

Számítógépes Hálózatok

1. gyakorlat

Elérhetőségek

- honlap: <http://szalaigj.web.elte.hu/>
- email: szalaigindl@inf.elte.hu
- szoba: 2.507 (déli tömb)

Követelmények

- Maximum 4 hiányzás
- Minden óra elején pár kérdéses teszt
- Jegy összetétele:
 - 40% SocketZH félév végén
 - 25% Teszt óra elején – 10 kérdés az előző heti EA anyagából
 - 35% Órai feladatmegoldás / házi feladat

Órai / Házi feladatok

- Órai feladatok:
 - Gyakorlat idejében lehet megoldani és bemutatni.
 - Ha nem sikerül végezni, akkor a következő órára fele pontszámért be lehet mutatni.
 - Ez után már bemutatásra nincs lehetőség.
- Házi feladat:
 - Általában valamilyen programozási feladat
 - Általában 1 hét lesz a megoldásra
 - 1 hét késés lehetséges fele pontszámért

Óra elei kisZH

- Elérés:
 - <https://canvas.elte.hu>

≡ 2018/19/1 XK85DZ-IP-08abcSZHG - Számítógépes hálózatok GY. (BSc,08,A) > Kvízek

2018/19/1

Kezdőlap

Hirdetmények

Feladatok

Fórumok

Értékelések

Résztvevők

Oldalak

Fájlok

Tematika

Tanulási eredmények

Kvízek


Modulok

Beállítások


+ Kvíz

⚙️

▼ Gyakorló kvízek

 Demo kvíz

Elérhető Többes határidő | Határidő Többes határidő | 5 pont | 5 kérdés

 ⚙️

Ponthatárok

$$\begin{aligned} \text{Jegy} = & \\ & (\text{szerzettTeszt} / \text{maxTeszt}) * 0,25 \\ & + \\ & (\text{szerzettFeladat} / \text{maxFeladat}) * 0,35 \\ & + \\ & (\text{szerzettZH} / \text{maxZH}) * 0,4 \end{aligned}$$

Százalék	Érdemjegy
0 - 49 %	Elégtelen (1)
50 - 59 %	Elégséges (2)
60 - 74 %	Közepes (3)
75 - 84 %	Jó (4)
85 – 100 %	Jeles (5)

Python történelem és tulajdonságok

- Guido Van Rossum holland programozó készítette a 90-es évek elején
 - nevét a Monty Python Repülő Cirkusza után kapta
- Python tulajdonságai:
 - Interpretált
 - Interaktív, azaz egy parancssor héjat ad
 - Objektum-orientált nyelv
 - Hordozható
 - Kezdők számára általa könnyen elsajátíthatóak a programozási alapismeretek

Python parancssor

```
#python  
python> import this  
python> print "Hello world!"  
python> user_name="Jozsi"  
python> print "Hello " + user_name  
python> user_age=25  
python> print "You are " + str(user_age) + " years old."
```


Egyszerű számítások

```
Python>10+2
```

```
12
```

```
Python>2*2
```

```
4
```

```
Python>3**2
```

```
9
```

```
Python>10%2
```

```
0
```

Változók

```
Python> a = 42
Python> b = 32
Python> c = a + b
Python> print(c)
74
Python> c = 'valami'
Python> print(a+c)
ERROR
```

String műveletek

```
Python>print 'alma'.upper()  
ALMA
```

```
Python>print( "LO" in "Hello".upper() )  
True
```

```
Python>print "Decimal Number: %d, Float: %f, String: %s" %  
(12,33.4,"almafa")  
Decimal Number: 12, Float: 33.400000, String: almafa
```

Listák

```
Python> players = [12,31,27,'48',54]
Python> print players
[12, 31, 27, '48', 54]
Python> players[0]
12
Python> players[-1]
54
Python> players + [22, 67]
[12, 31, 27, '48', 54, 22, 67]
Python> print len(players)
5
```

Listák

```
Python> players = [12,31,27,'48',54]
Python> players.append(89)
Python> print len(players)
6
Python> players[2:]
[27, 48, 54, 89]
```

Halmazok

```
Python> mylist = [8,3,2,3,2,4,6,8,2]
Python> myset = set(mylist)
Python> print mylist
[8, 3, 2, 3, 2, 4, 6, 8, 2]
Python> myset = {8,3,2,3,2,4,6,8,2} # alternative
Python> print myset
set([8, 2, 3, 4, 6])
Python> type(myset)
<type 'set'>
Python> mysortedlist = sorted(mylist)
Python> print mysortedlist
[2, 2, 2, 3, 3, 4, 6, 8, 8]
```

Szótár

```
Python> team = {  
    91: "Ayers, Robert",  
    13: "Beckham Jr,",  
    3: "Brown, Josh",  
    54: "Casillas, Jonathan",  
    21: "Collins, Landon"}  
Python> len(team)  
5  
Python> team[3] = "Chihiro"  
Python> print team.has_key(91)  
True  
Python> print team.has_key('alma')  
False
```

Szótár

```
Python> team = {  
    91: "Ayers, Robert",  
    13: "Beckham Jr,",  
    3: "Brown, Josh",  
    54: "Casillas, Jonathan",  
    21: "Collins, Landon"}  
Python> print team.keys()  
[91,13,3,54,21]  
Python> print team.values()
```


Elágazások

```
if 100 in team:  
    print 'Yes, 100 is in the team'  
elif 76 in team:  
    print '100 is not in the team, but 76 is in it...'  
else:  
    print 'Both 100 and 76 are not in the team'
```

Ciklus

```
mylist = [3,65,2,77,9,33]
```

```
for i in mylist:  
    print 'Element:', i
```

Írassuk ki a lista elemeit növekvő sorrendben!

```
for i in xrange(2,10,2): #2-től 9-ig 2-esével  
    print i
```

Ciklus

```
for i in range(5):  
    print "Number" + str(i)  
  
for (k,v) in team.iteritems():  
    print "Player name: %s; #: %d" % (v,k)  
  
Player name: Brown, Josh; #: 3  
Player name: Nassib, Ryan; #: 12  
...
```

```
i=1  
while i<10:  
    print i  
    i+=1
```

Python script futtatása

```
#vi test.py

#!/usr/envbin/python
x = 1
for i in range(1,5):
    x+=i
    print x,i,'alma', 'x*x = %d' % (x*x)
    print(str(i) + " alma")

#python test.py
#chmod +x test.py && ./test.py
```

Függvények

```
#!/usr/env/bin/python

def is_even(num):
    if (num % 2) == 0:
        return True
    else:
        return False

for i in range(1,10):
    if (is_even(i)):
        print("Szam:"+str(i))

print("Vege")
```

Feladat 1.

Írjunk függvényt ami megadja egy bemenetben kapott évszámról, hogy szökőév-e. Egy év szökőév, ha osztható négygyel, de akkor nem, ha osztható százszal, hacsak nem osztható négyszázzal. Példák: 1992,1996,2000,2400 szökőév, de 1993, 1900 nem.

Feladat 2.

A hét napjait jelöljük 0-6-ig (Hétfő,...,Vasárnap).
Írjunk egy függvényt, ami megadja mikor kell
kelnünk az adott napon (hétköznapi '7:00' hétvégén
'10:00'), kivéve ha vakációzunk, mert akkor
hétköznapi '10:00' hétvégén 'OFF'

`alarm_clock(1, False) → '7:00'`

`alarm_clock(6, False) → '10:00'`

`alarm_clock(0, True) → '10:00'`

`alarm_clock(6, True) → 'OFF'`

Feladat 3.

Írjunk függvényt ami megadja az n. fibonacci számot

$\text{fibonacci}(0) \rightarrow 0$

$\text{fibonacci}(1) \rightarrow 1$

$\text{fibonacci}(2) \rightarrow 1$

$\text{fibonacci}(3) \rightarrow 2$

...

$\text{fibonacci}(n) \rightarrow \text{fibonacci}(n-2) + \text{fibonacci}(n-1)$

VÉGE
KÖSZÖNÖM A FIGYELMET!