumpunktid igalt poolt kätte ja seega peab enne eksamit materjali põhimõtteliselt nullist selgeks tegema endale.”
- “Eemaldaks esiplaanilt kõik lisaülesanded ja vähendaks olulisel määral kõiki kursuse tekste. Oleks palju mugavam, kui õppejõud oma terve mõttekäigu kirja panemise asemel kirjutaks ainult sellest, **mis on tegelikult vajalik**”

Meie vastus

- Punktisüsteemiga oli valdav enamus väga rahul!
- Teie oskused ja motivatsioonid on erinevad: me laseme teil ise otsustada, kui palju tahate programmeerimisoskusele keskenduda.
- Üritame nüüd selgemini välja tuua põhitee läbi kursuse, aga kõik lisaülesanded jäävad...
- Eksamist räägime varem.
NB! Palun proovieksamitel kaasa teha!

Põhitee seis aastal 2023!

- Courses lehel on kõik kätte saadav.
- Moodle'i keskkonnas oleme siin üritanud sellist selgemat teed kaardistada.
- Me saame nüüd ka mõned asjad tee pealt ära visata, sest...



Süvenduspraktikum!

- Aine temaatika on väga põnev!
- Ja seotud meie teadustööga:
tõetruu programmianalüüs.
- **Simmo Saan** ja **Karoliine Holter** on väga targad ja toredad — ärge jääge sellest tarkusest ilma.
- Simmol on põnev programm loodud... tulge reedel kohale!



Praktikumide korraldus

- Neljapäeval on kodutööde tähtaeg.
Juba enne seda ilmub uus kodutöö!
- Reedel vaatame eelmise kodutöö, lisaülesandeid ja muidki huvitavaid asju!
- Üritage enne oma praktikumi kodutööga tegeleda!
- Esmaspäeval-teisipäeval on praktikumid, kus näidatakse ära, kuidas kodutöö lahendada ja kodus jääb see lõpuni teha.



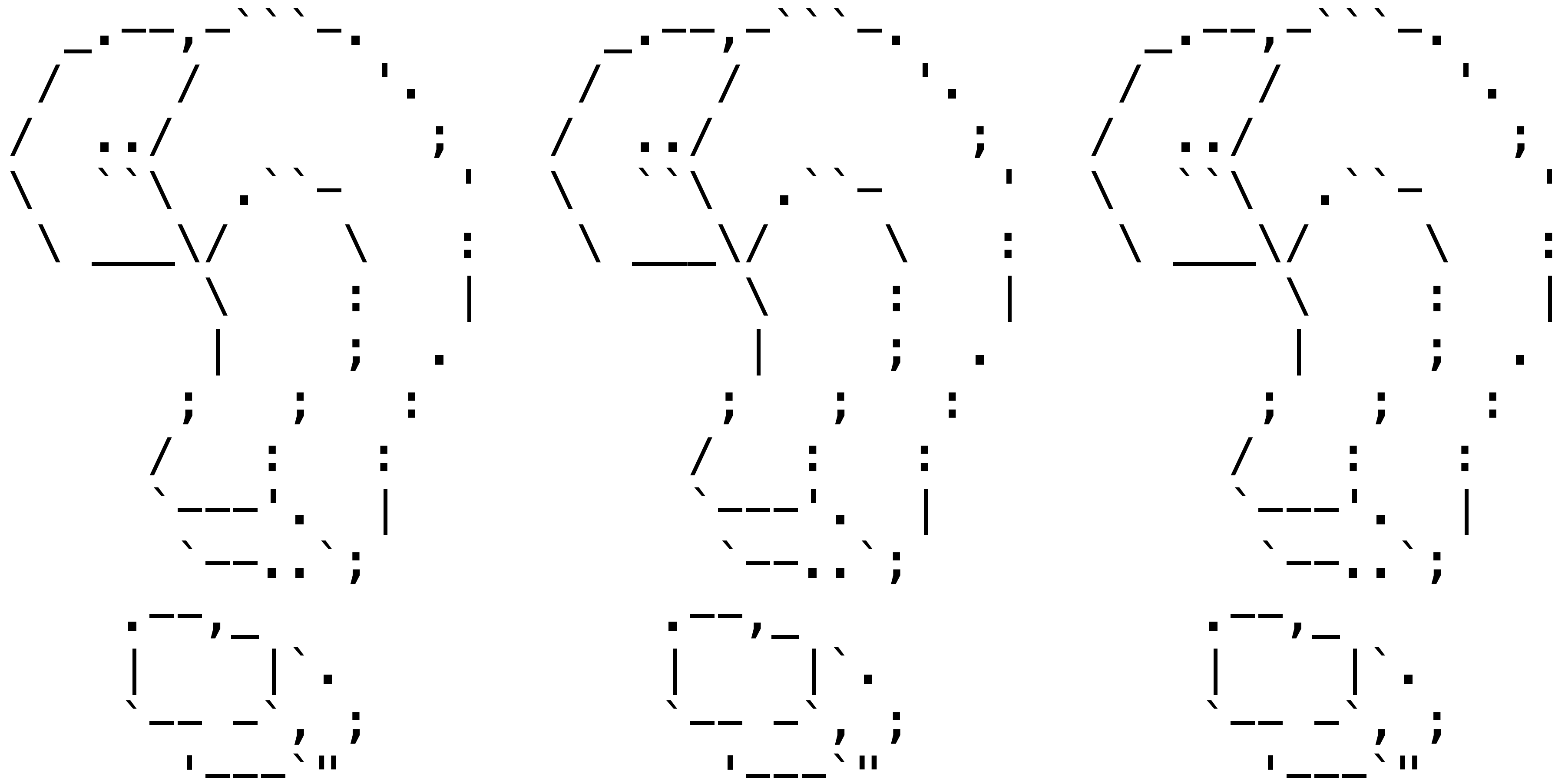
Spoiler
alert!

ChatGPT ja Copilot

- Programmeerimine on **lihtne!**
Raske on kindlaks teha, et programm on **õige!**
- AI assistentide kasutamine on siin aines lubatud, aga...
 - Meie lahendused on ilmselt GitHubis olemas, seega ta võib olla proovieksamil parem kui eksamil.
 - Alusosa on mõeldud **olulise mõtlemisoskuse arendamiseks** — seal tehke ise ka natuke trenni!
- Pigem kasutada GitHub Copilot: meil on siin suur koodibaas, teenus alati üleval ja tudengitele tasuta.



<https://meta.stackoverflow.com/questions/421831/temporary-policy-chatgpt-is-banned>

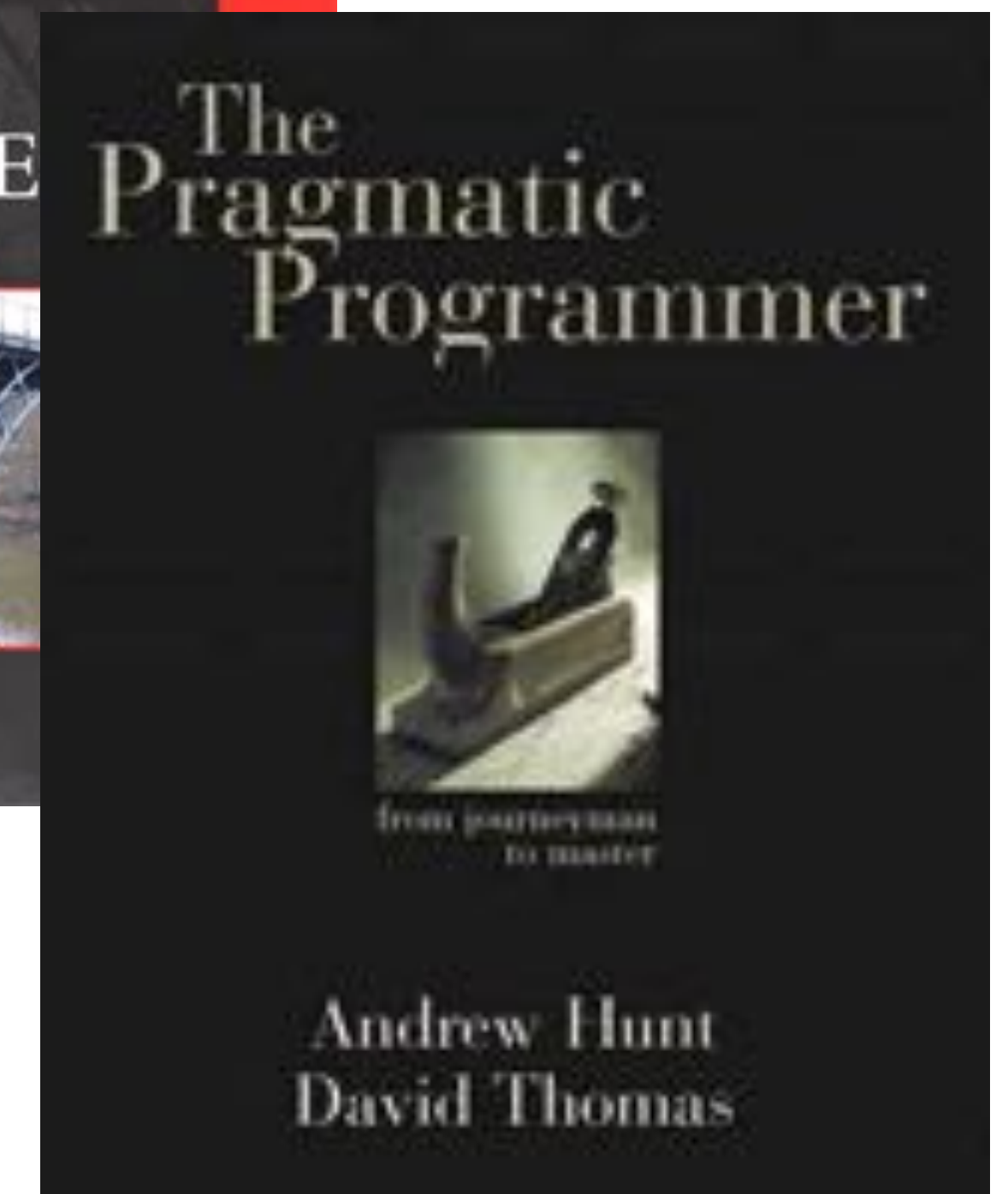
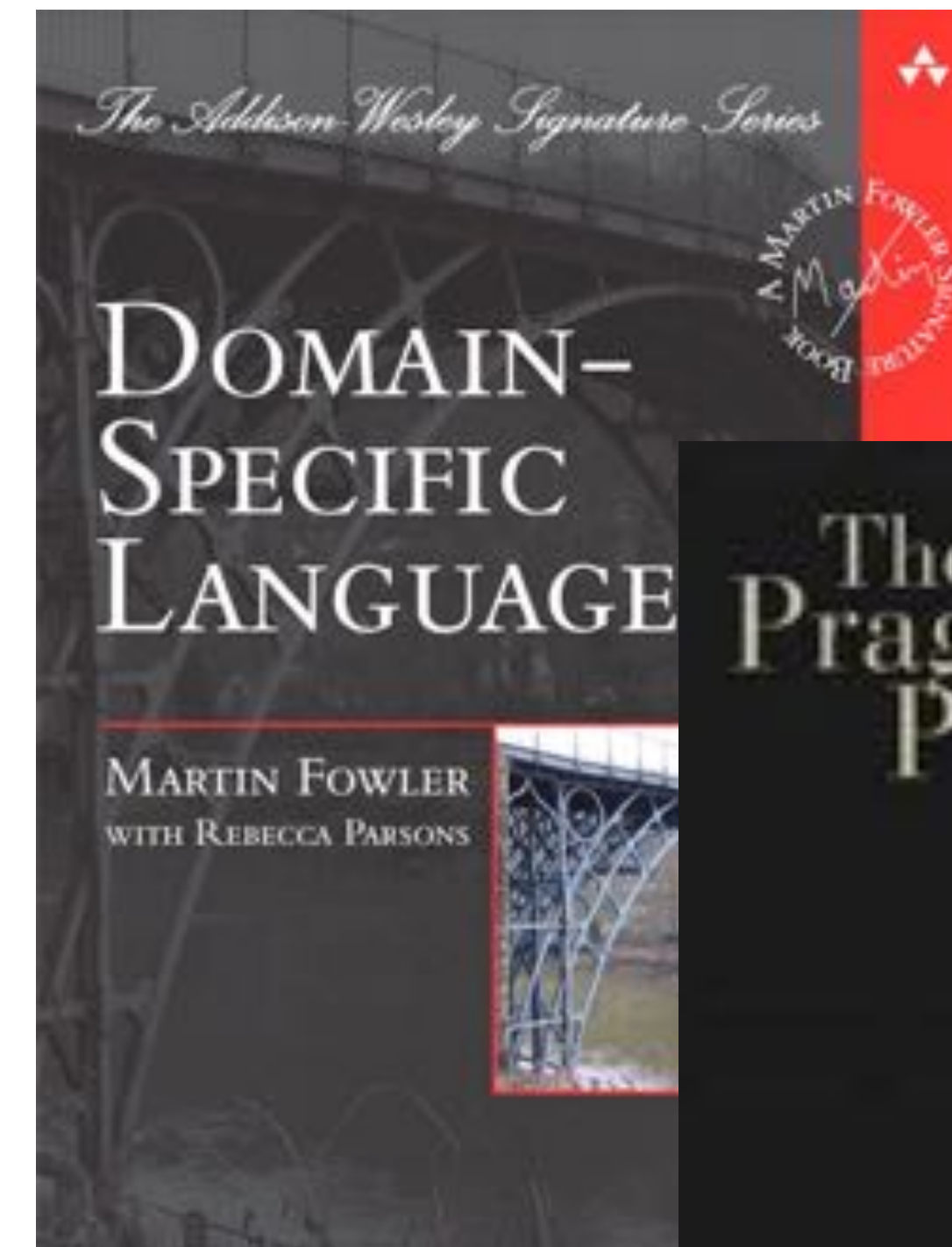


Translaatori realiseerimine

OMG Keda see huvitab?

DSLide loomine

- Selle asemel, et 20 progejat peaksid aru saada & implementeerima äri loogikat...
- Piisab, kui 1 AKT tudeng loob DSLi, milles ärimehed ise saavad äri loogika defineerida.
- Tõsisemalt, PP tip #17: “Program Close to the Problem Domain”
- “The best code is no code at all.”
— Jeff Atwood



Aga miks on aine kohustuslik?

- 1992 Bose Award (MIT's School of Engineering teaching award)
- 1995 Taylor L. Booth Education Award.
- 2011 ACM Karl V. Karlstrom Outstanding Educator Award
- 2012 ACM SIGCSE Award for Outstanding Contribution to Computer Science Education



Süüdlane: Hal Abelson

“If you don’t understand interpreters,
you can still write programs;
you can even be a competent programmer.
But you can’t be a master.”

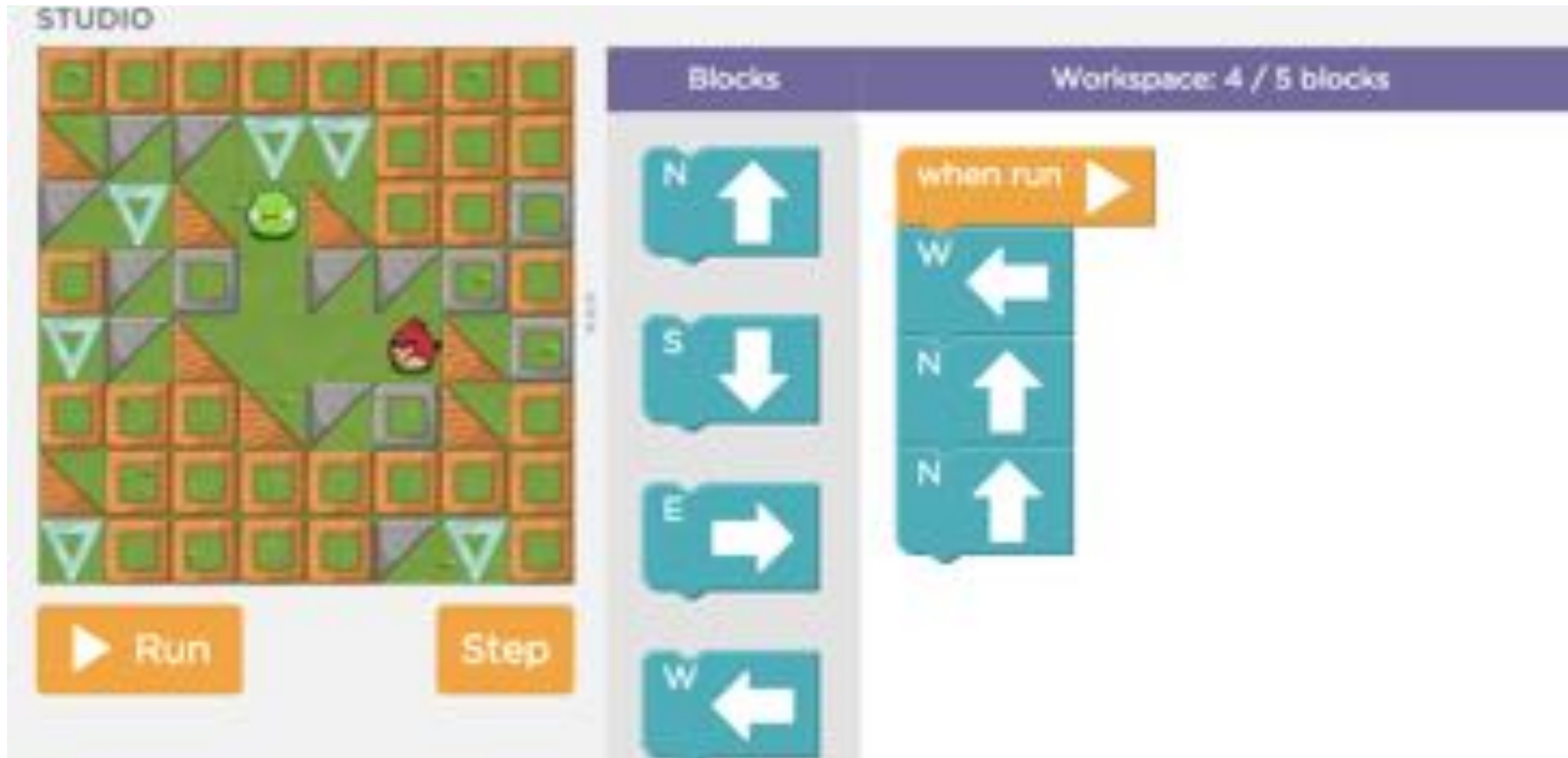
–Hal Abelson

Programmi tähendus

- Jah, programmidel on tähendus! (semantika)
- Programm ei ole lihtsalt sümbolite jada, mida niisama stackoverflow'st kopeerida.
- Hea programmeerija omab intuiitivset arusaamist programmi täitmisest.
- Seda nimetatakse operatsiooniliseks semantikaks.

code.org

- Lesson 1 & 2: Flurbi liigutuste paberile liimimine.
- Lesson 3: Hiire liigutamise harjutused.
- Lesson 4: Angry Bird'i liigutamine klotsidega.
- Lesson 5: **Debugging**.



Kuhu lind läheb?

Siin me oskame ennustada, kuhu lind läheb, sest me oskame peas linnu sammud läbi astuda.

Küsimus 4-aastastele!?

- <https://code.org/curriculum/course1/5/Teacher>
- “Students will predict where a program will fail!”
- Minu neljaaastane tütar lihtsalt võttis kõik klotsid ära ja hakkas uuesti algusest tegema.
- Mõistagi ma sain väga vihaseks...

Küsimus Teile

- Kas oskate Java programmi käitumist ennustada?
- Reflekteerige: Kas minu arusaamine arvutiprogrammist on süntaksi tasemel või saan hästi aru, kuidas programmi täidetakse?
- Kas oskad põhjendada, miks antud programm töötab õigesti?
- Kas põhimõtteliselt kujutad ette, kuidas saaks veenduda, et programm töötab õigesti või saab seda ainult katsetada/testida?

Milleks AKT?

- Kuidas ma kontrollin, kas te tõesti saate Java programmi tähendusest aru?
- Ma võiks anda ülesande kirjutada translaator, mis Java programmist genereeriks baitkoodi.
- Aga äkki töötab ka vastupidi: Kui ma kirjutan translaatorit, siis ma saan natuke paremini aru Java programmi tähendusest.

Järgmiseks korraks



- Töövahendite paigaldus! 
- Lugemine: Preface (lk v-ix) ja esimene peatükk (lk 1-6).
- Kuulata kompilaatoritest BBC sarjast 50 Things That Made the Modern Economy

“There are estimates that the number of programmers needed in the U.S. exceeds 200,000. This is entirely misleading. It is not a quantity problem; we have a quality problem. One bad programmer can easily create two new jobs a year. Hiring more bad programmers will just increase our perceived need for them. If we had more good programmers, and could easily identify them, we would need fewer, not more.”

–David Parnas

Dekoratiivsete piltide autorid ja litsentsid.

- COVIDi morfoloogia illustratsioon. Alissa Eckert, MSMI, Dan Higgins, MAMS (avalik omand).
- Hal Abelson. Joi Ito (CC-2.0-BY).
- Grace Hopper. Lynn Gilbert (CC BY-SA 4.0).