

Kolos

1 Kolokwium

- Link: <https://cogsys.io/kolos-gr01.html>
- Czas: 45 minut
- Dozwolone:
 - kartka,
 - długopis,
 - <https://www.python.org/>
 - <https://docs.python.org/3/tutorial/introduction.html>
- Niedozwolone:
 - ChatGPT
 - kolega/koleżanka z pracowni
 - komunikacja z osobami z zewnątrz

2 Trójkąt

2.1 Zadanie

```
# Sprawdź czy z odcinków a, b, c można zbudować trójkąt  
# a jeżeli tak to czy będzie on trójkątem prostokątnym.  
a = 5  
b = 13  
c = 1  
c = 11  
c = 12  
# sprawdź czy program zadziała dla innych wartości c
```

3 Prosty wzór

3.1 Zadanie

```
# Za pomocą pętli for narysuj wzór z kropek
nn = 5
# .
# ..
# ...
# ....
# .....
```

4 Letters

4.1 Zadanie

```
# zmień poniższy kod żeby zamiast
# index: 0 element: a | a
# index: 1 element: b | bb
# index: 2 element: c | ccc
# index: 3 element: d | dddd
# index: 4 element: e | eeeee
# index: 5 element: f | fffffff
# index: 6 element: g | ggggggg
# index: 7 element: h | hhhhhhhh
#
# rysować
#
# index: 0 element: h | h
# index: 1 element: g | gg
# index: 2 element: f | fff
# index: 3 element: e | eeee
# index: 4 element: d | ddddd
# index: 5 element: c | cccccc
# index: 6 element: b | bbbbbb
# index: 7 element: a | aaaaaaa

from string import ascii_lowercase as alpha
total = 8
letters = alpha[:total]
for idx, letter in enumerate(letters):
    print(
        f"index: {idx} element: {letter} | " +
        f"{letter * (idx + 1)}")

print("OK")
```

```
index: 0 element: a | a
index: 1 element: b | bb
index: 2 element: c | ccc
index: 3 element: d | dddd
index: 4 element: e | eeeee
```

```

index: 5 element: f | ffffff
index: 6 element: g | gggggggg
index: 7 element: h | hhhhhhhh
OK

```

5 Pattern

5.1 Zadanie

```

# Rozwiń poniższy kod, żeby narysować
# index: 0 letter: a |      a      a      a      a      a
# index: 1 letter: b |      bbb     bbb     bbb     bbb     bbb
# index: 2 letter: c |      cccc    cccc    cccc    cccc    cccc
# index: 3 letter: d |      ddddddd ddddddd ddddddd ddddddd ddddddd
# index: 4 letter: e |      eeeeeeee eeeeeeee eeeeeeee eeeeeeee eeeeeeee
# index: 0 letter: a |      aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
# index: 1 letter: b |      bbbbbbbb bbbbbbbb bbbbbbbb bbbbbbbb bbbbbbbb
# index: 2 letter: c |      ccccc    ccccc    ccccc    ccccc    ccccc
# index: 3 letter: d |      ddd      ddd      ddd      ddd      ddd
# index: 4 letter: e |      e        e        e        e        e
# index: 0 letter: a |      a        a        a        a        a
# index: 1 letter: b |      bbb      bbb      bbb      bbb      bbb
# index: 2 letter: c |      ccccc    ccccc    ccccc    ccccc    ccccc
# index: 3 letter: d |      ddddddd ddddddd ddddddd ddddddd ddddddd
# index: 4 letter: e |      eeeeeeee eeeeeeee eeeeeeee eeeeeeee eeeeeeee
# index: 0 letter: a |      aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
# index: 1 letter: b |      bbbbbbbb bbbbbbbb bbbbbbbb bbbbbbbb bbbbbbbb
# index: 2 letter: c |      ccccc    ccccc    ccccc    ccccc    ccccc
# index: 3 letter: d |      ddd      ddd      ddd      ddd      ddd
# index: 4 letter: e |      e        e        e        e        e
from string import ascii_lowercase as alpha
total = 8
letters = alpha[:total]
for idx, letter in enumerate(letters):
    print(
        f"index: {idx} element: {letter} |" +
        f" {letter * (idx + 1)} ")

```

```

index: 0 element: a | a
index: 1 element: b | bb
index: 2 element: c | ccc
index: 3 element: d | dddd
index: 4 element: e | eeeee
index: 5 element: f | ffffff
index: 6 element: g | gggggggg
index: 7 element: h | hhhhhhhh

```