

# Programmeerimise põhikursus Javas (ITI0011)

Praktikum 01 - sissejuhatus, Hello World

# Aine korraldus

- <https://ained.ttu.ee/>
    - Registreeruda ainele
    - Materjalid, tulemused, foorum
  - <http://courses.cs.ttu.ee/pages/ITI0011>
    - Materjalid
- 
- 2 praktikumi nädalas
  - 20 harjutusülesannet
  - 4 koduülesannet
  - 3 tunnikontrolli
  - eksam

# Hinde saamine

- 20 harjutusülesannet x 1p = maks 20p
- 4 koduülesannet x 8p = maks 32p
  - iga kaitstud koduülesanne annab võimaluse saada 1 pall parema hinde:
    - 0 kodutööd => maksimaalne hinne "1"
    - ...
    - 4 kodutööd => maksimaalne hinne "5"
- 3 tunnikontrollid x 1p = maks 3p
- Võimalikud lisapunktid
- Enne eksamit läheb arvesse maksimaalselt 50 p
- Eksam 60p
  - vähemalt 31p peab eksamil saama, et eksam oleks sooritatud
- Punktid liidetakse,
  - 91p.. => "5", 81p-90p => "4", ..., 51p-60p => "1", 0p-50p => "0"

# Ülesannete kaitsmine

- Kõik ülesanded tuleb laadida git'i
  - emailiga saadetud ülesanded ei lähe arvesse
- Ülesande eest saab punkte pärast kaitsmist
  - kaitsmata ülesanne annab 0 punkti
- Kaitsmine toimub õppejõule
- Kaitsmise eesmärk on aru saada, kas tudeng on koodi ise kirjutanud ja kas ta saab sellest aru:
  - õppejõud küsib küsimusi tudengi koodi kohta
  - õppejõud laseb parandusi/täiendusi teha koodis
- Kui tudeng koodist aru ei saa, saab ta ülesande eest 0 punkti
- Pigem realiseerida vähem funktsionaalsust aga põhiline oleks kõigest aru saada.
- **Plagiaadi esitamise eest saavad kõik autorid 0 punkti**

# Konsultatsioon

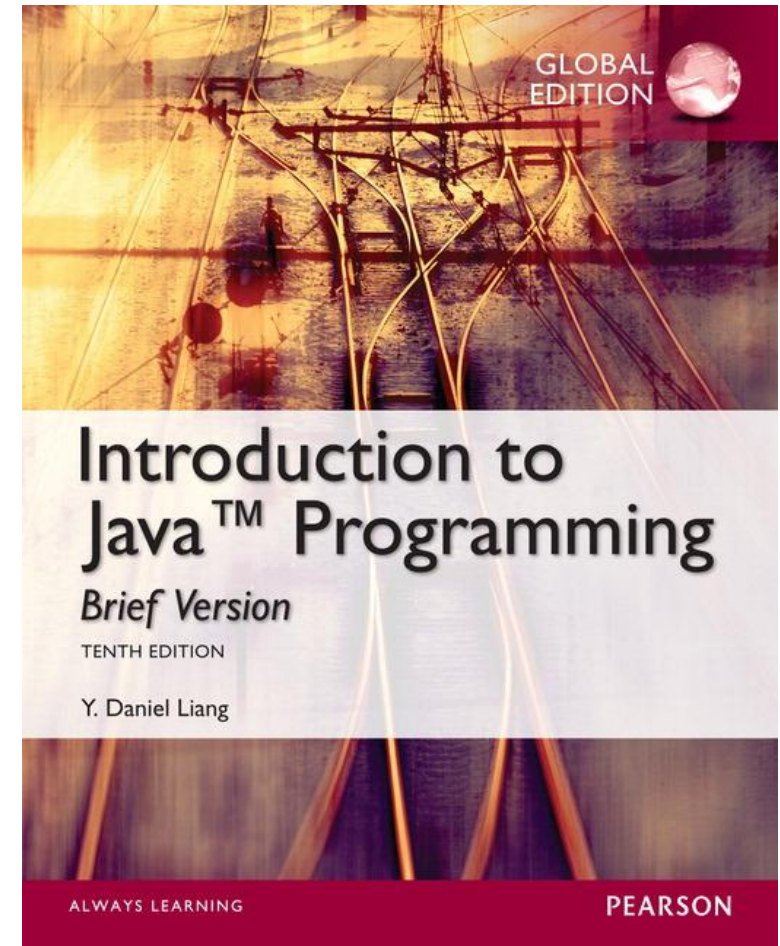
- Konsultatsioon toimub kaks korda nädalas:
  - E14.00 (ICT-403)
  - N16.00 (ICT-402)
- Kindlasti tuleb registreerida doodle'is
  - Kui registreerunud on 0 tudengit, siis õppejõud kohale ei tule
  - Vastavalt registreerunud tudengite arvule plaanime kohale õppejõud
- Konsultatsioonis saab:
  - abi lahendatavate ülesannetega seoses
  - kaitsta ülesandeid, mille tähtaeg on möödas
- Tunnis kaitsmise puhul võib õppejõud tudengi suunata konsultatsiooni, kus on rohkem aega, et tudengiga tegeleda.

# Kodused ülesanded

- Kaitsmine toimub kindlatel aegadel
- Ülesanne annab kokku maksimaalselt 8 punkti
  - sellest põhiosa (tavaliselt 4p) on kohustuslik, et ülesanne arvestatud saada
- Hilinemisel kuni kolm nädalat ehk kuni 6 praktikumi (kaasa arvatud) saab kuni 50% punktidest (8p => 4p, 7p => 3p).
  - Peale seda ülesannet enam kaitsta ei saa
- Teemad:
  - HW01 - Aardejaht, kaitsmised 4. nädalal (22. - 26. veebruar)
  - HW02 - API, veebist lugemine, kaitsmised 8. nädal (21. - 25. märts)
  - HW03 - GUI (võimalik teha Androidi peal), kaitsmised 12. nädal (18. - 22. aprill)
  - HW04 - Gomoku AI, kaitsmised 16. nädal (16. - 20. mai)
- Iga kaitstud ülesanne võimaldab saada parema hinde
  - 0 kaitstud kodust ülesannet => maksimaalselt hinne "1"
  - 1 kaitstud ülesanne => maksimaalsel "2"
  - ..
  - 4 kaitstud ülesannet => maksimaalselt "5"
  - Punktid peavad ka võimaldama vastava hinde saamist.

# Kursuse õpik

- **Põhiõpik:**  
Introduction to  
Java Programming (Brief Version)  
Y. Daniel Liang
- <http://math.hws.edu/javanotes/>  
(üldiselt hea, GUI osa on vananenud)
- Thinking in Java:  
<http://mindview.net/Books/TIJ4>  
(vanem versioon 3 on tasuta allalaetav)
- Oracle dokumentatsioon:  
<http://docs.oracle.com/javase/tutorial/>



# Iseseisev harjutamine

- <http://codingbat.com/java>
  - Väikesed ülesanded erinevate raskusastmetega
  - Eksamile tulevad samalaadsed ülesanded
  - Alustage lihtsamatest: Warmup-1, String-1
  - Seejärel saate liikuda edasi kuni String-3, Array-3
- <https://www.codecademy.com/learn/learn-java>
  - õpi ja lahenda ülesandeid
- <http://www.learnjavaonline.org/>
- MIT Introduction to Programming in Java:  
<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-092-introduction-to-programming-in-java-january-iap-2010/>



# Kursuse esialgne kava

1. sissejuhatus, hello world
2. koodi konventsioon, git, testimine, primitiivsed muutujad
3. massiiv, tsükliid, lause kontroll
4. sõne
5. veahaldus, erindid, sisend-väljund
6. sisend-väljund, andmestruktuurid
7. **HW01 kaitsmine**
8. **HW01 kaitsmine**
9. andmestruktuurid, failide lugemine/kirjutamine
10. OOP algus, objektid
11. OOP, pärimine, polümorfism
12. OOP, interface
13. API kasutamine, veebist lugemine, välised teegid
14. Sõne parsimine, regulaaravaldised
15. **HW02 kaitsmine**
16. **HW02 kaitsmine**
17. JavaFX, SceneBuilder
18. JavaFX jätk
19. JavaFX jätk
20. Antroid
21. Androidi jätk
22. GUI
23. **HW03 kaitsmine**
24. **HW03 kaitsmine**
25. rekursioon
26. rekursioon,
27. minimax
28. alpha-beta
29. gomoku, kordamine
30. gomoku, kordamine
31. **HW04 kaitsmine**
32. **HW04 kaitsmine, kordamine**

# Arendamiseks vajalik keskkond

- Java SE JDK 8u72:
  - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
  - SE - Standard Edition
  - JDK - Java Development Kit (vajalik arendamiseks)
  - JRE - Java Runtime Environment (piisav käivitamiseks, sisaldub JDK-s)
- IntelliJ community edition, 15.0.3
  - <https://www.jetbrains.com/student/> (tudengi versioon)
  - <https://www.jetbrains.com/idea/#chooseYourEdition> community edition (kes tahab, võib ka ultimate'i kasutada raha eest)
- git <https://git-scm.com/downloads>

# Java programm

- Programm kirjutatakse lähtekoodi (java-laiendiga failid, näiteks Hello.java)
- Programm kompileeritakse
  - java-failidest luuakse class-failid
  - class-failid on sisaldavad java baitkoodi
- Kompileeritud java baitkood pannakse käima
  - programm käivitatakse JVM (Java Virtual Machine) sees
  - JVM käivitab sama kompileeritud programmi sama tulemusega erinevate platvormide (Windows, Linux, Mac) peal

# Hello world!

- IntelliJ's:
  - **File** -> **New..** -> **Project**
  - Java (peaks olema valitud)
  - Midagi muutma ei pea, **Next**
  - **Next**
  - Project name - projekti nimi, näiteks "EX00"
  - **Finish**
  - Paremalt projekt "EX00", selle all "src" kaust
  - "src" (*source*) on kaust, kuhu lähtekood kirjutatakse
  - parem klikk "src" kaustal, **New..** -> **Java Class**
  - Name: HelloWorld, **OK**

# Hello World!

- Kood:

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.print("Hello world!");  
    }  
}
```

# Koodi käivitamine

- Run -> Run ... -> HelloWorld
- All paneelil on näha väljund, peaks olema umbes selline:

```
"C:\Program Files\Java\jdk1.8.0_40\bin\java" ....  
Hello World!  
Process finished with exit code 0
```

# Programmikoodi osad - klass

- Java kood on alati klassi sees
- Reeglina on ühes failis üks klass ning failinimi ja klassinimi peavad ühtima
  - klassinimi Hello peab olema failis Hello.java

```
public class HelloWorld {  
    klassi kirjeldus  
}
```

# Programmi osad - *main* meetod

- Kui java fail käivitada, otsitakse ühte konkreetset funktsiooni/meetodit: main
- main meetod on kirjeldatud järgmiselt:

```
public static void main(String[] args) {  
    käsklused/kood  
}
```



# Programmi osad - kokku

- Esialgu te ei pea aru saama, mida need kirjeldused täpselt tähendavad
- Võtke aluseks järgnev kood:

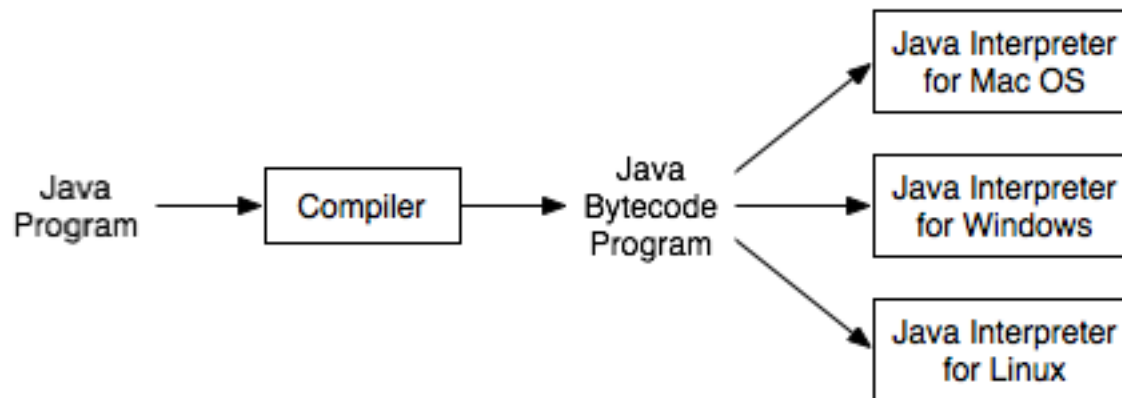
```
public class HelloWorld {  
    public static void main(String[] args) {  
        käsklused/kood  
    }  
}
```

# *main* meetodi seletus

- Märksõnad, mida *main* meetodi kirjeldamiseks kasutatakse:
  - `public` - tegemist on avaliku meetodiga (kättesaadav kõikidele)
  - `static` - tegemist on staatilise meetodiga (vastupidiselt instantsi meetodile)
  - `void` - meetod ei tagasta midagi
  - `main` - meetodi nimi
  - `()` - sulgudes on argumendid, mida see meetod aktsepteerib/nõuab
  - `String[] args` - *main* meetodile antakse automaatselt kaasa käsurealt tulevad argumendid. Konkreetsemalt on tegemist sõnede massiiviga, mis on kättesaadav muutuja `args` kaudu.

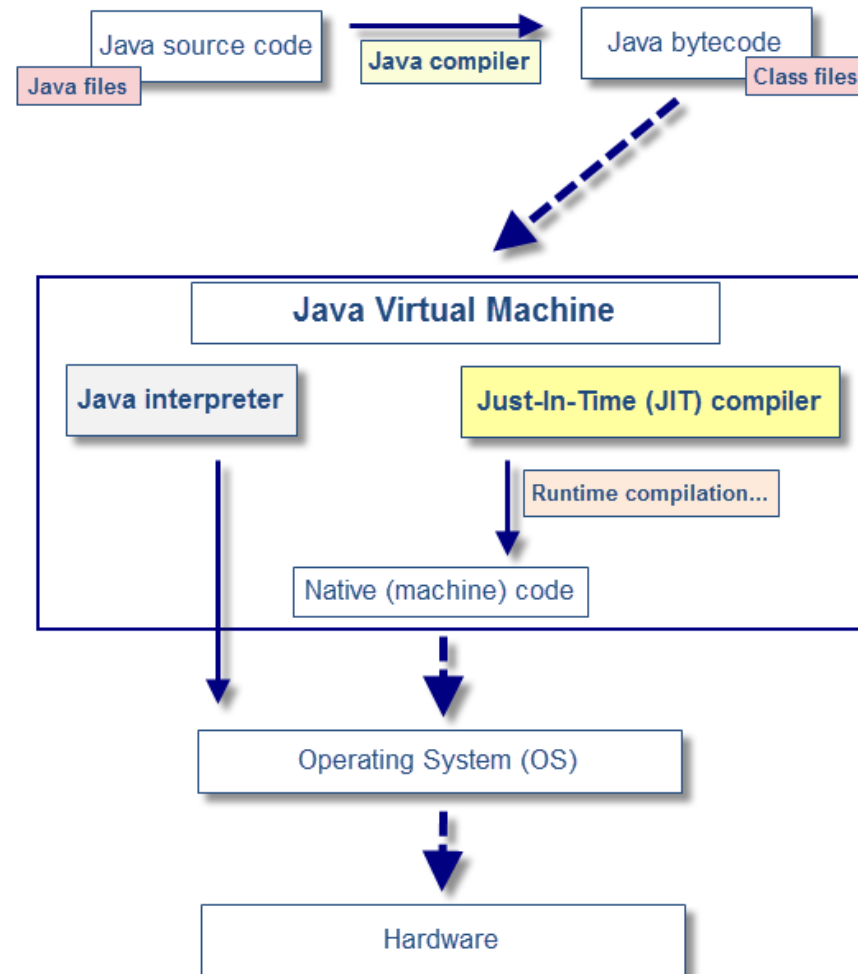
# Koodi kompileerimine

- Java lähtekood kompileeritakse Java baitkoodiks
- Java baitkood käivitatakse JVM-s (Java Virtual Machine)
- JVM on implementeeritud erinevate operatsioonisüsteemide jaoks
  - sama baitkoodi saab jooksutada erinevate platvormide peal



<http://math.hws.edu/javanotes/c1/s3.html>

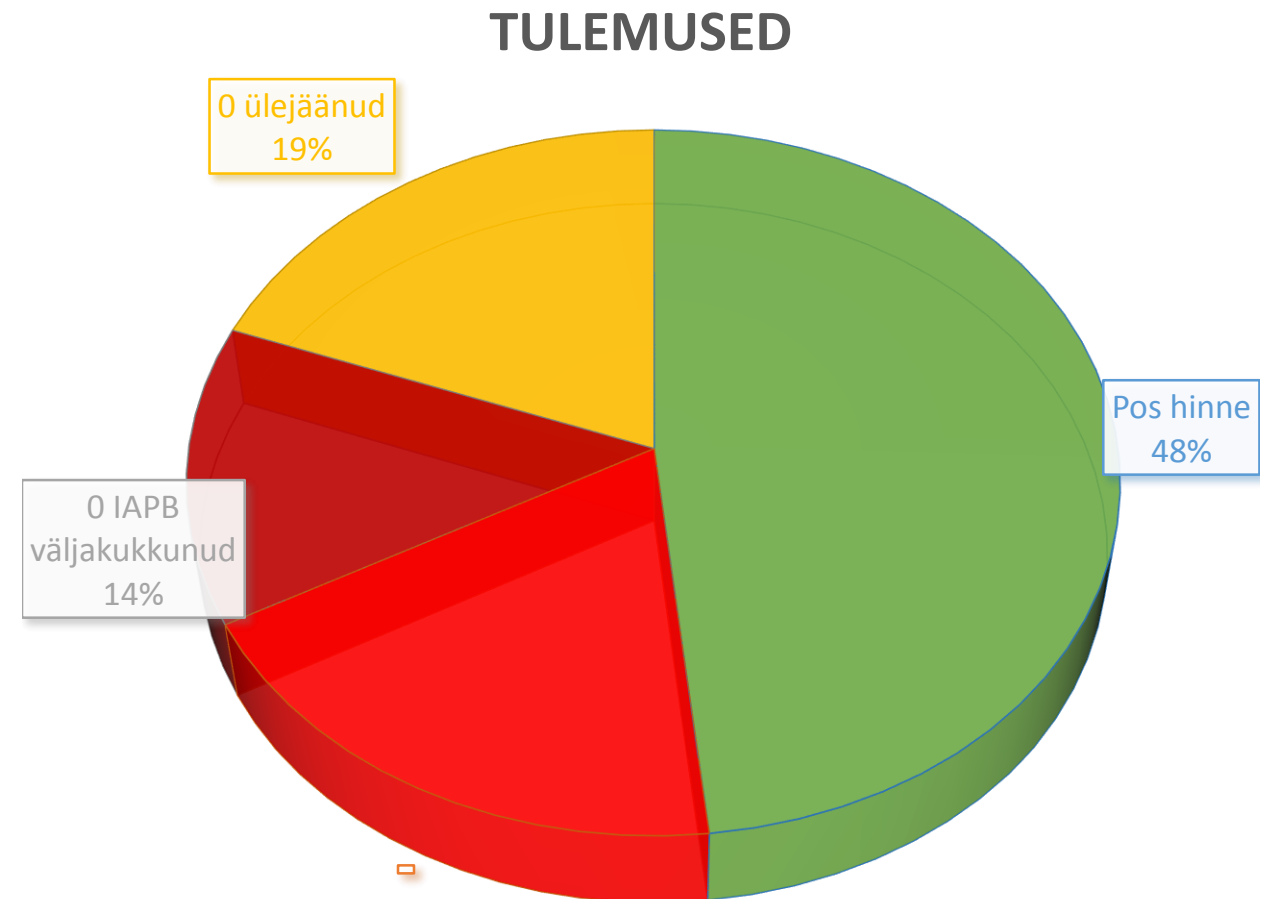
# Baitkoodi käivitamine



<http://javaeesupportpatterns.blogspot.com/2013/07/java-just-in-time-compilation-more-than.html>

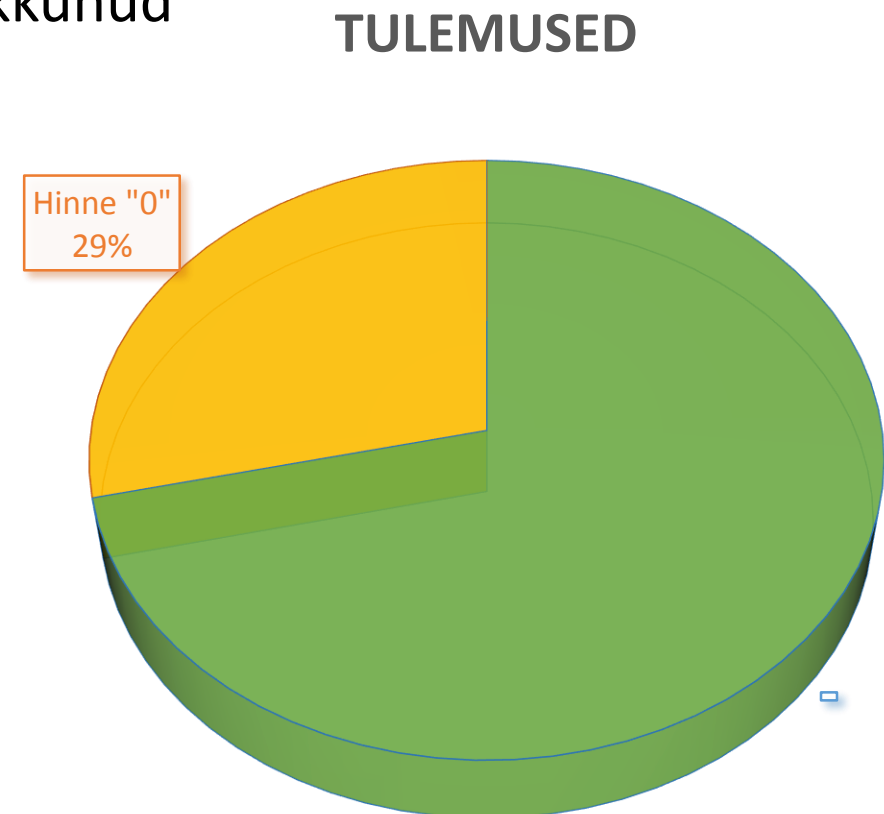
# Aasta 2015 numbrites

- Aine deklareeris 176 tudengit (punase joone seisuga)
- Positiivse hinde (1..5) sai 85 tudengit
- Läbi kukkus 91 tudengit (51.7%):
  - nendest 33 tudengit (19% kõikidest tudengitest) ei olnud IAPB õppekavas
  - 24 tudengit (15% kõikidest) olid IAPB tudengid, kes tänasel päeval enam ei õpi
  - 34 tudengit (19%) olid ülejäänud IAPB tudengid

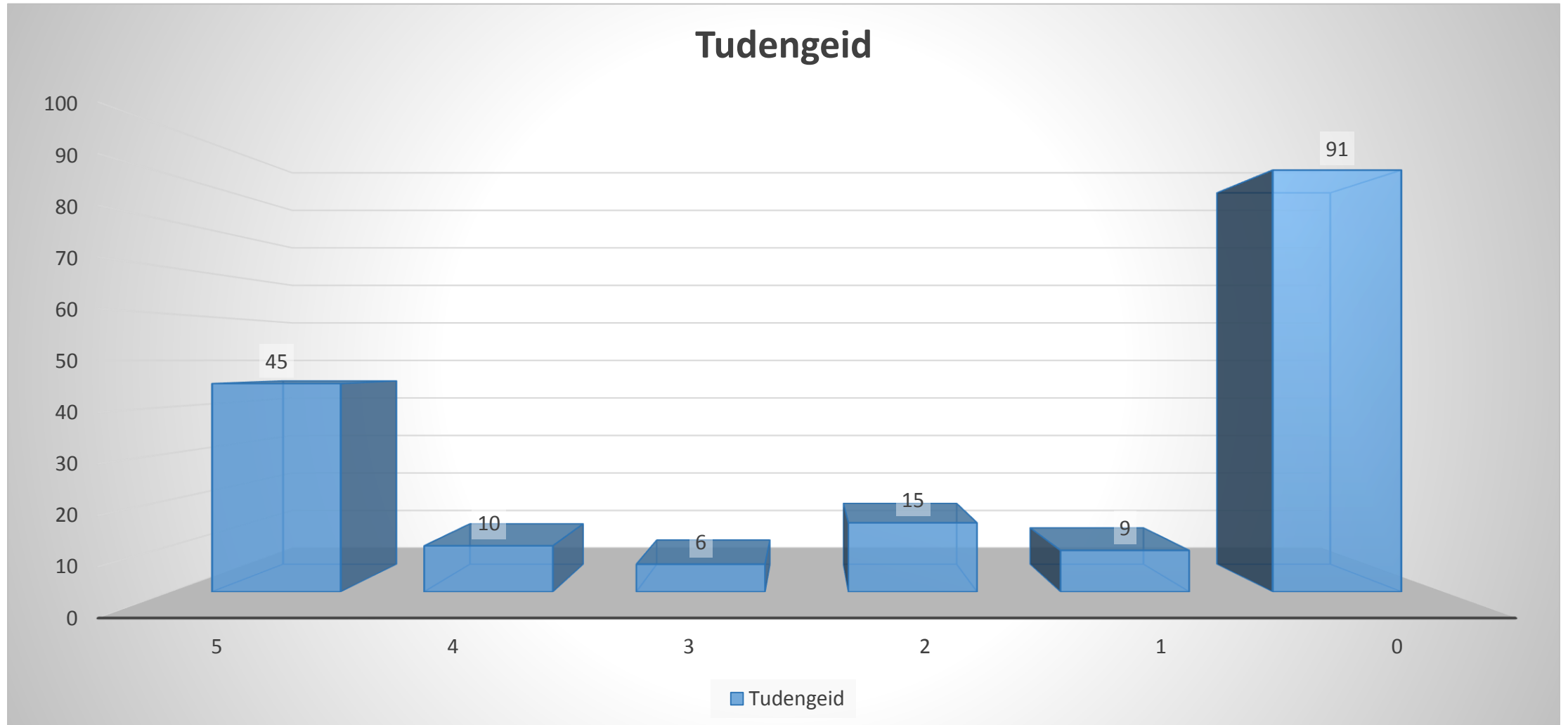


# Aasta 2015 numbrites

- Kui arvestada tudengeid:
  - kellele see aine on kohustuslik (IAPB õppekava)
  - ja kes 2. kursuse keskpaigaks pole välja kukkunud
- Positiivne hinne 85 tudengit
- Hinne "0" 34 tudengit



# 2015 lõpptulemused



# Järgmisel korral

- git
- esimene testülesanne git - aitame igal tudengil selle läbi teha
- Java primitiivsed muutujad
- tingimuslause
- funktsioonid