

Test wiedzy w wersji Python

XVI OIJ, zawody I stopnia, tura testowa
6 października 2021



1. Która instrukcja służy do sprawdzenia warunku i (jednokrotnego) wykonania instrukcji zależnie od tego czy ten warunek jest spełniony?

- condition
- if
- check
- true
- while

2. W jaki sposób zapisać w zmiennej x długość napisu zapisanego w zmiennej y typu `str`?

- `x == y->length`
- `x <- size(y)`
- `x = len(y)`
- `x := y->size`
- `set(x, size(y))`

3. W jaki sposób obliczyć wynik działania a^3 ?

- `a ^ 3`
- `exp(a, 3)`
- `a * a * a`
- `a * 3`

4. Rozważmy następujący fragment programu:

```
elems = ...
...
p = 0
for i in range(1, len(elems)):
    if elems[i] > elems[p]:
        p = i
```

Co zawiera zmienna p po wykonaniu powyższego kodu?

- wartość największego elementu z `elems`
- liczbę elementów `elems`
- liczbę elementów `elems` większych niż `elems[0]`
- pozycję pierwszego największego elementu z `elems`

5. Które z poniższych fragmentów obliczają 33 procent liczby zapisanej w zmiennej x typu `float`?

- `x / 3`
- `x * 0.33`
- `x % 33`
- `x / 100 * 33`
- `33 % x`
- `x * 33%`

6. Rozważmy poniższą funkcję:

```
def oblicz(n):
    wynik = 0
    for i in range(1, n + 1):
        wynik += n // i
    return wynik
```

Jaki wynik zwróci wywołanie `oblicz(25)`?

7. Ile gwiazdek wypisze poniższy fragment programu?

```
for i in range(6):
    for j in range(6):
        print('*', end='')
```

8. Rozważmy poniższą funkcję:

```
def funkcja(napis):
    wynik = ''
    for i in range(1, len(napis), 2):
        wynik += napis[i]
    return wynik
```

Jaki będzie wynik wywołania `funkcja('olimpiada')`?



9. Rozważmy poniższą funkcję:

```
def funkcja(a, b):
    wynik = 0
    while b > 0:
        wynik += a
        b -= 1
    return wynik
```

Zakładając, że zmienne a oraz b są całkowite, dodatnie i nie przekraczają 100, który z poniższych kodów jest równoważny wywołaniu `funkcja(a, b)`?

- `a // b`
- `a - b`
- `a + b`
- `a * b`

10. Ile równa jest najmniejsza wspólna wielokrotność liczb 1,2,3,...,10?

11. Rozważmy poniższą funkcję:

```
def rad(n):
    wynik = 1
    i = 2
    while i * i <= n:
        if n % i == 0:
            wynik *= i
            while n % i == 0:
                n //= i
        i += 1
    if n > 1:
        wynik *= n
    return wynik
```

Ile jest liczb naturalnych n mniejszych niż 255, dla których `rad(n) == 10`?

12. Rozważmy poniższą funkcję:

```
def js(n):
    wartosc = 1
    i = 1
    while True:
        wartosc *= i
        if wartosc > n:
            return i - 1
        i += 1
```

Jaka jest największa wartość parametru n , dla której `js(n) == 4`?

13. Rozważmy poniższą funkcję:

```
def mx(tab, x):
    wynik = 0
    for y in tab:
        wynik = max(wynik, x ^ y)
    return wynik
```

Niech `tab = [11, 10, 24, 26]`. Jaką najmniejszą wartość może zwrócić wywołanie `mx(tab, x)` dla odpowiednio dobranej nieujemnej całkowitej wartości x ?

14. Rozważmy poniższą funkcję:

```
def ws(tab, p, x):
    if p >= len(tab):
        return False
    if tab[p] == x:
        return True
    return ws(tab, p + 1, x)
```

Dla jakich zawartości `tab` wywołanie `ws(tab, 2, 7)` zwróci `True`?

- `[3, 7, 1, 7, 5, 2]`
- `[4, 9, 4, 1]`
- `[1, 2, 7]`
- `[5, 8, 0, 2, 7]`
- `[2]`
- `[7, 1, 2, 3, 4]`

15. Celem poniższej funkcji jest zmodyfikować `tab`, aby odwrócić kolejność jej elementów. Jak można uzupełnić warunek w ..., aby to osiągnąć?

```
def odwracaj(tab):
    n = len(tab)
    i = 0
    while ...:
        tab[i], tab[n - 1 - i] = tab[n - 1 - i], tab[i]
        i += 1
```

- `i < n`
- `i < n // 2`
- `i <= n // 2`
- `i < n - 1 - i`
- `i <= n`

16. Ile zer ma liczba $20^{50} \cdot 50^{20} \cdot 90^{90}$ na końcu zapisu dziesiętnego?

17. Poprawnym nawiasowaniem nazywamy napis, który może powstać z wyrażenia arytmetycznego przez opuszczenie wszystkiego poza znakami nawiasów. Na przykład napis $()(())$ jest poprawnym nawiasowaniem, mógł powstać na przykład z wyrażenia $(2 + 2) * (1 + (2 + 3) * 4)$. Celem poniższej funkcji jest zbadać czy przekazany napis jest poprawnym nawiasowaniem.

```
def czy_poprawne_nawiasowanie(naw):
    min_balans = 0
    balans = 0
    for x in naw:
        if x == '(':
            balans += 1
        elif x == ')':
            balans -= 1
        else:
            return False
    min_balans = min(min_balans, balans)
    return ...
```

Jak należy uzupełnić ..., aby to osiągnąć?

- balans == 0
 - (min_balans < balans) and (balans >= 0)
 - min_balans == 0
 - (min_balans >= 0) and (balans == 0)
 - min_balans <= balans
18. Jaka jest najmniejsza liczba naturalna n , której zapis w systemie rzymskim ma dokładnie 10 znaków? Odpowiedź podaj w systemie dziesiętkowym.
-
19. Ile zmiennych typu `str` przechowujących napisy długości 1000000 znaków (ASCII) można zadeklarować w programie, aby zużycie pamięci (związane z przechowywaniem tych zmiennych) wynosiło około 50 MB?
- około pięćset
 - około pięciu
 - kilka tysięcy
 - około pięćdziesiąt
 - jedną

20. Rozważmy poniższy fragment programu:

```
for i in range(1, n + 1):
    j = 1
    while j * j <= n:
        print('* ', end='')
        j += 1
```

Jaką liczbę gwiazdek wypisaną przez program można osiągnąć, odpowiednio dobierając n ?

- 400
 - 512
 - 350
 - 256
 - 150
21. Rozważmy poniższą funkcję:

```
def wykonuj(n):
    for i in range(1, n + 1):
        j = i
        while j % 2 == 0:
            j //= 2
        print('* ', end='')
```

Zakładamy, że operacje arytmetyczne oraz porównania na zmiennych typu `int` zajmują czas stały. Jaka jest pesymistyczna złożoność obliczeniowa funkcji `wykonuj`?

- $\Theta(n \log n)$
 - $\Theta(n)$
 - $\Theta(n^2)$
 - $\Theta(\sqrt{n})$
 - $\Theta(\log n)$
 - $\Theta(n\sqrt{n})$
22. Które z poniższych zbiorów liczb można podzielić na dwa zbiory o równej sumie? Każdy element powinien trafić do dokładnie jednego zbioru.
- {2, 3, 4, 5, 6, 7, 8}
 - {1, 2, 3, 5, 8, 13}
 - {1, 2, 3, 4, 5, 6, 7}
 - {1, 2, 4, 8, 16, 32, 65}
 - {2, 3, 5, 7, 11}
23. Która z podanych liczb jest największa?
- FF_{16}
 - 200
 - 1111111_2
 - $A0_{11}$
24. Ile jest całkowitych nieujemnych kwot, których nie można wydać nominałami 7 oraz 11?
-

25. Dla jakich wartości parametru n poniższa funkcja poprawnie kończy swoje działanie (tzn. nie powoduje błędu wykonania)?

```
def wykonuj(n):
    if n == 100:
        return
    wykonuj(n + 2)
```

- 1000
- 10
- 100
- 1

26. Ile list rozmiaru 1 000 000 składających się z liczb naturalnych można posortować z użyciem funkcji `sorted` na przeciętnym komputerze z roku 2020, aby czas wykonania był równy około dwóch sekund? Zakładamy, że sortowania wykonywane są jedno po drugim bez współbieżności.

- kilka tysięcy
- nie można wykonać nawet jednego
- kilka
- kilkaset

27. 60 dzieci ustawiło się w kółku i zaczęło odliczać do dwóch (tzn. pierwsze dziecko mówi 1, drugie dziecko mówi 2, trzecie 1, czwarte 2, piąte 1 itd.). Każde dziecko, które mówi 2 natychmiast wypada z kółka, a wyliczanka jest dalej kontynuowana. Po pełnym okrążeniu przeprowadzana jest analogiczna wyliczanka tylko do trzech (tzn. dzieci liczą 1, 2, 3 i każde dziecko, które mówi 3 wypada). W kolejnych okrążeniach następują analogiczne wyliczanki, ale do 4, 5, 6, ... W pewnym momencie wyliczanka jest do większej liczby niż liczba dzieci i nikt już dalej nie odpadnie. Ile dzieci pozostanie w kółku?

28. Rozważmy następującą funkcję:

```
def oblicz(operacje):
    wynik = 0
    for x in operacje:
        if x == '+':
            wynik += 1
        if x == '*':
            wynik *= 2
    return wynik
```

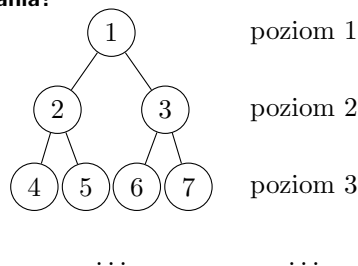
Podaj napis złożony z czterech znaków `+` oraz czterech znaków `*`, który należy przekazać jako parametr `operacje`, aby `oblicz(operacje) == 35`?

29. Rozważmy poniższą funkcję:

```
def oblicz(n, k):
    wyniki = [[0] * (k + 1) for __ in range(n + 1)]
    for i in range(n + 1):
        wyniki[i][0] = 1
        for j in range(1, min(k, i) + 1):
            wyniki[i][j] += wyniki[i - 1][j]
            wyniki[i][j] += wyniki[i - 1][j - 1]
    return wyniki[n][k]
```

Jaki jest wynik wywołania `oblicz(10, 5)`?

30. Na poniższym rysunku wierzchołki są na trzech poziomach, każdy wierzchołek na poziomie innym niż ostatni jest połączony z dwoma na poziomie kolejnym. Wierzchołki są numerowane od 1 kolejno poziomami, a na każdym poziomie od lewej do prawej. Rozważmy podobny rysunek, ale o sześciu poziomach. Ile wynosi największa możliwa do osiągnięcia suma wartości wierzchołków na ciągu kolejno połączonych wierzchołków bez zwracania?



Fragment pełnego drzewa binarnego (do trzech poziomów).

