

## Задания к работе

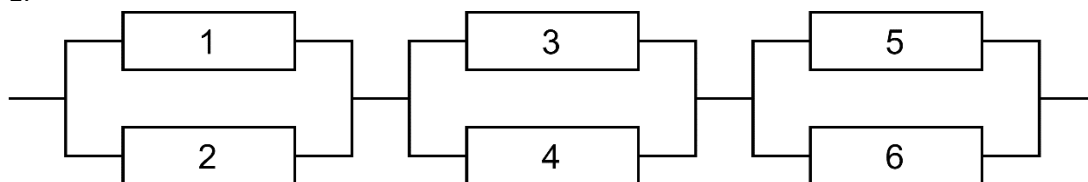
1. Построить структурную схему системы;
2. Выполнить анализ системы методом, основанным на теоремах теории вероятности;
3. Выполнить анализ системы методом минимальных путей и минимальных сечений;
4. Составить для системы функцию алгебры логики;
5. Применить алгоритм разрезания для вычисления вероятности безотказной работы системы;
6. Составить программу, моделирующую работу системы при разном режиме обслуживания и случайном времени отказов и восстановлений.

### Задание варианта №4

Система состоит из 3 узлов, каждый из которых резервируется дублирующим элементом.

#### Выполнение

1. 1.



2. Вероятность отказа узла равна произведению вероятности отказа основного и дублирующего элемента, тогда вероятность безотказной работы узла  $P_j = 1 - q_i q_{i+1}, j = 1, 2, 3, i = 1, 3, 5$ .

Вероятность безотказной работы системы равна произведению этих вероятностей:  $P = (1 - q_1 q_2)(1 - q_3 q_4)(1 - q_5 q_6)$ , а вероятность отказа:  $Q = 1 - P$ .

3. Минимальные пути: 1-3-5, 1-3-6, 1-4-5, 1-4-6, 2-3-5, 2-3-6, 2-4-5, 2-4-6;

Минимальные сечения: 1-2, 3-4, 5-6.

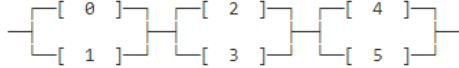
$$\begin{aligned}
 P = & p_1 p_3 p_5 + p_1 p_3 p_6 + p_1 p_4 p_5 + p_1 p_4 p_6 + p_2 p_3 p_5 + p_2 p_3 p_6 + p_2 p_4 p_5 + p_2 p_4 p_6 - \\
 & p_1 p_3 p_5 p_6 - p_1 p_3 p_4 p_5 - p_1 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_5 - p_1 p_2 p_3 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 - \\
 & p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_3 p_4 p_5 p_6 + p_1 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_5 p_6 + p_1 p_2 p_3 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + \\
 & p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_3 p_4 p_5 + p_1 p_2 p_3 p_4 p_5 + \\
 & p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + \\
 & p_1 p_2 p_3 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + \\
 & p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + \\
 & p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_6 + \\
 & p_1 p_2 p_3 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_6 + \\
 & p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_4 p_5 p_6 + p_1 p_2 p_4 p_5 p_6 + \\
 & p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + \\
 & p_1 p_2 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + p_1 p_2 p_3 p_4 p_5 p_6 + \\
 & p_1 p_2 p_3 p_4 p_6 + p_1 p_2 p_4 p_5 p_6 + p_2 p_3 p_4 p_5 p_6 + p_2 p_3 p_4 p_5 p_6 + p_2 p_3 p_4 p_5 p_6 + p_2 p_3 p_4 p_5 p_6 - \\
 & p_1 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - \\
 & p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_5 p_6 - \\
 & p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - \\
 & p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - \\
 & p_1 p_2 p_3 p_4 p_5 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - \\
 & p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - \\
 & p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6 - \dots
 \end{aligned}$$

$$Q = q_1 q_2 + q_3 q_4 + q_5 q_6 - q_1 q_2 q_3 q_4 - q_1 q_2 q_5 q_6 - q_3 q_4 q_5 q_6 + q_1 q_2 q_3 q_4 q_5 q_6$$

5. Разрезание:  $x_1(x_3x_5 \vee x_3x_6 \vee x_4x_5 \vee x_2x_3x_5 \vee x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6) \vee$   
 $\bar{x}_1(x_2x_3x_5 \vee x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6) = x_1(x_3(x_5 \vee x_6) \vee x_4x_5 \vee x_2(x_3 \vee x_4)(x_5 \vee x_6)) \vee$   
 $\bar{x}_1x_2(x_3 \vee x_4)(x_5 \vee x_6)$

## Пример работы программы

Система варианта №4:



Вероятности безотказной работы элементов  $P = [0.99584276598381, 0.99584276598381, 0.9718421254018641, 0.9718421254018641, 0.9627$   
Вероятности отказов  $Q = [0.004157234016189937, 0.004157234016189937, 0.028157874598135915, 0.028157874598135915, 0.0372463172448$

Анализ системы на основе теорем теории вероятности:  $P = (1-q_0q_1)(1-q_2q_3)(1-q_4q_5) = 0.9978037009483269$ ;  $Q = 1 - P = 0.00219629905$

Анализ системы методом минимальных путей и сечений:

Минимальные пути:  $\{0, 2, 4\}, \{0, 2, 5\}, \{0, 3, 4\}, \{0, 3, 5\}, \{1, 2, 4\}, \{1, 2, 5\}, \{1, 3, 4\}, \{1, 3, 5\}$

Минимальные сечения:  $[\{0, 1\}, \{2, 3\}, \{4, 5\}]$

[illegible]

Вероятность отказа методом минимальных путей и сечений:  $Q = q_0q_1 + q_2q_3 + q_4q_5 - q_0q_1q_2q_3 - q_0q_1q_4q_5 - q_2q_3q_4q_5 + q_0q_1q_2q_3q_4q_5$   
 $Q = 0.002196299051673146$

Функция работоспособности:  $y = x_1x_3x_5 \vee x_1x_3x_6 \vee x_1x_4x_5 \vee x_1x_4x_6 \vee x_2x_3x_5 \vee x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6$

Функция неработоспособности  $y = !x_1!x_2 \vee !x_3!x_4 \vee !x_5!x_6$

$$\text{Разрезание: } y = x1x3x5 \vee x1x3x6 \vee x1x4x5 \vee x1x4x6 \vee x2x3x5 \vee x2x3x6 \vee x2x4x5 \vee x2x4x6 = x1(x3x5 \vee x3x6 \vee x4x5 \vee x2x3x5 \vee x2x3x6 \vee x2x4x5 \vee x2x4x6) \vee !x1(x2x3x5 \vee x2x3x6 \vee x2x4x5 \vee x2x4x6) = x1(x3(x5 \vee x6) \vee x4x5 \vee x2(x3 \vee x4)(x5 \vee x6)) \vee !x1x2(x3 \vee x4)(x5 \vee x6)$$

Теоретически:  $P = 0.9892266453948466$ ;  $Q = 0.010773354605153385$

Вероятность безотказной работы в результате эксперимента: 0.989479

Вероятность отказа в результате эксперимента: 0.010521

Теоретически:  $P = 0.9945004115789966$ ;  $Q = 0.0054995884210033985$

Вероятность безотказной работы в результате эксперимента: 0.994436

Вероятность отказа в результате эксперимента: 0.005564

Теоретически:  $P = 0.9889063713210583$ ;  $Q = 0.011093628678941703$

Вероятность безотказной работы в результате эксперимента: 0.988831

Вероятность отказа в результате эксперимента: 0.011169

Вывод: в ходе лабораторной работы была построена структурная схема системы, проведён её анализ методами теории вероятности, методом минимальных путей и сечений, для системы была составлена функция алгебры логики и к этой функции был применён алгоритм разрезания. Для полученной системы была написана программа, выполняющая анализ системы и моделирующая работу системы для трёх разных режимов. Результаты эксперимента совпадают с теоретическими данными с точностью до десятичных долей единицы.

### Код программы

```
import numpy
import random
```

```

from itertools import combinations

print('''Система варианта №4:
  ┌───[ 0 ]───┐ ┌───[ 2 ]───┐ ┌───[ 4 ]───┐
  │           │ │           │ │           │
  └───[ 1 ]───┘ └───[ 3 ]───┘ └───[ 5 ]───┘
''')

n = 6 # КОЛИЧЕСТВО ЭЛЕМЕНТОВ
P = []
Q = []
for i in range(0, int(n/2)):
    tmp = random.random() * 0.1
    P.append(1 - tmp)
    Q.append(tmp)
    P.append(1 - tmp)
    Q.append(tmp)
print("Вероятности безотказной работы элементов P = {} \nВероятности отказов Q = {
} \n".format(P, Q))

min_ways = [{0, 2, 4}, {0, 2, 5}, {0, 3, 4}, {0, 3, 5}, {1, 2, 4}, {1, 2, 5}, {1,
3, 4}, {1, 3, 5}]
min_cuts = [{0, 1}, {2, 3}, {4, 5}]

P_tv = (1 - Q[0]**2)*(1 - Q[2]**2)*(1-Q[4]**2)
Q_tv = 1 - P_tv
print("Анализ системы на основе теорем теории вероятности: P = (1-q0q1)(1-
q2q3)(1-q4q5) = {}; Q = 1 - P = {} \n".format(P_tv, Q_tv))

print("Анализ системы методом минимальных путей и сечений:")
print("Минимальные пути: {} \nМинимальные сечения: {} \n".format(min_ways, min_cuts
))

def getStrExpression(combinations, ch, sign):
    res_str = ""
    for comb in combinations:
        res_str += " " + sign + " "
        for j in range(n):
            if j in comb:
                res_str += "{}{}".format(ch, j)

    return res_str

def uniCombinations(combinations, k, set_):
    res = []
    for comb in combinations:
        t = set()
        for j in range(len(set_)):
            if j in comb:
                t = t | set_[j]
        res.append(t)

```

```

    return res

def computeCombinations(combinations, p_q):
    res = 0
    for comb in combinations:
        t = 1
        for j in range(n):
            if j in comb:
                t *= p_q[j]
        res += t
    return res

ways_num = [i for i in range(0, len(min_ways))]
exp_str = ""
exp_res = 0
for k in range(1, len(min_ways) + 1):
    comb = list(combinations(ways_num, k))
    comb = uniCompinations(comb, k, min_ways)
    if (k % 2 != 0):
        exp_res += computeCombinations(comb, P)
    else:
        exp_res -= computeCombinations(comb, P)

    str_ = getStrExpression(comb, "p", "+" if (k % 2 != 0) else "-")
    exp_str += str_
exp_str = "P =" + exp_str[2:]
print("Вероятность безотказной работы методом минимальных путей и сечений: {}".format(exp_str))
print("P = {}".format(exp_res))

cuts_num = [i for i in range(0, len(min_cuts))]
exp_str = ""
exp_res = 0
for k in range(1, len(min_cuts) + 1):
    comb = list(combinations(cuts_num, k))
    comb = uniCompinations(comb, k, min_cuts)
    if (k % 2 != 0):
        exp_res += computeCombinations(comb, Q)
    else:
        exp_res -= computeCombinations(comb, Q)

    str_ = getStrExpression(comb, "q", "+" if (k % 2 != 0) else "-")
    exp_str += str_
exp_str = "Q =" + exp_str[2:]
print("Вероятность отказа методом минимальных путей и сечений: {}".format(exp_str))
print("Q = {}".format(exp_res))

print("Функция работоспособности: y = x1x3x5 v x1x3x6 v x1x4x5 v x1x4x6 v x2x3x5 v x2x3x6 v x2x4x5 v x2x4x6")
print("Функция неработоспособности y = !x1!x2 v !x3!x4 v !x5!x6")

```

```
print("Разрезание:  $y = x_1x_3x_5 \vee x_1x_3x_6 \vee x_1x_4x_5 \vee x_1x_4x_6 \vee x_2x_3x_5 \vee x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6 = x_1(x_3x_5 \vee x_3x_6 \vee x_4x_5 \vee x_2x_3x_5 \vee x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6) \vee !x_1(x_2x_3x_5 \vee x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6) = x_1(x_3(x_5 \vee x_6) \vee x_4x_5 \vee x_2(x_3 \vee x_4)(x_5 \vee x_6)) \vee !x_1x_2(x_3 \vee x_4)(x_5 \vee x_6) \setminus n$ ")
```

```
#####
```

```
def schemFunc(X):
    res = X[0] and X[2] and X[4] or X[0] and X[2] and X[5] or X[0] and X[3] and X[4] or X[0] and X[3] and X[5]
    res = res or X[1] and X[2] and X[4] or X[1] and X[2] and X[5] or X[1] and X[3] and X[4] or X[1] and X[3] and X[5]
    return res
```

```
def step():
    res = []
    for i in range(n):
        res.append(False if random.random() > P[i] else True)
    return res
```

```
for k in range(3):
    if (i != 0):
        P = []
        Q = []
        for i in range(0, int(n/2)):
            tmp = random.random() * 0.1
            P.append(1 - tmp)
            Q.append(tmp)
            P.append(1 - tmp)
            Q.append(tmp)
        P_tv = (1 - Q[0]**2)*(1 - Q[2]**2)*(1-Q[4]**2)
        Q_tv = 1 - P_tv
    print("Теоретически: P = {}; Q = {}".format(P_tv, Q_tv))
```

```
N = 1000000
```

```
ok = 0
```

```
for i in range(N):
    X = step()
    if (schemFunc(X)):
        ok += 1
```

```
print("Вероятность безотказной работы в результате эксперимента: ", ok/N)
```

```
print("Вероятность отказа в результате эксперимента: ", (N-ok)/N, "\n")
```