

4 NEDĒĻU PYTHON KURSS

RĪGAS PROGRAMMĒŠANAS SKOLA



Viena no **pirmajām programmēšanas skolām** Latvijā un Lietuvā ar vairāk kā 800 absolventiem

Mācam **intensīvu un koncentrētu** programmēšanas kursu, kas **iepazīstina** studentus ar **programmētāja darba perspektīvām**

Praktiska pieeja programmēšanai – akadēmisks standarts mācību secībai

Piedāvājam **piekļuvi skolas telpām** arī ārpus lekciju laika, lai veicinātu mācību procesu

RĪGAS PROGRAMMĒŠANAS SKOLA



Piedāvājam iespēju saviem studentiem atkārtot kursu par velti gada laikā

Visas lekcijas tiek ierakstītas un ir pieejamas privātā YouTube kanālā

Organizējam vies-lekcijas ar sadarbības uzņēmumu speciālistiem

Veicinām komunikāciju starp pasniedzējiem un studentiem

Palīdzam atrast prakses/darba vietu pēc absolvēšanas

Izsniedzam **sertifikātu** par 40 akadēmisko stundu apmeklējumu, kā arī pasniedzēju **rekomendācijas**

Valdis



- o Izglītība: Maģistra grāds datorzinātnēs
- o Pieredze programmēšanā: 20+ gadi
- o Hobiji: šahs, prāta spēles, riteņbraukšana
- o E-pasts: valdis.saulespurens@gmail.com



Ko es mācīšu?



- o Python
 - o levads programmēšanā
 - o Sintakse
 - o Datu struktūras un loģikas principi
 - o Klases / OOP
 - o Konsoles aplikācijas
 - o Datu apstrāde
 - o Datorā iebūvēto iespēju izmantošana

Ko es mācīšu?





Kā atrast risinājumus

Kādēļ Python?

- Python izstrādātāji ir pieprasīti darba tirgū
- TOP 5 valoda dažādās nozarēs
- Python gan profesionāliem, gan amatieriem
- Python lasāmība
- Ātri prototipi
- Python lietotāji(Google, Reddit, Instagram, Dropbox, NASA, Netflix, Spotify, un neskaitami citi)











Kas ir programmēšana?





Kas ir programmēšana?



Programmēšana – Formālu(!) instrukciju nodošana datoram lai veiktu dators veiktu kādas darbības.

Programmēšanas valoda – formāla valoda, kurā nofiksētas instrukcijas datoram.

Sintakse – forma (Kā jāraksta) Semantika – nozīme(meaning) (Ko dara?)

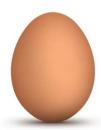
Algoritms



•Kā uzvārīt olu?

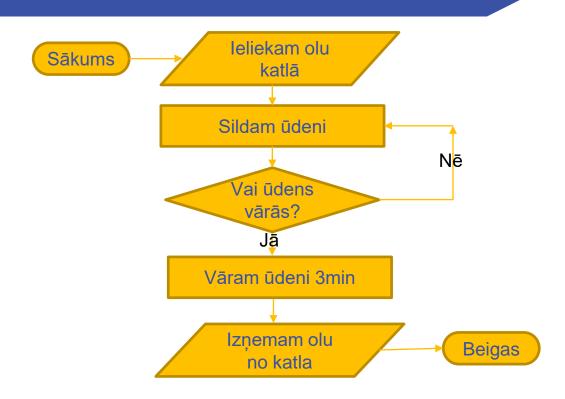






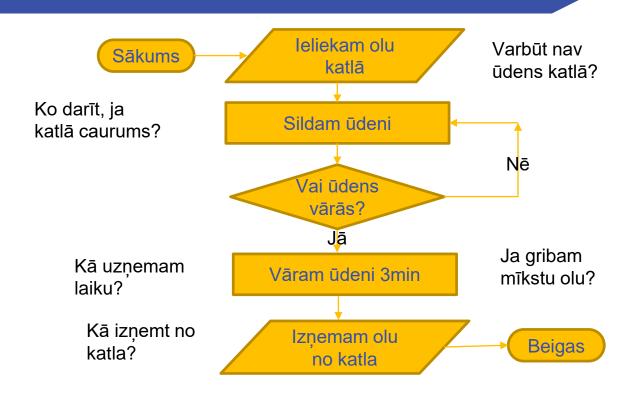
Algoritms





Algoritms





Abstrakcija



- Abstraction metode, kā tikt galā ar komplekso.
- Visu nevar un nevajag zināt!
- Slēdža princips



Kas vajadzīgs programmētājam?



- Spēja iedziļināties
- Abstrakta domāšana
- Vienkārša matemātika (01)
- Loģiska domāšana



Datoru un Programmēšanas Vēsture



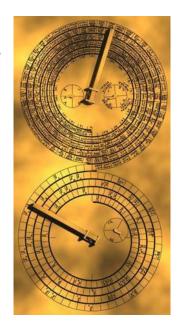
•Kas tas varētu būt?



Antikythera mehanisms



- •Analogs dators ~150-100 p.m.ē
- Pielietojums
 - Astronomija/Astroloģija
- Kalendārs
- •Olimpiādei?



Blaise Pascal 1623 - 1662



- -Zinātnieks/filozofs/matemātiķis
- Mehaniskā kalkulātora izgudrotājs
- Pascalline
 - Saskaita/Atņem



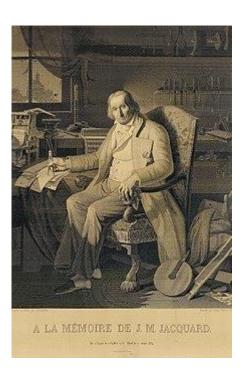


Joseph Marie Jacquard 1752 - 1834



- •franču audējs/tirgotājs
- Jacquard loom
- •punched cards instrukcijām/datiem

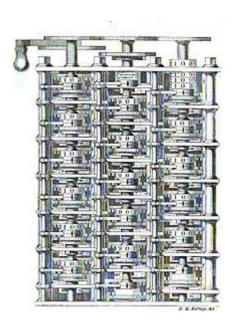


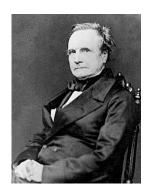


Charles Babbage 1791 - 1871



- ■izgudrotājs / matemātiķis datora «tēvs»
- Difference Engine sarežģīts kalkulātors
- Analytic Engine
 - pirmais vispārēji programmējamais dators
 - •izvēles instrukcija conditional branching
- Garbage In Garbage Out princips

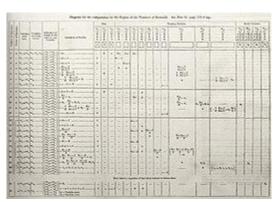




Ada Lovelace 1815 - 1852



- •matemātiķe/rakstniece
- •pirmā programmētāja pasaulē
- Analytic Engine popularizētāja
- dzejnieka Bairona meita

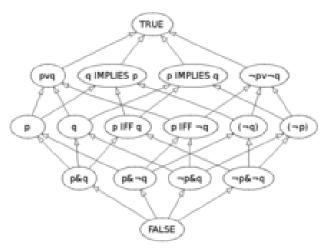


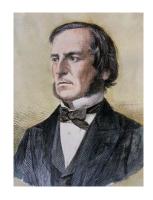


George Boole 1815 - 1864



- •matemātiķis pašmācības ceļā
- Būla loģikas izgudrotājs
 - **-**AND
 - ■OR
 - ■NOT
- •Izmanto 0,1 Nepatiess/Patiess





Alan Turing 1912 - 1954



- datorzinātnes pamatlicējs
- •kodu atšifrētājs WW2 laikā
- Turinga tests
- Turinga valodu vienlīdzība

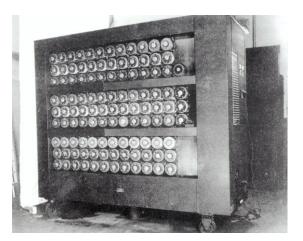


Image supplied by Bletchley Park Trust/SSPL.



Konrad Zuse 1910 - 1995



- •Elektro-mehaniskā datora izgudrotājs
- Z3 izmanto relejus(1941)
- •Plankalkul pirmā programmēšanas valoda augstā līmeni





John von Neumann 1903-1957



20gs. izcilākais daudznozaru matemātiķis
kvantu fizika,spēļu teorija u.c
Cellular automata
von Neumann datoru arhitektūra
dati/programma vienlaidus



Programmēšanas valodu evolūcija 1945-1965



- •FORTRAN (1954) matemātika
- •LISP (1958) AI
- •COBOL (1959) bizness
- •ALGOL (1960) komiteja
- BASIC (1964) iesācējiem



Programmēšanas valodu evolūcija 1965-1980



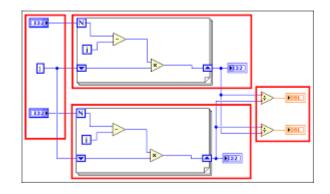
- •C (1972) Unix sistēmu
- Pascal (1970) mācību
- ■Prolog (1972) deklaratīva
- •SQL (1978) datu bāzēm
- •C++ (1980) vispārēja sistēmu



Programmēšanas valodu evolūcija 1980-1990



- -Ada (1983) komiteja
- •MATLAB (1984) matemātika
- •LabVIEW (1986) vizuāla
- •Perl (1987) skriptu
- •Matematica (1988) matemātika



Programmēšanas valodu evolūcija 1990 - 2000



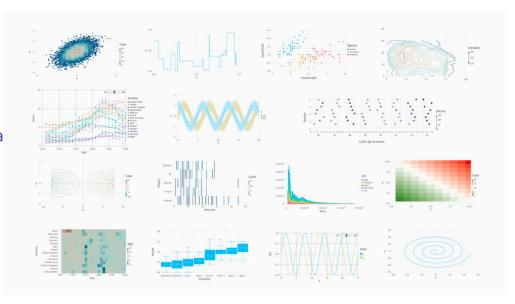
- ■Python (1991) vispārēja
- •Java (1995) vispārēja
- •Javascript (1995) skriptu
- •PHP (1995) skriptu
- •Ruby (1995) vispārēja



Programmēšanas valodu evolūcija 2000 - mūsdienas

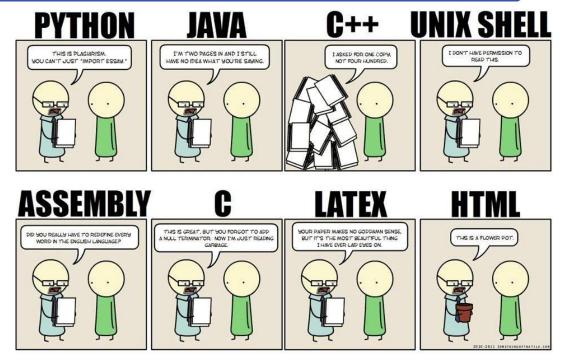


- ■C# (2001) vispārēja MSFT
- Scala (2003) vispārēja
- •Go (2009) vispārēja
- •Rust (2010) sistēmu Mozilla
- •Julia (2012) datu analīze



Alternatīva vēsture





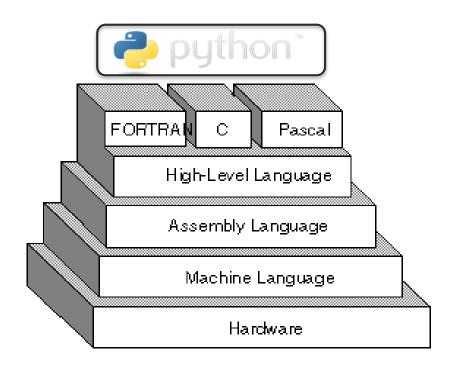
http://james-iry.blogspot.com/2009/05/brief-incomplete-and-mostly-wrong.html

Programmēšanas valodu tipi



- o Compiled language
- o Interpreted language

- o High-level
- o "Low" level



Guido van Rossum – Python izveidotājs



- Programmētājs no Nīdērlandes (Amsterdamas Universitāte)
- Python labvēlīgais diktātors uz mūžu(BDFL)
- •1989.g sāk veidot jaunu valodu ar uzsvaru uz lasāmību
- ■2005-2012 strādā Google
- ■2013 Dropbox







Python filozofija – baterijas iekļautas!





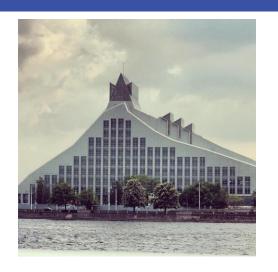
Standarta biblioteka risinājumiem

Python - bibliotēkas













Daudz papildus Python bibliotekas – tīmeklim, matemātikai, grafiskām vidēm un citiem pielietojumiem.

Python filozofija



The Zen of Python

- Beautiful is better than ugly.
- •Explicit is better than implicit.
- •Simple is better than complex.
- •Complex is better than complicated.
- •Readability counts.

(Pavisam 20 aforismi https://github.com/python/peps/blob/master/pep-0020.txt)

Ko Tu spēsi kursa beigās?



- 1. Patstāvīgi izveidot un atkļūdot nelielas Python programmas
- 2. Lasīt citu programmas, kas atbilst vispārejiem koda standartiem
- 3. Mācēsi atrast palīdzību veidojot lielākas Python programmas
- 4. Pārzināsi programmēšanas palīgrīku pamatus(version control etc)

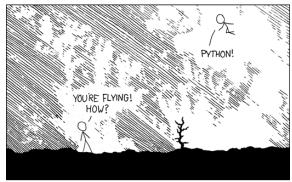
Ko mēs mācīsim?

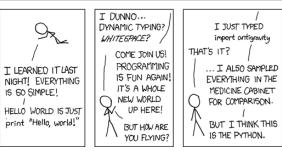


- 1. Nedēļa programmēšanas pamati, ievads Python valodā
- 2. Python izmantošana ikdienas darbu veikšanā
- 3. Python datu apstrādē un vizualizācijā(ievads data science)
- 4. Tīmekļa lapu izveide un uzturēšana izmantojot Python Django framework

Kādēļ Tu esi šeit?







https://xkcd.com/353/

Paldies par uzmanību!



Jautājumi?:)