

**ID: 0003**

**2. Opravný termín, semestrální test ISJ, 8.6.2012, Login:**

1. (2 points) Jaký je rozdíl mezi seznamem (list) a n-ticí (tuple) v Pythonu?
  - A. Pro n-tici platí, že nelze měnit její velikost ani obsah.
  - B. Na n-tici nelze aplikovat operátor \*.
  - C. Seznam není vestavěný typ.
  - D. N-tici nelze iterovat.
2. (2 points) K čemu slouží příkaz puts v jazyk Ruby?
  - A. Tisk řetězce na výstup.
  - B. Zvýšení velikosti pole o 1.
  - C. Přidání prvku do pole.
  - D. Vyjmutí prvního prvku z pole.
3. (2 points) Který z následujících jazyků nemá vestavěnou podporu pro regulární výrazy?
  - A. Python 3
  - B. Ruby
  - C. C
  - D. Python 2
4. (2 points) K čemu slouží pass příkaz v Pythonu?
  - A. Vyhození výjimky.
  - B. Prázdná operace.
  - C. Násilné ukončení cyklu.
  - D. Ukončení programu s návratovou hodnotou 0.
5. (2 points) Který z těchto jazyků je nejvíce objektový?
  - A. Ruby
  - B. Python 3
  - C. C
  - D. Python 2
6. (2 points) Který z následujících příkazů neexistuje v jazyce Python?
  - A. puts
  - B. cut
  - C. del
  - D. raise
7. (2 points) Která z následujících tvrzení platí pro datový typ seznam (list) v Pythonu?
  - A. Můžeme měnit jeho obsah.
  - B. Je takzvaným immutable typem.

C. Nepatří do vestavěných typů jazyka Python.

D. Můžeme měnit jeho velikost.

8. (2 points) Které řetězce odpovídají následujícímu regulárnímu výrazu  $[0-9a-fA-F]\{2\}$ ?

A. 0L

B. 16.0

C. 147

D. a9

9. (2 points) Které z následujících jazyků mají zabudovanou podporu (standardní knihovnu) regulárních výrazů?

A. Python

B. Java SE

C. Ruby

D. C

10. (2 points) Která z následujících tvrzení platí pro datový typ n-tice (tuple) v jazyce Python?

A. Je takzvaným immutable typem.

B. Můžeme měnit jeho velikost.

C. Nepatří do vestavěných typů jazyka Python.

D. Můžeme měnit jeho obsah.

11. (2 points) Napište jaký je rozdíl mezi n-ticí a seznamem v Pythonu.

12. (2 points) Napište jediným RE přidání oddělení řádu mocnin tečkou (1214248 -> 1.214.248).

13. (2 points) Srovnejte jazyky Ruby a Python.

14. (2 points) Napište regulární výraz pro nalezení data ve formátu YYYY-MM-DD.

15. (2 points) Napište vše co víte o iterátorech v Pythonu.

16. (2 points) Napište program v jazyce Python3, Python2 nebo Ruby, který ze standardního vstupu, kterým je CSV soubor obsahující  
login;počet\_bodů  
login;počet\_bodů

login;počet\_bodů

atd...

vypiše na standardní výstup celkový počet bodů všech studentů a na nový řádek login studenta s největším počtem bodů.

17. (2 points) Doplňte tělo funkce `divisible(sez)` v Pythonu 2 tak, aby vrátil nový seznam čísel, která jsou dělitelná číslem 3.

`sez` je seznam obsahující proměnné typu `int`

18. (2 points) Napište program v Pythonu 3, který vypíše na standardní výstup počet nepřekrývajících se řetězců typu YYYY-MM-DD na standardním vstupu.

19. (2 points) Napište funkci `myfilter(array)` v jazyce Ruby tak, aby ze vstupního seznamu čísel vytiskla na standardní výstup pouze lichá čísla a čísla dělitelná 10.

20. (2 points) Doplňte tělo funkce fakt(num) v Pythonu 3, taky aby vracela faktoriál celého čísla.

21. (2 points) Regulárnímu výrazu:  $\wedge((66)\{3\}(?:78)\{2\}(81)\{2\})\{2\}\{3\}$  odpovídá řetězec:

- A. 66666678788166666678788178
- B. 6666667881666666788178
- C. 6666667878818178788181666666787881817878818178788181
- D. 666666787881816666667878818178
- E. 6666667878818178788181666666787881817878818181
- F. 6666667878818178788181666666787881817878818178

22. (2 points) Co vypíše následující program v Ruby na standardní výstup?

```
x = 1
z = 2
n = 1
a = [1,4,2,4]
(n...5).each{|y| z += x += y}
x.times {|i| z -= 2}
n.upto(a.length) {|y| z -= y}
n.upto(a.first+a.first) {|y| x -= y}
print x.abs - z.abs
```

- A. 11
- B. 2
- C. 13
- D. 0
- E. 3

23. (2 points) Jak dlouhá je odvěsna pravoúhlého trojúhelníka pokud je druhá odvěsna dlouhá 108 cm a přepona měří 117 cm?

- A. 47 cm
- B. 48 cm
- C. 46 cm
- D. 49 cm
- E. 45 cm
- F. 50 cm

24. (2 points) Co vypíše následující program v Pythonu 3 na standardní výstup?

```
class Houbogen:
    def __init__(self, n):
        self.i = n + -4
        self.j = 4 - n
    def __iter__(self):
        return self
    def __next__(self):
        self.j -= 1
        if self.i > self.j:
            raise StopIteration()
        elif sum((self.i, self.j)) % 2 == 1:
            self.j -= 1
            print('houbaraz:', self.j, end=', ')
            return self.j - self.i
        else:
            self.j -= 1
            print('houbadva:', self.i-self.j, end=', ')
            return self.j - self.i
```

n = 0

```
for h in Houbogen(n):
    print('les:', h, end=', ')
```

- A. houbaraz: 2, les: 6, houbaraz: 0, les: 4, houbaraz: -2, les: 2, houbaraz: -4, les: 0,
- B. houbadva: -5, les: 5, houbadva: -3, les: 3, houbadva: -1, les: 1, houbadva: 1, les: -1,
- C. houbaraz: 2, les: 5, houbadva: -3, les: 3, houbadva: -1, les: 1, houbadva: 1, les: -1,
- D. houbadva: 5, les: 5, houbadva: 3, les: 3, houbadva: 1, les: 1, houbadva: -1, les: -1,
- E. houbadva: 5, les: 5, houbaraz: -1, les: -2, houbadva: 0, les: 0,
- F. houbaraz: 1, les: -3, houbadva: 1, les: 1, houbadva: -1, les: -1,

25. (2 points) Co vypíše následující program v Pythonu 2 na standardní výstup?

```
class Houbogen:
    def __init__(self, n):
        self.i = n + -3
        self.j = 4 - n
    def __iter__(self):
```



```

        return self
def next(self):
    self.i += 1
    if self.i > self.j:
        raise StopIteration()
    elif sum((self.i, self.j)) % 2 == 1:
        self.i += 1
        self.j -= 1
        return self.i - self.j
    else:
        self.i += 1
        return self.i - self.j
n = 1
print sum(Houbogen(n))

```

- A. 3
- B. 2
- C. -8
- D. -6
- E. -3
- F. -10

26. (2 points) Napište program v jazyce Python 3 který sečte libovolně velké matice a vytiskne výsledek na standardní výstup. Prvky matice jsou odděleny mezerou. Matice na vstupu jsou od sebe odděleny prázdným řádkem a mají stejnou velikost. Příklad vstupu:

```

1 1
2 2

```

```

3 3
4 4

```

Odpovídající výstup:

```

4 4
6 6

```

27. (2 points) Co vypíše následující program v Pythonu 3 na standardní výstup (neuvažujte chybový výstup)?

```
def f(n):
    print('fraz:', n, end=', ')
    n += 2
    while n < 8:
        yield n
        n += 1
        print('fdva:', n, end=', ')
    print('ftri:', n, end=', ')
n = 0
x = f(n)
print('mráz:', n, end=', ')
print('mdva:', n, end=', ')
next(x)
print('mtri:', next(x), end=', ')
```

- A. mráz: 0, mdva: 0, fraz: 0, fdva: 3, mtri: 3,
- B. mráz: 0, fraz: 0, mdva: 2, fdva: 3, fdva: 4, mtri: 4,
- C. mráz: 0, fraz: 0, mdva: 2, mtri: 0,
- D. fraz: 0, mráz: 2, fdva: 4, mdva: 4, fdva: 6, mtri: 0,
- E. mráz: 0, fraz: 0, mdva: 2, fdva: 3, mtri: 0,
- F. mráz: 0, fraz: 0, mdva: 2, fdva: 4, mtri: 0,

28. (2 points) Regulárnímu výrazu:  $^{\wedge}[2]\backslash\{\}s\backslash\{\}d\backslash\{\}w[^9][1-8](19)\$$  odpovídá řetězec:

- A. . 4\$4619

- B. . 5\$7419
- C. 2 5\$1519
- D. 9 155819
- E. ^ 5\$551
- F. ^ 4\$441
- G. ^ 6\$7419
- H. ^ 929419