

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Кафедра вычислительной математики и программирования

Дисциплина: «Разработка ПО для высокопроизводительных систем»

Отчет по лабораторной работе №2 по NumPy

Выполнил: Павлов М.М.

Группа: М8О-103М-20

Преподаватель: Поповкин А.В.

```
import numpy as np
import sys
import scipy.spatial

#1
a = np.dot(np.ones((5,3)), np.ones((3,2)))
print(a)
print()

#2
b = np.arange(20)
print(b)
b[(3 < b) & (b <= 8)] *= -1
print(b)
print()

#3
c = np.zeros((5,5))
c += np.arange(5)
print(c)
print()

#4
def generator():
    for x in range(10):
        yield x
d = np.fromiter(generator(),dtype=float,count=-1)
print(d)
print()

#5
e = np.linspace(0,1,12)[1:-1]
print(e)

#6
```



```

print()
#12
l1 = np.arange(8)
l2 = l1 + 0.5
l3 = 1.0 / np.subtract.outer(l1, l2)
print(l3)
print()
#13
for dtype in [np.int8, np.int32, np.int64]:
    print(np.iinfo(dtype).min, np.iinfo(dtype).max)
for dtype in [np.float32, np.float64]:
    print(np.finfo(dtype).min, np.finfo(dtype).max)
#14
np.set_printoptions(threshold=sys.maxsize)
m = np.zeros((25,25))
print(m)
print()
#15
n = np.arange(100)
o = np.random.uniform(0,100)
p = (np.abs(n-o)).argmin()
print(n,o)
print(n[p])
print()
#16
r = np.zeros(10, [ ('position', [ ('x', float, 1),
                                ('y', float, 1)]),
                  ('color', [ ('r', float, 1),
                              ('g', float, 1),

```

```
('b', float, 1))]]))
```

```
print(r)
```

```
print()
```

```
#17
```

```
Z = np.random.random((10,2))
```

```
D = scipy.spatial.distance.cdist(Z,Z)
```

```
print(D)
```

```
print()
```

```
#18
```

```
s = np.arange(10, dtype=np.int32)
```

```
print(Z.astype(np.float32, copy=False))
```

```
print()
```

```
#19
```

```
#####
```

```
#20
```

```
t = np.arange(9).reshape(3,3)
```

```
for index, value in np.ndenumerate(t):
```

```
    print(index, value)
```

```
print()
```

```
for index in np.ndindex(t.shape):
```

```
    print(index, t[index])
```

```
print()
```

```
#21
```

```
X, Y = np.meshgrid(np.linspace(-1,1,10), np.linspace(-1,1,10))
```

```
D = np.hypot(X, Y)
```

```
sigma, mu = 1.0, 0.0
```

```
G = np.exp(-((D - mu) ** 2 / (2.0 * sigma ** 2)))
```

```
print(G)
```

```
print()
```

#22

n = 10

p = 3

Z = np.zeros((n,n))

np.put(Z, np.random.choice(range(n*n), p, replace=False), 1)

print(Z)

print()

#23

X = np.random.rand(5, 10)

print(X - X.mean(axis=1, keepdims=True))

print()

#24

Z = np.random.randint(0,10,(3,3))

print(Z)

print(Z[Z[:,0].argsort()])

print()

#25

Z = np.random.randint(0,3,(3,10))

print(Z)

print((~Z.any(axis=0)).any())

print()

#26

Z = np.ones(10)

I = np.random.randint(0,len(Z),20)

Z += np.bincount(I, minlength=len(Z))

print(Z)

print()

#27

w,h = 16,16

```
I = np.random.randint(0, 4, (h,w,3)).astype(np.ubyte)
r = I.reshape(-1,3)
u = np.unique(r, axis=0)
print(I)
print(r)
print()
print(len(u))
print()
#28
A = np.random.randint(0,10, (3,4,3,4))
sm = A.reshape(A.shape[:-2] + (-1,)).sum(axis=-1)
print(sm)
print()
#29
#np.diag(np.dot(A, B))
```