

Pytania dla języka Python

XIV OIJ, zawody I stopnia, tura testowa
16 września 2019 – 13 stycznia 2020



Poniżej znajdują się pytania testowe z zawodów I stopnia XIV Olimpiady Informatycznej Juniorów (oij.edu.pl) – na teście wiedzy (do rozwiązania pod adresem oij.edu.pl/sio) trzeba odpowiedzieć na 30 pytań wylosowanych z tej listy oraz listy pytań algorytmicznych. Nie musisz odpowiedzieć na wszystkie pytania, aby zakwalifikować się dalej. Poza testem, do rozwiązania będą jeszcze zadania programistyczne. Przewodnik dla stawiających pierwsze kroki z Olimpiadą można przeczytać na stronie oij.edu.pl/zawodnik/przewodnik/. Serdecznie zapraszamy do startu!

1. W języku Python definicja funkcji nie zwracającej wartości powinna się zaczynać od:

- nazwy funkcji
- def
- nie można deklorować funkcji które nie zwracają wartości
- let

2. Co wypisze kod w języku Python:

```
print('2+2')
```

- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)
- 2+2
- 4
- 22

3. Pod jakim indeksem znajduje się ostatni element listy zadeklarowanej kodem w języku Python:

```
tablica = [1, 2, 3, 4, 5, 6]
```

- 5
- 6
- 7
- deklaracja tablicy jest niepoprawna

4. Do czego służy operator % w języku Python?

- do podzielenia przez siebie dwóch liczb
- do policzenia ile procent jednej liczby stanowi druga
- do podzielenia liczby przez 100
- do policzenia reszty z dzielenia jednej liczby przez drugą

5. Funkcja abc zwraca znak i przyjmuje jako argumenty dwie zmienne a, b typu liczbowego. Pierwszy wiersz jej deklaracji w języku Python brzmi:

- def abc[a, b]:
- abc(a, b) is
- char abc(a, b):
- def abc(a, b):

6. Operator bitowy XOR (alternatywa rozłączna) w Python to:

- %
- |
- ^
- &

7. Jaką wartość ma zmienna słowo zdefiniowana kodem Python:

```
słowo = 5 * "a"
```

- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)
- 5 a
- 5a
- aaaaa

8. Operator potęgowania w Python to:

- **
- @
- ^
- w Pythonie nie ma operatora potęgowania

9. Co wypisze kod w języku Python:

```
print('2' + '2')
```

- 4
- 22
- 2+2
- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)

10. Jak można przejść do następnej linii (wypisać znak końca linii) w języku Python?

- print('')
- print('\n')
- print()
- newline()



11. Mamy daną listę zdefiniowaną kodem w Pythonie:
`t = [2, 4, 8, 16, 32]`; Jaką wartość ma `t[-1]`?

- 0
- zachowanie takiej operacji jest niezdefiniowane
- 2
- 32

12. Mamy daną listę zdefiniowaną kodem w Pythonie:
`t = ['o', 'j', 'a', 'i', 'k']`. Jakie słowo będzie utworzone po złączeniu w jedno słowo wartości z listy `t[0]`, `t[3]`, `t[1]`?

- mai
- oij
- 031
- ika

13. Lista `tab` rozmiaru 3000 jest wypełniona kolejnymi liczbami parzystymi, przy czym `tab[0] = 2`, `tab[1] = 4`, itd. Jaka wartość znajduje się pod indeksem 2019?

- 4038
- 4040
- 2019
- 2

14. Co wypisze poniższy kod Python?

```
x = 'd' - 'a'  
print(x)
```

- 2
- c
- 3
- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)

15. Ile razy wypisze się znak X dla danego kodu Python?

```
for i in range(ord('a'), ord('z')):  
    print('X')
```

- 25
- 30
- 32
- 26

16. Co w języku Python zwróci `f(f(12))` jeśli funkcja `f` jest zadeklarowana w ten sposób:

```
def f(x):  
    return x*x+2*x+3
```

- 29585
- 170
- 29586
- 171

17. Co w języku Python zwróci `f(1)` jeśli funkcja `f` jest zdefiniowana w ten sposób:

```
def f(x):  
    if x == 3: return x * x  
    return f(x+1)
```

- 9
- program zapętlony się (funkcja nigdy nie zwróci żadnej wartości)
- 1
- 3

18. Ile razy wypisze się znak X dla danego kodu Python?

```
for i in range(3, 27, 4):  
    print('X')
```

- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)
- 6
- 0
- 7

19. Co wypisze fragment kodu w języku Python:

```
x = 10  
if x == 5:  
    print(x)  
    print('3')  
else:  
    print(x+1)
```

- 53
- 311
- 11
- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)

20. Co wypisze fragment kodu w języku Python (proszę założyć, że zmienna `x` istnieje i ma wartość liczbową):

```
if x == 5: print(2*x+6)  
else: print('16')
```

- zależy od początkowej wartości zmiennej `x`
- 14
- 5
- 16

21. Które z tych operatorów w języku Python służą do porównywania wartości zmiennych?

- ==
- >=
- <
- =

22. Dla których z podanych wartości zmiennej `x` poniższe wyrażenie w języku Python będzie prawdziwe:

```
(x==7) or ((x<12) and (x>=11))
```

- 11
- 12
- żadna z pozostałych odpowiedzi nie jest poprawna
- 7

23. Które z tych operatorów w języku Python można wstawić w miejsce `#` w wyrażeniu `5#7` by było ono prawdziwe?

- `<=`
- `>=`
- `<`
- `==`

24. Jaką liczbę należy wstawić w miejsce `#` w poniższej pętli w języku Python by wykonała się ona dokładnie 5 razy?

```
for i in range(1, #):  
    pass
```

- 6
- 4
- nie istnieje taka liczba
- 5

25. Mamy daną listę zdefiniowaną kodem w Pythonie:

```
t = ['oij', 'xd', 'abc', 'quiz', 'u']
```

Które z tych wyrażeń zwróci `u`?

- `t[3][1]`
- `t[5]`
- `t[3][2]`
- `t[4]`

26. Co wypisze podany fragment kodu w języku Python:

```
a = 5  
if a == 3: print('prawda')  
else: print('falsz')
```

- `prawda`
- `prawdafalsz`
- `falsz`
- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)

27. Co wypisze ten fragment kodu w języku Python?

```
print('1\n1')
```

- 1 1
- 11
- 1\n1
- 1
- 1

28. Które z podanych fragmentów kodu w Python są poprawnymi sposobami wczytania napisu o długości 3 znaków, do zmiennej `s`?

- `s = 3`
`for i in s:`
 `i = input()`
- `s = '000'`
`for i in range(3):`
 `s[i] = input()`
- `s = []`
`for i in range(3):`
 `s[i] = input()`
- `s = input()`

29. Które z podanych fragmentów kodu w Pythonie są poprawnymi sposobami wypisania zmiennej `s` typu `str`?

- `for c in s:`
 `print(c)`
- `for c in s:`
 `print(c, end=' ')`
- `for c in s:`
 `print(c, end='')`
- `print(s)`

30. Które z tych fragmentów kodu w języku Python wypiszą wszystkie małe, a następnie wielkie litery alfabetu angielskiego od `a` do `z` (`abc[...]` `zABC[...]` `Z`) (w kolejnych liniach)?

- `for c in range(ord('a'), ord('Z')):`
 `print(chr(c))`
- `for c in range(ord('a'), ord('z') + 1):`
 `print(chr(c))`
`for c in range(ord('A'), ord('Z') + 1):`
 `print(chr(c))`
- `for c in range(52):`
 `print(chr(ord('a')+c))`
- `for c in range(26):`
 `print(chr(ord('a')+c))`
`for c in range(26):`
 `print(chr(ord('A')+c))`



31. Co zostanie wypisane po wykonaniu następującego kodu Python?

```
s = 'ab'
s += s
s += 'a'
s += s
print(s)
```

- zachowanie tego kodu jest niezdefiniowane
- aba
- abaaba
- ababaababa

32. Który z tych kodów w Pythonie wypisze 256?

- ```
x = 1
for _ in range(8):
 x *= 2
print(x)
```
- ```
print(1>>8)
```
- ```
print(2**8)
```
- ```
print(1<<8)
```

33. Dane są dwie zmienne z wartościami typu bool: a oraz b. Który z warunków (będących kodem w Pythonie) wyliczy się do True tylko wtedy, gdy dokładnie jedna z tych zmiennych jest True?

- a or b
- a ^ b
- (a and not b) or (not a and b)
- a and b

34. Co wypisze następujący kod w Pythonie?

```
print('Witaj', end='')
if 'A' < 'a':
    print('!')
```

- Witaj
- zachowanie tego kodu jest niezdefiniowane, bo nie można porównywać znaków w ten sposób
- Witajend!
- Witaj!

35. Co wypisze następujący kod Python?

```
a = 'a'
A = 'AA'
print('Witaj', end='')
if A < a:
    print('!')
```

- Witaj
- Witajend!
- Witaj!
- zachowanie tego kodu jest niezdefiniowane, bo nie można porównywać znaków w ten sposób

36. Co wypisze następujący kod w Pythonie?

```
x = 1
print(x / 2)
```

- 0.5
- 1
- 0
- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)

37. Jak sprawdzić czy wartości a i b są równe?

- a===b
- a=b
- a.equals(b)
- a==b

38. Co można wstawić odpowiednio w miejsca X oraz Y, aby poniższy fragment kodu Python wypisał TAK?

```
if X or Y:
    print('TAK')
```

- 0 oraz 0
- 0 oraz 1
- 1 oraz 0
- 1 oraz 1

39. Co można wstawić odpowiednio w miejsca X oraz Y, aby poniższy fragment kodu Python wypisał TAK?

```
if X and Y:
    print('TAK')
```

- 1 oraz 1
- 1 oraz 0
- 0 oraz 0
- 0 oraz 1



40. Co można wstawić odpowiednio w miejsca X oraz Y, aby poniższy fragment kodu Python wypisał TAK?

```
if not X or Y:  
    print('TAK')
```

- 0 oraz 1
- 1 oraz 0
- 1 oraz 1
- 0 oraz 0

41. Co można wstawić odpowiednio w miejsca X oraz Y, aby poniższy fragment kodu Python wypisał TAK?

```
if X and not Y:  
    print('TAK')
```

- 0 oraz 0
- 1 oraz 1
- 0 oraz 1
- 1 oraz 0

42. Co można wstawić odpowiednio w miejsca X oraz Y, aby poniższy fragment kodu Python wypisał TAK?

```
if X < Y and not Y % 2:  
    print('TAK')
```

- 3 oraz 4
- 1 oraz 3
- 2 oraz 2
- 4 oraz 3

43. Co można wstawić odpowiednio w miejsca X oraz Y, aby poniższy fragment kodu Python wypisał TAK?

```
if X <= Y and not X % Y:  
    print('TAK')
```

- 6 oraz 2
- 1 oraz 2
- 4 oraz 4
- 2 oraz 4

44. Co zwróci $f(6)$, jeśli funkcja f jest tak zdefiniowana w języku Python:

```
def f(n):  
    if n == 0: return 1  
    return f(n-1) * n
```

- 720
- 5040
- 120
- 1

45. Co wypiszą te instrukcje?

```
print('Ala')  
print('ma')  
print('kota')
```

- Ala ma kota
- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)
- Ala
ma
kota
- Alamakota

46. W zmiennej s zapisano napis. Jak pobrać jego długość?

- `len(s)`
- `s.len()`;
- `size(s)`
- `s.size()`;

47. Który zapis pętli for w języku Python spowoduje wykonanie jej 1000 razy?

- `for _ in range(999):`
 `pass`
- `for _ in range(1, 1000):`
 `pass`
- `for _ in range(1000):`
 `pass`
- `for _ in range(1000, 0, -1):`
 `pass`

48. Ile razy wypisze się litera X dla poniższego kodu Python?

```
a = 0  
while a < 1000:  
    print('X')  
    a += 1
```

- 1
- 999
- 1001
- 1000

49. Ile razy wypisze się litera X dla poniższego kodu Python?

```
i = 0  
while i < 1000:  
    print('X')  
    i += 1  
    break
```

- 1000
- 1
- 0
- 999

50. Ile razy wypisze się litera X dla poniższego kodu Python?

```
a = 0
while a < 1000:
    print('X')
    a += 1
    continue
```

- 1000
- 999
- 0
- 1

51. Co robi fragment poniższego kodu Python?

```
FAU = [0] * 10
for i in range(10):
    FAU[i] = i
```

- wstawia do tablicy FAU kolejne wartości od 1 do 10
- wstawia do tablicy FAU kolejne wartości od 0 do 9
- wstawia do tablicy FAU kolejne wartości od 9 do 1
- wstawia do tablicy FAU kolejne wartości od 10 do 0

52. Elementy listy w języku Python indeksujemy od:

- 1
- 0
- dowolnej wartości
- 1

53. Chcemy posortować wszystkie elementy listy tab rosnąco. W języku Python zrobimy to:

- tab.sort()
- sort(tab)
- sort(tab, tab+len(tab))
- tab = sorted(tab)

54. Jaka będzie zawartość zmiennej Wynik po wykonaniu poniższego fragmentu kodu?

```
Wynik = 0
if 17 % 5 == 2:
    Wynik += 1
if 10 % 3 == 2:
    Wynik -= 1
```

- 1
- 2
- 0
- 1

55. Które z poniższych wyrażeń mogłyby znaleźć się jako warunek instrukcji if?

- '1' < 4
- 'Ala ma kota' == 'Kot ma Ale'
- 5 != 0
- 13 % 2 == 1 and 5 + 2 > 3

56. Które z wyrażeń logicznych w Pythonie są równe wartości True?

- (13 - 6 + 49) % 7 == 3
- 'ABC' == 'DEF'
- 45 // 10 == 4
- (5 + 6) > 20

57. Do czego służy w języku Python operacja logiczna not?

- zwraca wartość przeciwną do wartości wyrażenia
- sprawia, że wyrażenie jest zawsze prawdziwe
- do podkreślenia, że ten fragment kodu jest ważny
- do oznaczania niekompilujących się linii

58. Jakiego operatora logicznego użyjesz, gdy chcesz połączyć dwa wyrażenia logiczne, a wynik ma być prawdziwy, gdy co najmniej jedno z nich jest prawdziwe?

- operatora not
- operatora or
- operatora ^
- operatora and

59. Jaki jest poprawny format zapisu instrukcji warunkowej if?

- if {warunek}(...)
- if warunek { ... }
- if warunek:
- if [warunek]{ ... }

60. Ile wynosi wartość zmiennej a po wykonaniu poniższego kodu Python?

```
a = 12 + 4*3
```

- 19
- 24
- 48
- 456

61. Ile wynosi wartość zmiennej `a` po wykonaniu poniższego kodu Python?

```
a = (12+4)*3
```

- 19
- 24
- 456
- 48

62. Ile wynosi wartość zmiennej `a` po wykonaniu poniższego kodu Python?

```
a=14//5
```

- 2
- kod nie jest poprawny (podczas wykonania zostanie zgłoszony błąd)
- 2.8
- 3

63. Ile wynosi wartość zmiennej `a` po wykonaniu poniższego kodu Python?

```
a=(3*12)%5
```

- 6
- 36
- 1
- 31

64. Ile wynosi wartość zmiennej `a` po wykonaniu poniższego kodu Python?

```
a = 22+5  
a *= 4
```

- 42
- 27
- 31
- 108

65. Ile wynosi wartość zmiennej `a` po wykonaniu poniższego kodu Python?

```
a=71  
a+=1
```

- 71
- 711
- 70
- 72

66. Ile wynosi wartość zmiennej `a` po wykonaniu poniższego kodu Python?

```
a=0  
a -= 1
```

- 2
- 1
- 1
- 0

67. Czy poniższy kod jest poprawny?

```
print('ok' if 2 + 2 == 4 else 'blad')
```

- nie, ponieważ po `else` powinien być dwukropek
- nie, ponieważ `2 + 2 == 4` zawsze jest prawdziwe i obrażamy inteligencję interpretera
- tak
- nie, ponieważ `if` nie może być w tym miejscu

68. Czy następujący kod zwróci błąd?

```
if 2 + 2 == 4 or 0 // 0:  
    print('a')
```

- tak, z innych powodów niż inne odpowiedzi 'tak'
- tak, ponieważ dzielenie przez 0 jest niedozwolone
- nie
- tak, ponieważ 0 jest jedyną liczbą, której nie można podzielić przez 0

69. Zmienne `a` i `b` przechowują wartości liczbowe. Które z podanych wyrażeń są poprawne?

- `a / b`
- `a.b`
- `a + b`
- `a - b`

70. Zmienna `a` przechowuje wartość liczbową, a zmienna `b` - tekst. Które z podanych wyrażeń są poprawne?

- `b[a]`
- `a + b`
- `print(a, b)`
- `a[b]`

71. Zmienne `a` i `b` przechowują wartości tekstowe. Które z podanych wyrażeń są poprawne?

- `a + b`
- `a[b]`
- `input(a, b)`
- `a - b`

72. Jak może się nazywać funkcja w języku Python?

- return
- import
- global
- keyword

73. W jaki sposób można zakończyć wykonywanie programu napisanego w języku Python?

- można wywołać funkcję `std.terminate`
- można użyć instrukcji `return` poza funkcją
- można użyć instrukcji `goto @end`;
- można wywołać funkcję `sys.exit`

74. Proszę opisać zachowanie programu:

```
if -0 < 0:  
    print('TAK')  
else:  
    print('NIE')
```

- zachowanie programu nie jest zdefiniowane
- program na pewno wypisze NIE
- program na pewno wypisze TAK
- wywołanie zakończy się błędem

75. Co wypisze następujący program w języku Python:

```
a = 48  
b = chr(a)  
print(b)
```

- 48
- 0
- 8
- 4

76. Co wypisze następujący program w języku Python:

```
a = 'a'  
b = ord(a)  
print(b)
```

- 97
- a
- b
- 1

77. Co wypisze następujący program w języku Python:

```
a = 7  
b = float(a)  
print(b)
```

- 7.0
- 14
- 77
- 0

78. Co wypisze następujący program w języku Python:

```
a = 3.14  
b = int(a)  
print(b)
```

- 314
- 3
- 3.14
- pi

79. Zmienna `x` przechowuje jednoliterowy tekst. Które z wyrażeń sprawdzają, czy wartością zmiennej `x` jest wielką litera alfabetu angielskiego?

- `'A' <= x <= 'Z'`
- `x.isupper()`
- `x < 'A' and x > 'Z'`
- `x <= 'Z' and 'A' <= x`

80. Zmienna `x` przechowuje jednoliterowy tekst. Które z wyrażeń sprawdzają, czy wartością zmiennej `x` jest mała litera alfabetu angielskiego?

- `x <= 'a' and x >= 'z'`
- `x <= 'z' - 'a'`
- `'a' <= x <= 'z'`
- `'a' <= x and x <= 'z'`

81. Której instrukcji można użyć, żeby zwrócić wartość z funkcji

- `halt`
- `return`
- `break`
- `continue`

82. Ile gwiazdek zostanie wypisanych po wywołaniu funkcji f(5):

```
def f(n):
    if n <= 0:
        return
    for _ in range(n):
        print('*')
    f(n-2)
```

- 9
- 3
- 5
- 8

83. Ile gwiazdek zostanie wypisanych po wywołaniu funkcji f(5):

```
def f(n):
    if n <= 0:
        return
    for _ in range(n):
        print('*')
    f(n)
```

- 9
- 5
- 13
- więcej niż 100

84. Co będzie wynikiem wywołania funkcji f('012345'):

```
def f(x):
    for i in range(1, len(x)):
        x[i], x[i-1] = x[i-1], x[i]
    return x
```

- wywołanie zakończy się błędem
- 012345
- 543210
- 123450

85. Co będzie wynikiem wywołania funkcji f('012345'):

```
def f(x):
    y = list(x)
    for i in range(len(y)):
        y[i] += 1
    return ''.join(y)
```

- wywołanie zakończy się błędem
- 123456
- 123450
- 012346

86. Co będzie wynikiem wywołania funkcji f('012345'):

```
def f(x):
    y = list(x)
    for i in range(len(y)):
        y[i], y[-i-1] = y[-i-1], y[i]
    return ''.join(y)
```

- 103254
- 345012
- 012345
- 543210

87. Co wypisze następujący program:

```
a = 5
b = a
a += 1
print(a, b)
```

- 6
- 5 5
- 6 5
- 6 6

88. Które z podanych programów zamieniają wartości zmiennych a i b?

- c = a
b = a
a = c
- a = b = a
- a, b = b, a
- c = b
b = a
a = c

89. Jaka jest wartość wyrażenia tab[1] po wykonaniu poniższego programu:

```
tab = [0] * 10
tab[9] = 1
for i in range(8, 0, -1):
    tab[i] = tab[i+1] * 2
```

- 512
- 256
- 0
- 1

90. Proszę opisać zachowanie programu:

```
import math
x = math.sqrt(2)
print(round(x * x))
```

- program wypisze 1.41421
- program wypisze 1
- program wypisze 2
- program wypisze 2.0

91. Jaka jest wartość funkcji $f(10)$:

```
def f(x):
    s = 0
    while s*s < x:
        s += 1
    return s
```

- 3
- 2
- 5
- 4

92. Jaka jest wartość funkcji $f(9)$:

```
def f(x):
    s = 0
    while s*s < x:
        s += 1
    return s
```

- 3
- 4
- 5
- 2

93. Jaka jest wartość funkcji $f(9)$:

```
def f(x):
    s = 0
    while s*s <= x:
        s += 1
    return s
```

- 2
- 3
- 5
- 4

94. Zmienne a i b przechowują dodatnie wartości liczbowe. Które z poniższych wyrażeń obliczają podłogę ilorazu $\frac{a}{b}$ (tzn. największą liczbę całkowitą, która nie jest większa niż $\frac{a}{b}$).

- $a \% b$
- $(a + 1) // b$
- a / b
- $a // b$

95. Zmienne a i b przechowują wartości liczbowe, które są dodatnie. Które z poniższych wyrażeń obliczają sufit ilorazu $\frac{a}{b}$ (tzn. najmniejszą liczbę całkowitą, która nie jest mniejsza niż $\frac{a}{b}$).

- $(a + b) // b$
- $a \% b$
- $a // b$
- $(a + b - 1) // b$

96. Opisz zachowanie programu:

```
print(3 += 1)
```

- program wypisze 3
- program wypisze 4
- wywołanie zakończy się błędem
- program wypisze 3++

97. Podaj wartość funkcji f , wywołanej na liście $[3, -2, 4, 1, 7, 10, 2]$:

```
def f(a):
    x = 0
    for e in a:
        if e < x:
            x = e
    return x
```

- 0
- 2
- 3
- 2

98. Podaj wartość funkcji f , wywołanej na liście $[3, 2, 4, 1, 7, 10, 2]$:

```
def f(a):
    x = 0
    for i in range(len(a)):
        if a[i] < x:
            x = a[i]
    return x
```

- 1
- 0
- 3
- 2

99. Zmienne `a` i `b` przechowują dodatnie liczby. Które z poniższych wyrażeń sprawdzają, czy wartość zmiennej `a` jest podzielna przez wartość zmiennej `b`.
- `a // b == b`
 - `a // b == 0`
 - `a % b == 0`
 - `a % b == b`
100. Jaki będzie wynik poniższej funkcji dla argumentu `x = 20`?
- ```
def f(x):
 return 2*x
```
- 0
  - 20
  - 40
  - 20
101. Ile gwiazdek wypisze poniższy program:
- ```
n = 14
for i in range(1, n+1):
    if n % i == 0:
        print('*')
```
- 5
 - 3
 - 10
 - 4
102. Ile gwiazdek wypisze poniższy program:
- ```
n = 14
for i in range(1, n+1):
 if n % i:
 print('*')
```
- 3
  - 5
  - 4
  - 10
103. W liście `T` długości 26 zapisano kolejne litery alfabetu. `T[0] = 'a', T[1] = 'b', itd.` Jakie słowo tworzą litery kolejno `T[14], T[10], T[8]`?
- rok
  - oki
  - las
  - ala
104. W liście `T` długości 100 zapisano kolejne liczby podzielne przez 3. `T[0] = 0, T[1] = 3, T[2] = 6, itd.` Jaka liczba znajduje się w `T[10]`?
- 33
  - 6
  - 30
  - 27
105. W liście `T` długości 100 zapisano kolejne liczby podzielne przez 10. `T[0] = 0, T[1] = 10, T[2] = 20, itd.` Jaka liczba znajduje się w `T[50]`?
- 50
  - 5000
  - 500
  - 1000
106. Które z poniższych identyfikatorów są poprawnymi nazwami zmiennej w języku Python?
- `druga_liczba`
  - `DRUGA_LICZBA`
  - `int`
  - `2ga_liczba`
107. Ile wynosi ograniczenie na liczbę znaków w jednym wierszu programu w języku Python?
- takie ograniczenie nie istnieje
  - 120
  - 80
  - 160
108. Dla których z podanych wartości `n`, wywołanie funkcji `f(n)` spowoduje wypisanie dokładnie dwóch gwiazdek:
- ```
def f(n):
    for i in range(1, n+1):
        if n % i == 0:
            print('*')
```
- 2
 - 1
 - 69
 - 71
109. Opisz zachowanie poniższego programu:
- ```
def f():
 2+2

print(f())
```
- program wypisze `None`
  - wykonanie zakończy się błędem
  - program nie wypisze nic
  - program wypisze 4

110. **Opisasz zachowanie poniższego programu:**

```
print(10//3)
```

- wykonanie zakończy się błędem
- program wypisze 3
- nie wiadomo; zachowanie programu nie jest zdefiniowane
- program wypisze 10

111. **Co można wstawić w miejsce ???, żeby poniższy kod był poprawny (nie powodował błędu):**

```
if 3 ??? 4:
 print('ok')
```

- =<
- ≤
- <=
- <<=

112. **Podaj co wypisze poniższy program:**

```
f1 = list('bccbbbc')
b = 0
c = len(f1) - 1
while b <= c:
 if f1[b] == 'b': b+= 1
 else:
 f1[c], f1[b] = f1[b], f1[c]
 c -= 1
print(''.join(f1))
```

- bbbbcc
- ccbbbc
- cbcbbc
- ccbbbbb

113. **Co wypisze poniższa funkcja po wywołaniu, jeśli kolejno wprowadzane dane z klawiatury będą 4 3 5 7?**

```
def wczytaj():
 x = input()
 if int(x) == 7: return
 wczytaj()
 print(x)
```

- 7  
5  
3  
4
- 5  
3  
4
- 4  
3  
5  
7
- 4  
3  
5

114. **Co wypisze poniższa funkcja po wywołaniu, jeśli kolejno wprowadzane dane z klawiatury będą 4 3 5 7?**

```
def wczytaj():
 x = input()
 if int(x) == 7: return
 print(x)
 wczytaj()
```

- 4  
3  
5
- 5  
3  
4
- 7  
5  
3  
4
- 4  
3  
5  
7

115. Co wypisze poniższa funkcja po wywołaniu f(11)?

```
def f(n):
 if n == 0: return
 f(n//2)
 print(n % 2, end='')
```

- 1110
- 1120
- 1011
- 1001

116. Które z podanych funkcji sprawdzają, czy ich argument, będący dodatnią liczbą całkowitą, jest potęgą dwójki?

- def f(x):  
 if x == 0: return True  
 return f(x//2)
- def f(x):  
 if x == 1: return True  
 if x % 2 == 0: return False  
 return f(x//2)
- def f(x):  
 if x == 1: return True  
 if x % 2 != 0: return False  
 return f(x//2)
- def f(x):  
 if x == 1: return True  
 if x % 2 != 0: return True  
 return f(x//2)

117. Które z podanych funkcji sprawdzają, czy ich argument zawiera literę a?

- def f(s):  
 for i in range(len(s)):  
 if s[i] == 'a': return True  
 return False
- def f(s):  
 return 'a' in s
- def f(s):  
 for i in range(len(s)):  
 if i == 'a': return True  
 return False
- def f(s):  
 for i in range(len(s)):  
 if s == 'a': return True  
 return False

118. Które z podanych funkcji sprawdzają, czy ich argument, będący dodatnią liczbą całkowitą, jest liczbą nieparzystą?

- def f(x):  
 if x < 2: return x  
 return f(x-2)
- def f(x):  
 return x == 2\*k+1
- def f(x):  
 return x % 2
- def f(x):  
 return k == 2\*x+1

119. Jaka jest wartość funkcji f(11111111111111111111)?

```
def f(n):
 if n > 0: return n % 10 + f(n // 10)
 return 0
```

- 25
- 18
- 11
- 110

120. Ile gwiazdek wypisze następujący program:

```
n = 7
for i in range(n):
 for j in range(i):
 print('*', end='')
```

- 29
- 21
- 28
- 15

121. Ile gwiazdek wypisze następujący program:

```
n = 7
for i in range(n):
 for j in range(i+1):
 print('*', end='')
```

- 28
- 15
- 21
- 29

122. Ile wynosi wartość zmiennej `d` po wykonaniu poniższego kodu Python?

```
a = 5
b = 2
d = a / b
```

- 2.5
- 3
- 3.0
- 2

123. Ile wynosi wartość zmiennej `zn` po wykonaniu poniższego kodu Python?

```
a = '7'
b = '0'
zn = ord(a) - ord(b)
```

- h
- 7
- 0
- b

124. Wskaż co będzie wynikiem poniższego programu:

```
def zwieksz(n):
 n += 1
 return n

x = 6
print(zwieksz(x))
print(x)
```

- 6  
6
- 7  
6
- 6  
7
- 7  
7

125. Ile wynosi wartość zmiennej `d` po wykonaniu poniższego kodu Python?

```
a = 5
b = 2
d = a // b
```

- 5
- 2.5
- 2
- 3