



Přednáška 2

Úvod do jazyka Python

Algoritmizace (podmínky, cykly), čtení a zápis souborů

155GIT1 Informatika 1, letní semestr 2020-2021

Podmínky

Cykly

Čtení a zápis

Martin Landa

`martin.landa@fsv.cvut.cz`

Fakulta stavební ČVUT v Praze
Katedra geomatiky

<http://geo.fsv.cvut.cz/gwiki/155GIT1>



[Podmínky](#)

[Cykly](#)

[Čtení a zápis](#)

Copyright © 2019-2021 Martin Landa, Ondřej Pešek

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation Licence, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.



Podmínky

Cykly

Čtení a zápis

1 Podmínky

2 Cykly

3 Čtení a zápis



Podmínky

Cykly

Čtení a zápis

```
[In ] prvocisla = [2, 3, 5, 7, 11]
[In ] if len(prvocisla) > 3:
[In ]     prvocisla = prvocisla[:3]
[Out] [2, 3, 5]
```

```
[In ] if len(prvocisla) > 3:
[In ]     prvocisla = prvocisla[:3]
[In ] else:
[In ]     prvocisla.append(7)
[Out] [2, 3, 5, 7]
```

```
[In ] if len(prvocisla) == 0:
[In ]     prvocisla.append(0)
[In ] elif len(prvocisla) > 1:
[In ]     prvocisla.pop()
[In ] else:
[In ]     prvocisla = []
[Out] [2, 3, 5]
[Out] ?
```



```
[In ] promenna = True
[In ] if promenna:
[In ]     print('pravda')
[In ] else:
[In ]     print('nepravda')
[Out] pravda
```

```
[In ] promenna = 1
[Out] pravda
[In ] promenna = 0
[Out] nepravda
[In ] promenna = ''
[Out] nepravda
[In ] promenna = [0, 1, 2]
[Out] pravda
[In ] promenna = []
[Out] ?
```



```
[In ] promenna = True
[In ] if not promenna:
[In ]     print('nepravda')
[In ] else:
[In ]     print('pravda')
[Out] nepravda
```

```
[In ] promenna = 0
[In ] if not promenna:
[In ]     print('nepravda')
[In ] else:
[In ]     print('pravda')
[Out] ?
```



```
[In ] if promenna:  
[In ]     print('pravda')  
[In ] elif promenna is None:  
[In ]     print('nic')  
[In ] else:  
[In ]     print('nepravda')
```

```
[In ] promenna = 10  
[Out] pravda  
[In ] promenna = 0  
[Out] nepravda  
[In ] promenna = None  
[Out] nic
```

```
[In ] all([True, True, True])
```

```
[Out] True
```

```
[In ] all([True, False, True])
```

```
[Out] False
```

```
[In ] any([True, False, True])
```

```
[Out] True
```

```
[In ] all([0, 1, 2])
```

```
[Out] False
```

```
[In ] any([0, 1, 2])
```

```
[Out] True
```

```
[In ] all(['a', ''])
```

```
[Out] False
```

```
[In ] any(['a', ''])
```

```
[Out] True
```





Podmínky

Cykly

Čtení a zápis

1 Podmínky

2 Cykly

3 Čtení a zápis



```
[In ] prvocisla = [2, 3, 5, 7, 11]
```

```
[In ] # prochazeni pomoci indexu
```

```
[In ] for i in range(len(prvocisla)):
```

```
[In ]     print(prvocisla[i])
```

```
[In ] # prochazeni prvku
```

```
[In ] for pc in prvocisla:
```

```
[In ]     print(pc)
```

```
[Out] 2
```

```
[Out] 3
```

```
[Out] 5
```

```
[Out] 7
```

```
[Out] 11
```



```
[In ] prvocisla = [2, 3, 5, 7, 11]
[In ] while len(prvocisla) > 3:
[In ]     prvocisla.pop()
```

```
[Out] [2, 3, 5]
```

„Nekonečný cyklus“

```
[In ] prvocisla = [2, 3, 5, 7, 11]
[In ] while True:
[In ]     prvocisla.pop()
[Out] ?
```



Podmínky

Cykly

Čtení a zápis

```
[In ] prvocisla = [2, 3, 5, 7, 11]
[In ] for p in prvocisla:
[In ]     if p > 5:
[In ]         break
[In ]     print(p)
[Out] [2, 3, 5]
```

```
[In ] for p in prvocisla:
[In ]     if p % 2 == 0:
[In ]         continue
[In ]     print(p)
[Out] [3, 5, 7, 11]
```



Podmínky

Cykly

Čtení a zápis

1 Podmínky

2 Cykly

3 Čtení a zápis



Funkce `open()` & `read()`

```
[In ] fd = open('soubor.txt')  
[In ] data = fd.read()  
[In ] fd.close()
```

Podmínky

Cykly

Čtení a zápis

Alternativní zápis `with`

```
[In ] with open('soubor.txt') as fd:  
[In ]     data = fd.read()
```

```
[In ] type(data)  
[Out] <class 'str'>
```

```
[In ] print(len(data))  
[Out] 10594
```

```
[In ] print(len(data.encode()))  
[Out] 10788
```



Podmínky

Cykly

Čtení a zápis

```
[In ] with open('soubor.txt', 'rb') as fd:
[In ]     data = fd.read()
[In ]     type(data)
[In ]     print(len(data))
```

```
[Out] <class 'bytes'>
[Out] 10788
```

Přístup k bajtům, převod na znaky

```
[In ] data[0]
[In ] type(data[0])
[In ] chr(data[0])

[Out ] 49
[Out ] <class 'int'>
[Out ] '1'
```

<https://cs.wikipedia.org/wiki/ASCII>



Podmínky

Cykly

Čtení a zápis

Načtení obsahu do paměti

```
[In ] with open('soubor.txt') as fd:  
[In ]     data = fd.read()  
[In ]     for radek in data.splitlines():  
[In ]         print(len(radek))
```

```
[Out] 28  
[Out] 39  
[Out] ...
```

Čtení řádků ze souboru

```
[In ] with open('soubor.txt') as fd:  
[In ]     for radek in fd.readlines():  
[In ]         print(len(radek))
```

```
[Out] 29  
[Out] 40  
[Out] ...
```



Režim `write`

```
[In ] with open('testament.txt', 'w') as fd:  
[In ]     fd.write('Miluji')
```

Pokud soubor existuje, tak bude přepsán!

Režim `append`

```
[In ] with open('testament.txt', 'a') as fd:  
[In ]     fd.write(' Python' + '\n')
```

Kontrola

```
[In ] with open('testament.txt') as fd:  
[In ]     print(fd.read())
```

```
[Out] Miluji Python  
[Out]
```



Podmínky

Cykly

Čtení a zápis

Víceřádkový řetězec

```
[In ] with open('testament.txt', 'w') as fd:  
[In ]     fd.write('''Ach, kdybych já byl studoval  
[In ] v svém mládí bláznivém, můj Bože,  
[In ] a na dobré se mravy dal,  
[In ] teď měl bych dům a měkké lože.''')
```

https://en.wikipedia.org/wiki/Le_Testament

Kontrola

```
[In ] with open('testament.txt') as fd:  
[In ]     print(fd.read())  
  
[Out] Ach, kdybych já byl studoval  
[Out] v svém mládí bláznivém, můj Bože,  
[Out] a na dobré se mravy dal,  
[Out] teď měl bych dům a měkké lože.
```
