

Федеральное государственное бюджетное образовательное учреждение высшего образования

**"МИРЭА – Российский технологический университет"**

**РТУ МИРЭА**

Институт кибербезопасности и цифровых технологий

ОТЧЕТ

**О ВЫПОЛНЕНИИ ПРАКТИЧЕСКОЙ РАБОТЫ №3**

**«Основные понятия статистического эксперимента» по дисциплине**

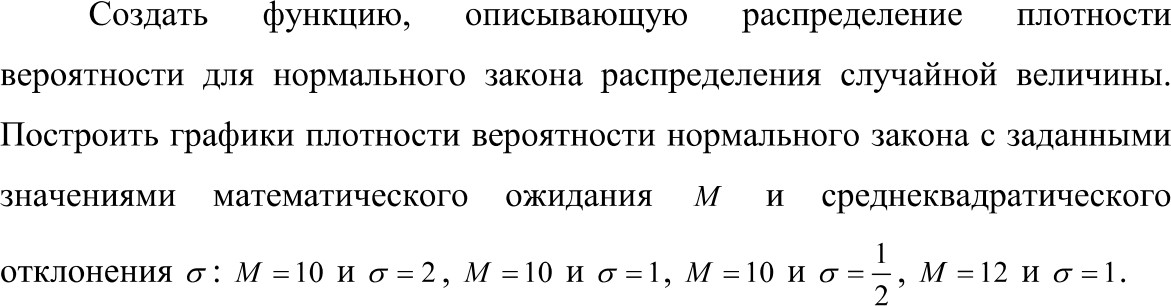
**«Моделирование систем»**

Выполнил: студент 3 курса

группы БСБО-11-21 Смирнов И.А.

Проверил: Хорсик И.А.

**Задание №1.**



**Решение:**

* param x: Значение случайной величины (может быть массивом)
* param mu: Среднее значение (математическое ожидание)
* param sigma: Стандартное отклонение
* return: Значение функции плотности вероятности для заданного x, mu и sigma

# Task 1  
def norm\_distr(x, mu, sigma):  
 coeff = 1 / (sigma \* np.sqrt(2 \* np.pi))  
 exp = np.exp(-(np.power(x - mu, 2) / (2 \* np.power(sigma, 2))))  
 prob\_dens = coeff \* exp  
 return prob\_dens  
  
  
params = [(10, 2), (10, 1), (10, 0.5), (12, 1)]  
  
x = np.linspace(0, 20, 1000)  
  
plt.figure(figsize=(12, 8))  
for mu, sigma in params:  
 y = norm\_distr(x, mu, sigma)  
 plt.plot(x, y, label=f'μ={mu}, σ={sigma}')  
  
plt.title('Плотность вероятности нормального распределения')  
plt.xlabel('Значение случайной величины')  
plt.ylabel('Плотность вероятности')  
plt.legend()  
plt.grid(True)  
plt.show()

Листинг 1. Код программы задания 1.

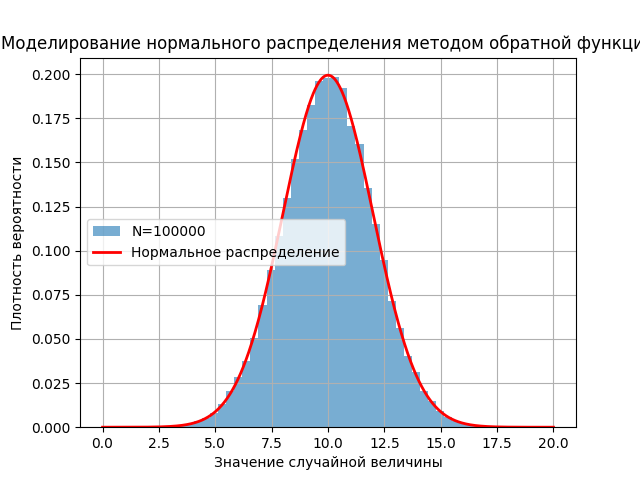
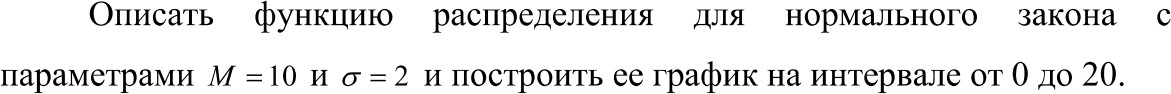


Рис. 2. Результат задания 1.

**Задание №2.**



**Решение:**

# Task 2  
  
cumul\_distr = norm.cdf(x, 10, 2)  
  
plt.figure(figsize=(8, 6))  
plt.plot(x, cumul\_distr, color='b')  
plt.title('Функция распределения нормального закона\n$\\mu=10, \\sigma=2$')  
plt.xlabel('Значение случайной величины')  
plt.ylabel('Значение функции распределения')  
plt.grid(True)  
plt.show()

Листинг 3. Код программы задания 2.

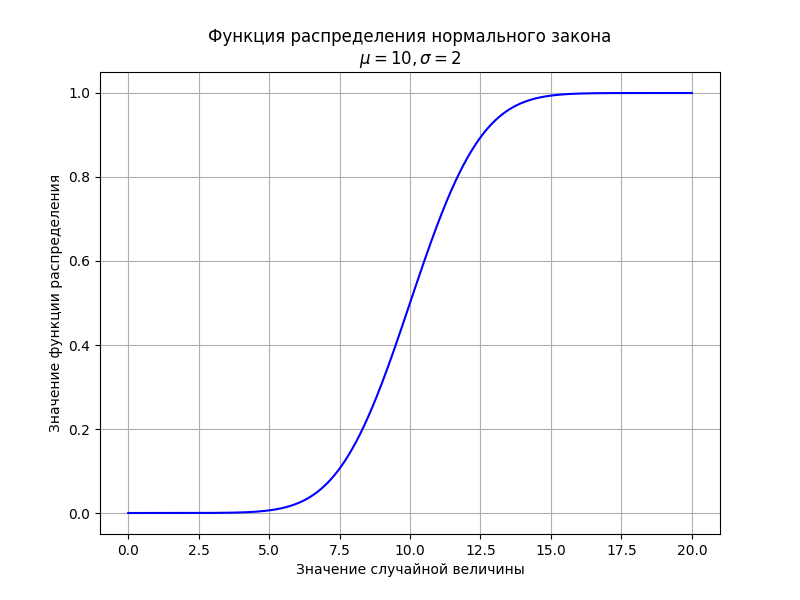
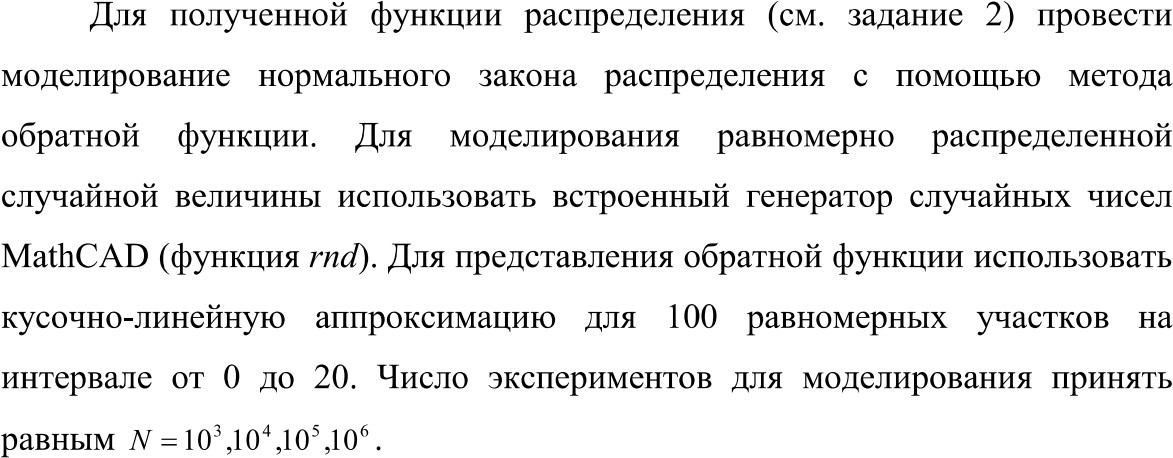


Рис. 4. Результат задания 2.

**Задание №3.**

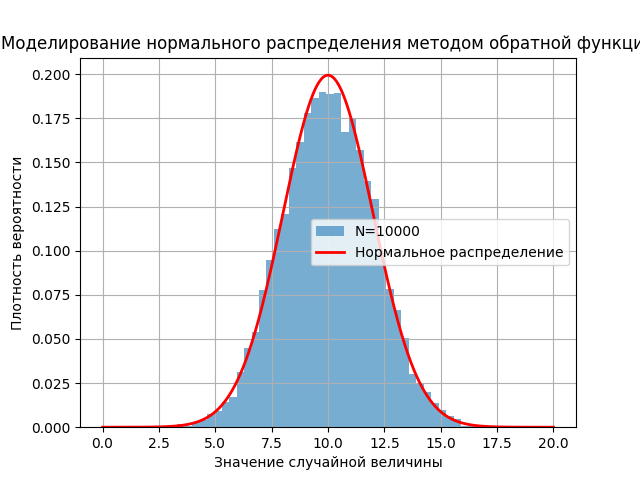
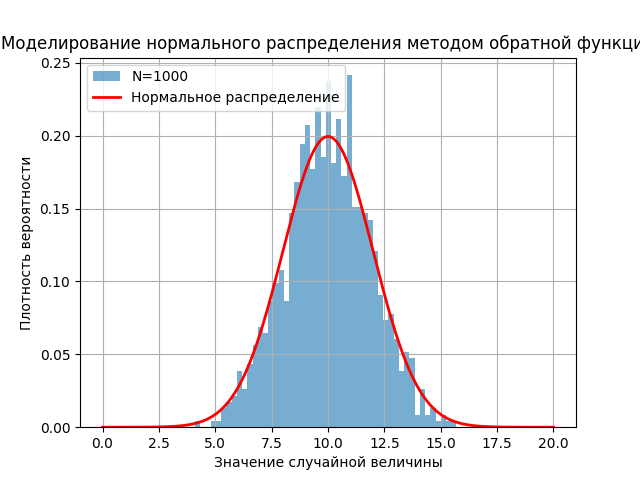


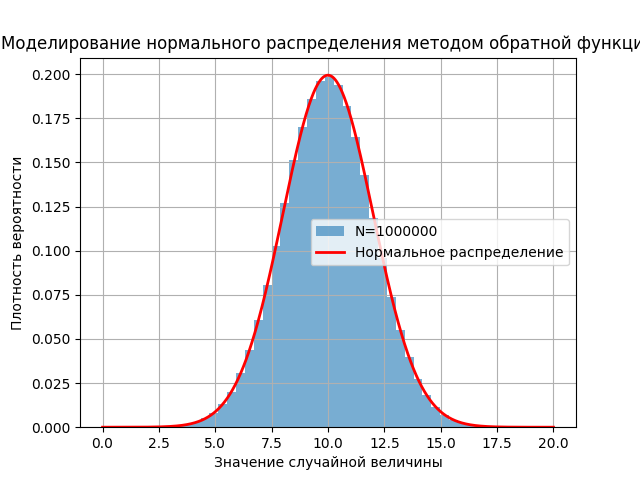
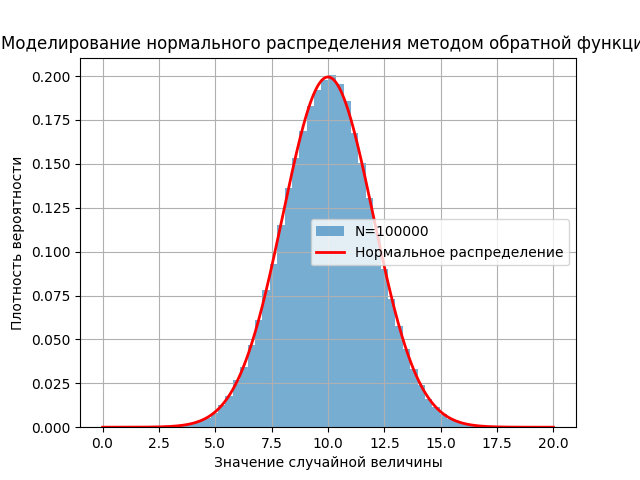
**Решение:**

# Task 3  
  
N\_values = [10 \*\* 3, 10 \*\* 4, 10 \*\* 5, 10 \*\* 6]  
  
mu = 10  
sigma = 2  
  
for N in N\_values:  
 random\_numbers = [random.uniform(0, 1) for \_ in range(N)]  
 normal\_random\_numbers = [norm.ppf(u, loc=mu, scale=sigma) for u in random\_numbers]  
  
 plt.hist(normal\_random\_numbers, bins=50, density=True, alpha=0.6, label=f'N={N}')  
  
 x = np.linspace(0, 20, 1000)  
 norm\_distr = norm.pdf(x, loc=mu, scale=sigma)  
 plt.plot(x, norm\_distr, color='red', linewidth=2, label='Нормальное распределение')  
  
 plt.title('Моделирование нормального распределения методом обратной функции')  
 plt.xlabel('Значение случайной величины')  
 plt.ylabel('Плотность вероятности')  
 plt.legend()  
 plt.grid(True)  
 plt.show()

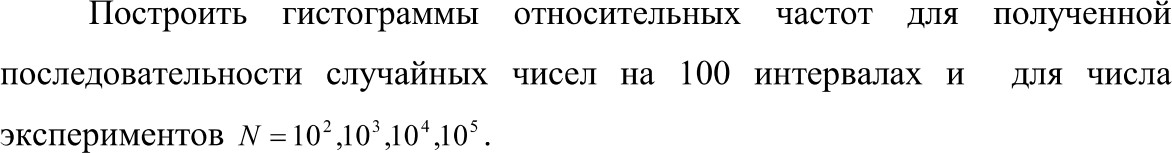
Листинг 5. Моделирование нормального закона.

Рис. 6. Результат задания 3.





**Задание №4.**



**Решение:**

# Task 4  
  
num\_bins = 100  
  
N\_values = [10 \*\* 2, 10 \*\* 3, 10 \*\* 4, 10 \*\* 5]  
  
for N in N\_values:  
 random\_numbers = [random.uniform(0, 1) for \_ in range(N)]  
 normal\_random\_numbers = [norm.ppf(u, loc=mu, scale=sigma) for u in random\_numbers]  
  
 plt.hist(normal\_random\_numbers, bins=num\_bins, density=True, alpha=0.6, label=f'N={N}')  
  
 x = np.linspace(0, 20, 1000)  
 norm\_distr = norm.pdf(x, loc=mu, scale=sigma)  
 plt.plot(x, norm\_distr, color='red', linewidth=2, label='Нормальное распределение')  
  
 plt.title('Гистограммы относительных частот для разных чисел экспериментов')  
 plt.xlabel('Значение случайной величины')  
 plt.ylabel('Относительная частота')  
 plt.legend()  
 plt.grid(True)  
 plt.show()

Листинг 7. Код решения задачи определения периода.

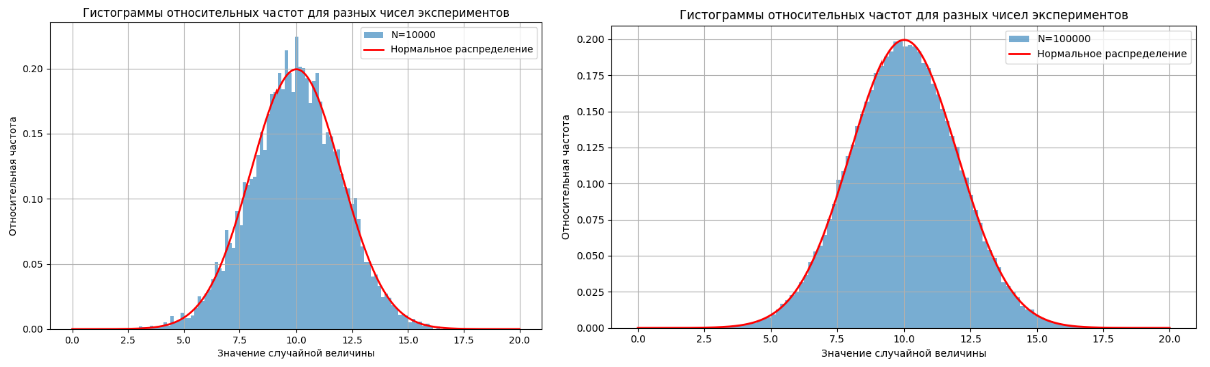
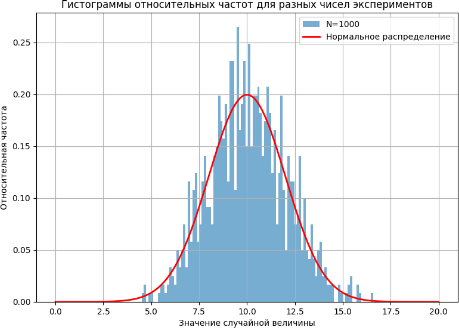
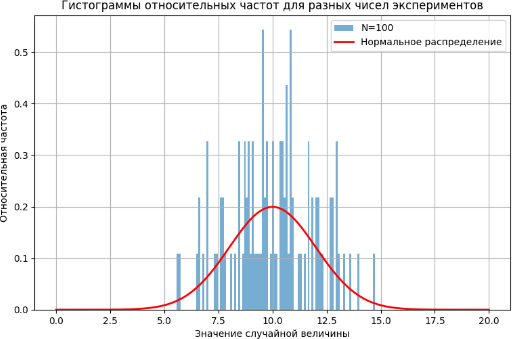
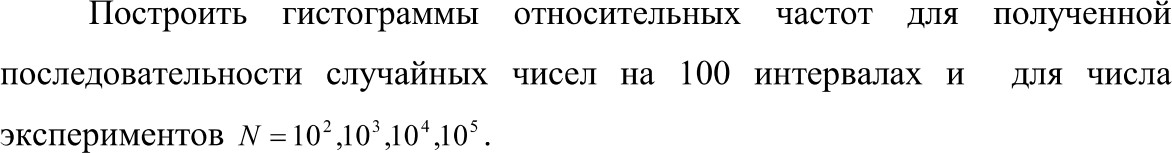


Рис. 8. Гистограммы относительных частот.

**Задание №5.**



**Решение:**

# Task 5  
  
x\_theor = np.linspace(0, 20, 1000)  
theor\_distr = norm.pdf(x\_theor, loc=mu, scale=sigma)  
  
mean\_squared\_errors = []  
for N in N\_values:  
 random\_numbers = [random.random() for \_ in range(N)]  
 normal\_random\_numbers = [norm.ppf(u, loc=mu, scale=sigma) for u in random\_numbers]  
 hist, bins = np.histogram(normal\_random\_numbers, bins=1000, density=True)  
 exp\_distr = hist \* np.diff(bins)  
 mean\_squared\_error = np.mean((theor\_distr[:1000] - exp\_distr) \*\* 2)  
 mean\_squared\_errors.append(mean\_squared\_error)  
  
plt.figure(figsize=(8, 6))  
plt.semilogx(N\_values, mean\_squared\_errors, marker='o', linestyle='-', color='b')  
plt.title('Зависимость средней квадратичной погрешности от числа экспериментов')  
plt.xlabel('Число экспериментов (логарифмическая шкала)')  
plt.ylabel('Средняя квадратичная погрешность')  
plt.grid(True)  
plt.show()

Листинг 9. Код программы задания 5.

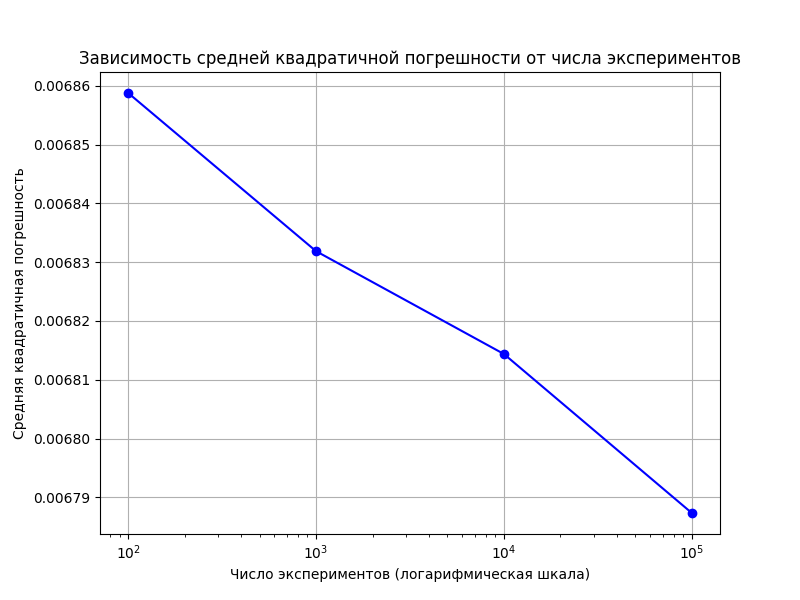


Рис. 10. Результат задания