|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | |  | |
| Test końcowy | | | *PS-06-01-Z1/03*  Ważne od: 2021.07.02 | |
| ***Programowanie Python na poziomie średnio zaawansowanym. Akademia IT***  ***SK-PRO-ORANGE-PYSZ***  ***Data: 28.10.2024 – 30.10.2024***  ***Wykładowca: Mariusz Gola***  ***TES\_241028\_2551500PRO\_140408AK\_*** | | |  | |
| ***Dane uczestnika testu*** | ***Wyniki – punktacja*** | |
| ***Imię: proszę uzupełnić*** | ***Maksymalna: 10 pkt.*** | |
| ***Nazwisko: proszę uzupełnić*** | ***Uzyskana:*** *10.* | |
| ***Data wypełnienia testu: 2024-10-23*** |  | |
| ***Miejscowość: proszę uzupełnić*** |  | |

1. **Jaki wynik zostanie zwrócony przez poniższy kod (co wyświetli się na ekranie) ?**

**def modlist1(\*arg):**

**print(len(arg))**

**lst1=[1,2,3,4,5]**

**modlist1(\*lst1)**

**0**

**5**

**1**

**Instrukcja wygeneruje błąd**

Liczba punktów: ***\_\_\_\_\_1***

1. **Jaki wynik zostanie zwrócony przez poniższy kod (co wyświetli się na ekranie) ?**

**lst1=[[4,2],[3,1],[1,77],[5,6]]**

**ts=lambda x:x[1]**

**lst2=sorted(lst1,key=ts)**

**print(lst2[1][0])**

**5**

**3**

**1**

**4**

Liczba punktów: ***\_\_1\_\_\_***

1. **Jaki wynik zostanie zwrócony przez poniższy kod (co wyświetli się na ekranie) ?**

**def fun():**

**num = 1**

**while num<4:**

**yield num**

**num += 1**

**for i in fun():**

**print(i, end="")**

**1 2 3**

**123**

**0123**

**kod wygeneruje błąd**

Liczba punktów: ***\_\_1\_\_\_***

1. **Jaki wynik zostanie zwrócony przez poniższy kod (co wyświetli się na ekranie) ?**

**class A:**

**def \_\_init\_\_(self):**

**print('a',end="")**

**class B:**

**def a(self):**

**print('b')**

**class C(A,B):**

**def c(self):**

**self.a()**

**o1=C()**

**o2=C()**

**aa**

**ab**

**ba**

**nic się nie wyświetli**

Liczba punktów: ***\_1\_\_\_\_***

1. **Jaki wynik zostanie zwrócony przez poniższy kod (co wyświetli się na ekranie) ?**

t=”przedszkole”

**print(t[1::2])**

**przedszkole**

**przedszko**

**reskl**

**nic się nie wyświetli**

Liczba punktów: ***\_\_1\_\_\_***

1. **Jaki wynik zostanie zwrócony przez poniższy kod (co wyświetli się na ekranie) ?**

class A:

X=0

def \_\_init\_\_(self,v=0):

self.Y=v

A.X+=v

a=A()

b=A(1)

c=A(2)

print(c.X)

**1**

**2**

**3**

**Instrukcja wygeneruje błąd**

Liczba punktów: ***\_1\_\_\_\_***

1. **Jaki wynik zostanie zwrócony przez poniższy kod (co wyświetli się na ekranie) ?**

**class A:**

**def \_\_init\_\_(self,v):**

**self.\_\_a=v+1**

**a=A(0)**

**print(a.\_\_a)**

**1**

**0**

**3**

**Instrukcja wygeneruje błąd**

Liczba punktów: ***\_\_1\_\_\_***

1. **Jaki wynik zostanie zwrócony przez poniższy kod ?**

**dct={}**

**dct["1"]=(3,4)**

**dct["2"]=(4,2)**

**for x in dct.keys():**

**print(dct[x][1], end="")**

**(3,4)**

**42**

**34**

**[4,2]**

Liczba punktów: ***\_1\_\_\_\_***

1. **Jaki wynik zostanie zwrócony przez poniższy kod ?**

**def fun (x,y):**

**if x==y:**

**return x**

**else:**

**return fun (x,y-1)**

**print(fun(2,4))**

**2**

**4**

**3**

**0**

Liczba punktów: ***\_\_\_1\_\_***

1. **Ile elementów przechowuje lista lst po wykonaniu poniższego kodu ?**

**lst=[i for i in range(-2,-1)]**

**dwa**

**żadnego**

**jeden**

**Instrukcja wygeneruje błąd**

Liczba punktów: ***\_1\_\_\_\_***