

# Principles of the \*x-type systems

Saulius Gražulis

Vilnius, 2020

Vilnius University, Faculty of Mathematic and Informatics  
Institute of Informatics



This set of slides may be copied and used as specified in the  
[Attribution-ShareAlike 4.0 International](https://creativecommons.org/licenses/by-sa/4.0/) license

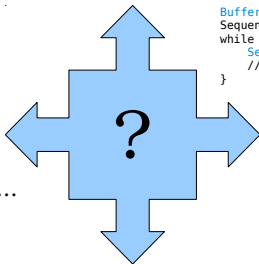


```
my $in = Bio::SeqIO->new(-format => 'fasta', -file => $file);
```

<http://www.bioperl.org>

/

FORTH, Lua  
Scheme, Lisp,  
Sasl, Haskell,  
Ada, Smalltalk ...



```
BufferedReader br = new BufferedReader(new FileReader(fileName));  
SequenceIterator stream = SeqIOTools.readFastaDNA(br);  
while (stream.hasNext()) {  
    Sequence seq = stream.nextSequence();  
    // do something with the sequence.  
}
```

<http://biojava.org/>

```
record = SeqIO.read("NC_005816.fna", "fasta")
```

<http://biopython.org/>

# The Unix operating system

- Kūrėjai:
  - Ken Thompson (Kenneth Lane Thompson), Dennis Ritchie, Brian Kernigan, Doug McIlroy, Joe Ossanna
- 1969 m. Bell Labs
- Alternatyva Multics sistemai
- „Laisvalaikio projektas“?
  - *Also during 1969, Thompson developed the game of 'Space Travel.' /.../ It did not take long, therefore, for Thompson to find a little-used PDP-7 computer with an excellent display processor ...*

# The acknowledgement of Unix

- 1983, Ritchie and Thompson received the Turing Award *for their development of generic operating systems theory and specifically for the implementation of the UNIX operating system*
- **2011**, Ritchie, along with Thompson, *was awarded the Japan Prize for Information and Communications* **for his work in the development of the Unix operating system**

# The principles of Unix

- Daug mažų, nepriklausomų, bet tinkamai sąveikaujančių programų
- Viena programa turi daryti vieną darbą, bet gerai
- Nedidelis branduolys, iš kurio išmesta viskas, ką galima išmesti
- Duomenys teksto pavidale
- Viskas yra failas
- Kuo mažiau, tuo geriau (less is more)

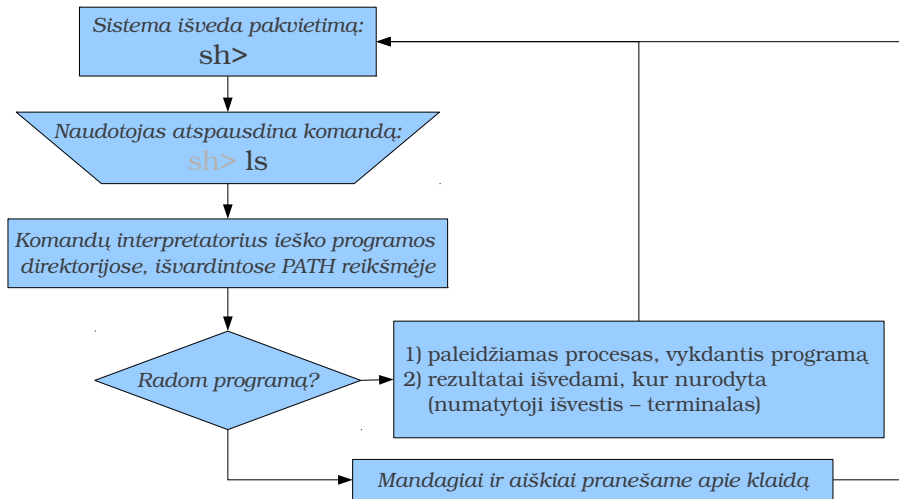
# Outcomes of this policy

- Bet kurią sistemos dalį nesunku pakeisti
- Bet kurią sistemos dalį nesunku suprasti
- Palygint nesunku suprasti sąveikas tarp sistemos dalių
- Sistemą lengva išplėsti ir pritaikyti naujiems uždaviniams spręsti
- *Taigi, hakerių ir mokslininkų rojus... :)*

# Working with a \*x type system

- login – sesija – logout.
- apvalkalėlis, arba komandų interpretatorius (shell)
- pakvietimas (prompt)
- aplinka: aplinkos kintamieji, sistemos komandos
- **komanda** papildomi argumentai  
**duomenu\_failai.txt** > **rezultatas.txt**
- procesai ir jų valdymas, failai

# The command interpreter





# Anatomy of a Unix command

- `sh$ progname input1.dat input2.dat > result.dat`
- `sh$ progname -a -b 1 -cd input1.dat input2.dat > result.dat`
- `sh$ progname -a -b 1 -cd -o result.dat input1.dat input2.dat`
- `sh$ progname --alpha --beta=1 --change-directory input1.dat input2.dat --output result.dat`

# Functional analogy

- `result := progame( input1, input2 );`
  - K: kada ši analogija bus pilna/teisinga?
- Pavyzdys:

```
result1 := progame( input1, input2 );
result2 := progame( input1, input2 );
if( result1 == result2 ) {
    print( "Foo" )
} else {
    print( "Bar" )
}
```

# User friendly?

- Unix **is** user friendly. It's just picky about who his friends are...
- So is Linux ...
- „*What is this horrible fascination with Unix? The operating system of the 1960s, still gaining in popularity in the 1990s. A horrible system, except that all the other commercial offerings are even worse.*“