

## Wykład 1

- Informacje o przedmiocie
- Zakres przedmiotu
- Literatura
- Pojęcia podstawowe
- Język Python

1

## Języki i biblioteki analizy danych

**Rok akademicki:** 2022/2023

**Liczba godzin:** Semestr 1, wyk. 30 godz., lab. 30 godz., proj. 15 godz.

**Wykład:** Marek Gajęcki

**Laboratorium:** Kaleta Zbigniew

**Projekt:** Marek Gajęcki

**Cel wykładu:** Celem przedmiotu jest zapoznanie studentów z językiem programowania Python oraz podstawowymi bibliotekami przydatnymi przy analizie danych.

**Slajdy i inne materiały:**  
<https://upel.agh.edu.pl>

2

## Zakres przedmiotu

- Pojęcia podstawowe
- Wprowadzenie do języka Python
- Wybrane mechanizmy języka Python
- Wybrane biblioteki języka Python

3

## Literatura

- D. Harel „Rzecz o istocie informatyki - algorytmika”
- J.G. Brookshear „Informatyka w ogólnym zarysie”
- J. Mieścicki „Wstęp do informatyki nie tylko dla informatyków”
- T.H. Cormen „Wprowadzenie do algorytmów”
- M. Lutz „Python. Wprowadzenie”
- M. Gągolewski „Przetwarzanie i analiza danych w języku Python”
- A. Boschetti „Python Podstawy nauki o danych”



4

## Dane dzisiaj

1024 Bajty = 1 Kilobajt  
1024 Kilobajty = 1 Megabajt  
1024 Megabajty = 1 Gigabajt  
1024 Gigabajty = 1 Terabajt  
1024 Terabajty = 1 Petabajt  
1024 Petabajty = 1 Eksabajt  
1024 Exabajty = 1 Zettabajt  
1024 Zettabajty = 1 Jottabajt

Pomiędzy zaraniem cywilizacji, a rokiem 2003 zostało stworzonych 5 Exabajtów informacji, dziś tworzymy tyle danych w 2 dni.

Prezes Google, Eric Schmidt, 2010 Konferencja Google Atmosphere

5

## Cała wiedza świata

- W 2000 r. tylko 25 proc. wszystkich danych zebranych na świecie była w **formie cyfrowej** (reszta na papierze, taśmie filmowej, płytach winylowych, kasetach itp.).
- W 2013 r. w cyfrowej formie zapisane zostało 98 proc. wszystkich danych zgromadzonych na świecie – 1200 eksabajtów (EB).
- Rok 2020 – każdy z nas wytwarza 1.7 MB/s, łączna liczba danych to 45 zettabajtów.
- IBM szacuje, że 90% istniejących danych zostało wytworzone w ostatnich 2 latach.

6

## Podstawowe pojęcia

**Zadanie algorytmiczne** – polega na określeniu:

- wszystkich poprawnych danych wejściowych
- oczekiwanych wyników jako funkcji danych wejściowych

**Algorytm** - specyfikacja ciągu elementarnych operacji, które przekształcają dane wejściowe na wyniki.

Algorytm można przedstawić w postaci:

- werbalnej (*opis słowny*)
- symbolicznej (*schemat blokowy*)
- programu

7

## Przykład zapisu algorytmu

**Problem:** równanie kwadratowe

**Dane:** współczynniki **a, b, c**

**Wyjście:** pierwiastki **x1, x2** albo informacja o ich braku

**Postać werbalna** algorytmu:

„Mając dane współczynniki a, b, c :

oblicz  $d = b^2 - 4ac$

Jeżeli d jest nieujemne :

oblicz  $p = \sqrt{d}$

oblicz  $x1 = (-b-p)/(2a)$

oblicz  $x2 = (-b+p)/(2a)$

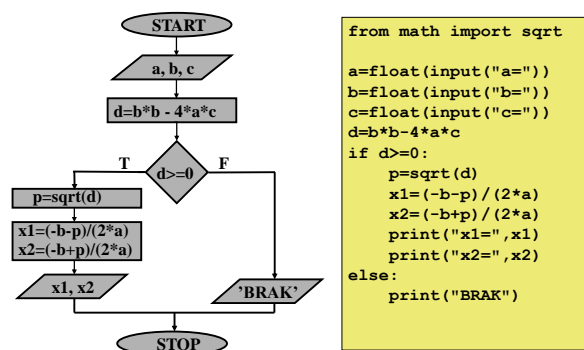
wypisz wartości x1, x2

Jeżeli d jest ujemne :

wypisz "BRAK PIERWIASTKÓW"

8

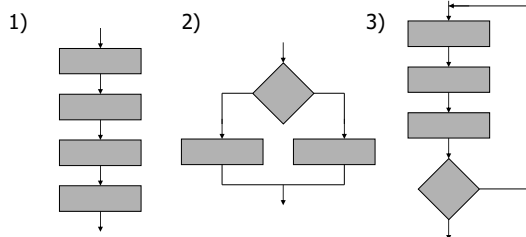
## Zapis symboliczny a program



9

## Budowa algorytmów

- Bezpośrednie następstwo
- Wybór warunkowy
- Iteracja warunkowa



10

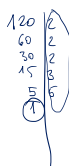
## Przykład zadania

**Problem:** Rozkład liczby na czynniki pierwsze

Proszę napisać program, który dla wczytanej liczby naturalnej wypisuje jej rozkład na czynniki pierwsze.

**Przykład:**

120 : 2, 2, 2, 3, 5



11

## Rozkład na czynniki pierwsze

**Rozwiązanie pierwsze**

```
n = int(input("n="))
b = 2
while n > 1:
    if n % b == 0:
        print(b)
        n = n // b
    else:
        b = b + 1
    # end if
# end while
print("koniec")
```

*C/C++*  
*a/b*

$13 \% 5 = 3$

$13 : 5 = 2 \text{ v } 3$

$13 \% 2 = 1$

$13 / 5 = 26$

end = None

while a > p:

a = a - 1

print(a)

print("koniec")

if x == y:  
x = x + 1  
y = y - 1  
end

12

# Położenie podzielników liczby

$121 = 11 \cdot 11$

Podzielniki liczby 120

13

# Rozkład na czynniki pierwsze

## Rozwiązanie drugie

```
from math import sqrt

n = int(input("n="))
b = 2
while b<=sqrt(n):
    if n % b == 0:
        print(b)
        n = n // b
    else:
        b = b+1
# end if
# end while
if n>1: print(n)
print("koniec")
```

14

# Porównanie rozwiązań

Liczba cyfr	Algorytm pierwszy	Algorytm drugi
6	0.07s	0.0s
7	0.58s	0.0s
8	7.75s	0.0s
9	1m7.2s	0.01s
10	9m43s	0.01s
11	1h23m	0.04s
12	12h27m	0.13s
13	4d9h	0.42s
14	37d12h	1.23s

Czy można jeszcze szybciej?

Handwritten notes include RSA calculations:  $n = 100$ ,  $n! = 1000$ ,  $\sqrt{n!} \approx 31.6$ , and a calculation for 5004f leading to  $10^{10}$ .

15

# Język Python

- Stworzony przez holendra Guido van Rossum w 1991 roku
- Open Source
- Interpretowany, interaktywny, zorientowany obiektowo
- Przenośny
- Bogata biblioteka funkcji
- Dobra dokumentacja
- Łatwy do nauki, przejrzysty w zapisie
- Funkcje, moduły, klasy, pakiety
- Dynamiczne typowanie
- Automatyczne zarządzanie pamięcią
- Obsługa wyjątków
- Struktury wysokiego poziomu: zbiory, krotki, listy, słowniki

16

# Język Python

Gdzie można używać:

- szybkie prototypowanie
- programowanie sieciowe (po stronie Klienta i serwera)
- programowanie ad hoc („skryptowanie”)
- w aplikacjach naukowych
- język rozszerzeń
- przetwarzanie XML
- aplikacje bazodanowe
- aplikacje GUI
- edukacja

Kto używa języka Python?

- Google (various projects)
- NASA (several projects)
- NYSE (one of only three languages "on the floor")
- Industrial Light & Magic (everything)
- Yahoo! (Yahoo mail & groups)
- i inni

17

# Popularność języków programowania

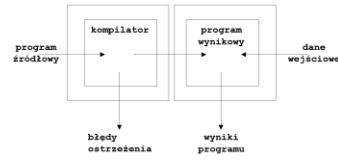
Sep 2022	Sep 2021	Change	Programming Language	Rating	Change
1	2	▲	Python	15.74%	+4.07%
2	1	▼	C	13.96%	+2.13%
3	3		Java	11.72%	+0.60%
4	4		C++	9.76%	+2.63%
5	5		C#	4.88%	-0.89%
6	6		Visual Basic	4.39%	-0.22%
7	7		JavaScript	2.82%	+0.27%
8	8		Assembly language	2.49%	+0.07%
9	10	▲	SQL	2.07%	+0.21%
10	9	▼	PHP	1.68%	-0.17%
11	24	▲	Objective-C	1.49%	+0.86%
12	14	▲	Go	1.16%	+0.02%
13	20	▲	Delphi/Object Pascal	1.09%	+0.32%
14	16	▲	MATLAB	1.06%	+0.04%
15	17	▲	Fortran	1.03%	+0.02%

www.tiobe.com

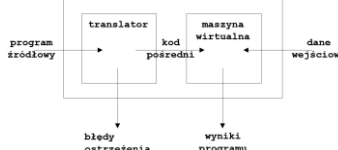
18

# Wykonanie programu

## Język C++



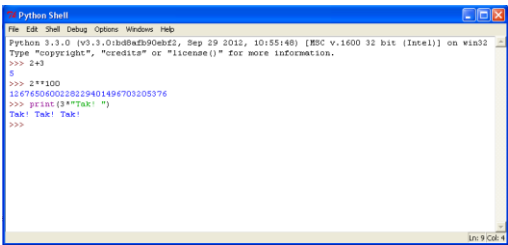
## Język Python



19

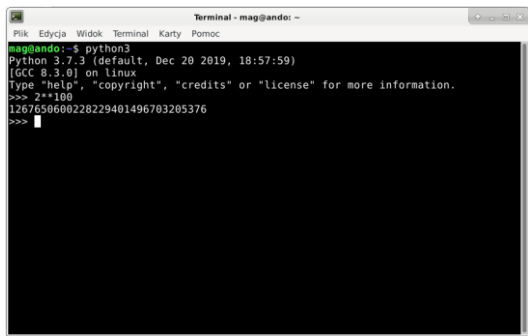
# Python w systemie Windows

<http://www.python.org>



20

# Python w systemie Linux



21

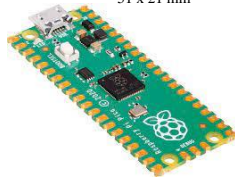
# Python na telewizorze



22

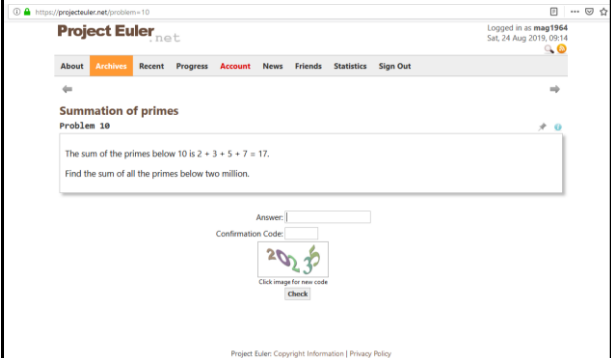
# Micro Python

Układ	Raspberry Pi RP2040
Rdzeń	ARM Cortex-M0+ Dual-Core 133 MHz
Pamięć SRAM	264 kB
Pamięć Flash	2 MB
Wyprowadzenia	40-pin / 23 GPIO cyfrowe + 3 piny ADC
Interfejsy	2x UART, 2x I2C, 2x SPI, do 16 kanałów PWM
Napięcie zasilania	5 V (USB)
Wymiary	51 x 21 mm



23

# Project Euler



24