# Лабораторная работа №11-12

Студент: Болдинов Алексей ЭФМО-02-24

Сайт:<https://mathhelpplanet.com/static.php?p=smo-s-otkazami>

**Пример 5. (с сайта)**

Решение на Питоне:

def primer5():  
 lam = 90 # Заявок в час  
 t\_ob = 2 # Минуты 1 обращение  
 intensive\_potoka = (1 / t\_ob) \* 60 # Интенсивность обращения 30 за час  
 print(intensive\_potoka)  
 otn\_pr\_spo\_SMO = intensive\_potoka / (lam + intensive\_potoka) # Относительная пропускная способность СМО  
 print(otn\_pr\_spo\_SMO)  
 p\_otkaza = 1 - otn\_pr\_spo\_SMO # Вероятность отказа в обслуживании составит  
 absolute = lam \* otn\_pr\_spo\_SMO # Абсолютная пропускная способность СМО,  
 # т.е. в среднем в час будут обслужены 22,5 заявки на переговоры  
 print(f"В среднем в час будут обслужены "  
 f"{absolute} заявки на переговоры. Очевидно, что при наличии только одного телефонного номера СМО будет плохо справляться с потоком заявок.")

Ответ программы:

30.0

0.25

В среднем в час будут обслужены 22.5 заявки на переговоры. Очевидно, что при наличии только одного телефонного номера СМО будет плохо справляться с потоком заявок.

Process finished with exit code 0

## Пример 6

Решение на Питоне:

def primer6():  
 import math  
  
 def calculate\_smo(arrival\_rate, service\_rate, target\_Q):  
 n = 1  
 results = []  
 while True:  
 rho = arrival\_rate / service\_rate  
 P0 = 1 / sum(rho \*\* k / math.factorial(k) for k in range(n + 1))  
 Q = 1 - (rho \*\* n / math.factorial(n)) \* P0  
 A = arrival\_rate \* Q  
 avg\_busy = A / service\_rate  
 results.append({"n": n, "Q": Q, "A": A, "avg\_busy": avg\_busy})  
 if Q >= target\_Q:  
 break  
 n += 1  
 return n, results  
  
 arrival\_rate = 90  
 service\_rate = 30  
 target\_Q = 0.9  
  
 optimal\_n, results = calculate\_smo(arrival\_rate, service\_rate, target\_Q)  
  
 print(f"Оптимальное число каналов: {optimal\_n}")  
 for row in results:  
 print(f"n={row['n']}, Q={row['Q']:.2f}, A={row['A']:.1f}, avg\_busy={row['avg\_busy']:.2f}")

Ответ программы:

Оптимальное число каналов: 6

n=1, Q=0.25, A=22.5, avg\_busy=0.75

n=2, Q=0.47, A=42.4, avg\_busy=1.41

n=3, Q=0.65, A=58.8, avg\_busy=1.96

n=4, Q=0.79, A=71.5, avg\_busy=2.38

n=5, Q=0.89, A=80.1, avg\_busy=2.67

n=6, Q=0.95, A=85.3, avg\_busy=2.84

Process finished with exit code 0

## Пример 7

Решение на Питоне:

def primer7():  
 import math  
  
 # Исходные данные  
 n = 3 # число каналов (ЭВМ)  
 arrival\_rate = 0.25 # интенсивность поступления заявок (лямбда)  
 service\_rate = 1 / 3 # интенсивность обслуживания заявок (1/t\_об)  
  
 # Расчёт интенсивности нагрузки  
 rho = arrival\_rate / service\_rate # интенсивность нагрузки  
  
 # Предельные вероятности состояний  
 p0 = 1 / sum(rho \*\* k / math.factorial(k) for k in range(n + 1))  
 p1 = rho \* p0  
 p2 = (rho \*\* 2 / math.factorial(2)) \* p0  
 p3 = (rho \*\* 3 / math.factorial(3)) \* p0  
  
 # Вероятность отказа  
 P\_otk = p3  
  
 # Относительная пропускная способность  
 Q = 1 - P\_otk  
  
 # Абсолютная пропускная способность  
 A = arrival\_rate \* Q  
  
 # Среднее число занятых каналов  
 avg\_busy = A / service\_rate  
  
 # Средняя занятость одной ЭВМ  
 avg\_utilization = avg\_busy / n \* 100  
  
 # Вывод результатов  
 print(f"Интенсивность нагрузки: ρ = {rho:.2f}")  
 print(f"Предельные вероятности состояний: p0 = {p0:.3f}, p1 = {p1:.3f}, p2 = {p2:.3f}, p3 = {p3:.3f}")  
 print(f"Вероятность отказа: P\_otk = {P\_otk:.3f}")  
 print(f"Относительная пропускная способность: Q = {Q:.3f}")  
 print(f"Абсолютная пропускная способность: A = {A:.3f}")  
 print(f"Среднее число занятых ЭВМ: avg\_busy = {avg\_busy:.3f}")  
 print(f"Средняя занятость одной ЭВМ: avg\_utilization = {avg\_utilization:.1f}%")

Ответ программы:

Интенсивность нагрузки: ρ = 0.75

Предельные вероятности состояний: p0 = 0.476, p1 = 0.357, p2 = 0.134, p3 = 0.033

Вероятность отказа: P\_otk = 0.033

Относительная пропускная способность: Q = 0.967

Абсолютная пропускная способность: A = 0.242

Среднее число занятых ЭВМ: avg\_busy = 0.725

Средняя занятость одной ЭВМ: avg\_utilization = 24.2%

Process finished with exit code 0