

Глава 2

```
In [1]: import numpy as np
import math
import matplotlib.pyplot as plt
import scipy
import scipy.stats
```

```
In [2]: # generate all sequences of length l
def generate(l):
    res = []

    for i in range(1, 2**l):
        b = bin(i)[2:]
        b = '0' * (l - len(b)) + b
        b = np.array(list(map(lambda x: int(x), b)))
        res.append(b)

    return np.array(res)
```

Задание 14

Будем пользоваться **Свойством 2.2**: $\rho = \max_s \rho(s)$

```
In [3]: def findRadius(H, title):
    r = 0
    for e in generate(H.shape[1]):
        r = max(r, np.sum(np.dot(e, H.T) % 2))

    print("{}: radius = {}".format(title, r))
```

```
In [4]: H1 = np.array([
    [1, 1, 0, 0, 0, 0],
    [1, 0, 1, 0, 0, 0],
    [1, 0, 0, 1, 0, 0],
    [1, 0, 0, 0, 1, 0],
    [1, 0, 0, 0, 0, 1]
])
H2 = np.array([
    [1, 1, 1, 0, 0, 0],
    [1, 1, 0, 1, 0, 0],
    [0, 1, 0, 0, 1, 0],
    [1, 1, 1, 1, 1, 1]
])
H3 = np.array([
    [0, 1, 1, 1, 0, 0],
    [1, 0, 1, 0, 1, 0],
    [1, 1, 0, 0, 0, 1]
])
H4 = np.array([
    [1, 0, 1, 0, 1, 0],
    [0, 1, 0, 1, 0, 1]
])
H5 = np.array([
    [1, 1, 1, 1, 1, 1]
])
```

```
In [5]: findRadius(H1, "n = 6, k = 1, d = 6")
```

```
n = 6, k = 1, d = 6: radius = 5
```

```
In [6]: findRadius(H2, "n = 6, k = 2, d = 4")
```

```
n = 6, k = 2, d = 4: radius = 4
```

```
In [7]: findRadius(H3, "n = 6, k = 3, d = 3")
```

```
n = 6, k = 3, d = 3: radius = 3
```

```
In [8]: findRadius(H4, "n = 6, k = 4, d = 2")
```

```
n = 6, k = 4, d = 2: radius = 2
```

```
In [9]: findRadius(H5, "n = 6, k = 5, d = 2")
```

```
n = 6, k = 5, d = 2: radius = 1
```

Эти коды нельзя улучшить с точки зрения радиуса покрытия так как: $\rho \leq n - k$. Для последнего кода ($n = 6, k = 5, d = 2$): $\rho = 0$ и его тоже нельзя улучшить.