

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра математического обеспечения и применения ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №3**  
**по дисциплине «Цифровая обработка сигналов»**  
**Тема: Исследование результатов фильтрации дискретного сигнала**

Студент гр.8382

\_\_\_\_\_

Нечепуренко Н.А.

Студент гр.8382

\_\_\_\_\_

Терехов А.Е.

Преподаватель

\_\_\_\_\_

Сучков А.И.

Санкт-Петербург

2021

## **Цели работы.**

Получение практических навыков выполнения фильтрации дискретных последовательностей с помощью рекурсивных и нерекурсивных фильтров, а также анализа получаемых результатов с помощью дискретного преобразования Фурье.

## **Основные теоретические положения.**

Лабораторная работа потребует знаний:

- в области дискретизации непрерывного сигнала;
- фильтрации дискретного сигнала с помощью дискретных нерекурсивных и рекурсивных фильтров;
- дискретного преобразования Фурье (ДПФ) для дискретных последовательностей;

и умений:

- в организации вычислительных процессов;
- в проведении компьютерных расчетов с визуализацией получаемых результатов;
- проведения анализа полученных результатов и формулировка выводов.

## **Постановка задачи.**

Для заданного дискретного сигнала применить соответствующие фильтры. Полученные результаты содержательно проинтерпретировать.

## **Порядок выполнения работы.**

1. Сформировать дискретный сигнал посредством дискретизации с шагом  $T = 1$  непрерывного сигнала, представляющего собой линейную ком-

бинацию косинусов вида  $A_k \cos(\omega_k t + \phi_k)$ . Частота каждой из гармоник не должна превышать  $\pi$ . Всего одиннадцать гармоник с упорядоченными по возрастанию частотами от 0 до  $\pi$ , изменяющимися с шагом  $\Delta\omega = 0.1\pi$ . Амплитуды гармоник  $A_k$  представляют собой целые числа со значениями от 1 до 11, определяемые случайным образом с помощью датчика равномерно распределенных случайных чисел. При необходимости нормализовать коэффициенты линейной комбинации посредством деления их на сумму полученных случайным образом амплитуд. Начальные фазы  $\phi_k$  представляют собой случайные числа в промежутке от 0 до 0.5. Дискретная последовательность должна включать в себя 32 отсчета ( $N = 31$ ).

2. Визуализировать исходные аналоговый и дискретизированный сигналы.
3. С помощью ДПФ найти дискретные отсчеты спектра дискретного сигнала и визуализировать их.
4. Для дискретного сигнала применить линейное сглаживание по 5-ти и 9-ти точкам, представить формулу для  $H(\omega)$  – передаточной функции (частотной характеристики) фильтра.
5. Визуализировать полученный после фильтрации дискретный сигнал совместно с исходным дискретным сигналом.
6. С помощью ДПФ найти дискретные отсчеты спектра дискретного сигнала после его фильтрации и визуализировать их совместно с отсчетами спектра исходного дискретного сигнала.
7. Проанализировать результат на соответствие значениям  $H(\omega)$ . Сделать выводы.
8. Повторить пункты 4)-7) для следующих фильтров:
  - Сглаживание полиномом 2-ой степени по 5 и 9 узлам.

- Сглаживание полиномом 4-ой степени по 7 и 11 узлам.
- Дискретный фильтр, соответствующий численному дифференцированию 1-го порядка.
- Дискретный фильтр, соответствующий численному интегрированию (прямоугольников, трапеций, Симпсона).

9. Содержательно проинтерпретировать результаты выполнения лабораторной работы, сделать выводы.

### **Генерация сигнала.**

Для выполнения работы был сгенерирован дискретный сигнал согласно п.1 порядка выполнения работы. График дискретного сигнала с шагом 1 и график соответствующего ему аналогового сигнала представлены на рисунке 1.



Рисунок 1 – Аналоговый и дискретный сигналы

Аналоговый сигнал является периодическим, период равен 20. Соответственно, дискретный сигнал тоже периодический.

Спектр дискретного сигнала  $s(t)$  тоже периодический, а значит может быть разложен в ряд Фурье.

$$s(t) = \sum_{k=-\infty}^{+\infty} x_k \delta(t - kT)$$

Рассмотрим часть сигнала, равную 1 периоду ( $T = 0..20$ ) и применим к ней дискретное преобразование Фурье. График дискретных отсчетов спектра дискретного сигнала приведен на рисунке 2.



Рисунок 2 – Дискретные отсчеты спектра дискретного сигнала

### Сглаживание полиномом первой степени.

Выполним сглаживание дискретного сигнала с помощью полинома 1 степени по 5 и 9 точкам.

Передаточные функции фильтра имеют вид:

$$H_5(\omega) = \frac{1}{5}(1 + 2\cos\omega + 2\cos 2\omega)$$

$$H_9(\omega) = \frac{1}{9}(1 + 2\cos\omega + 2\cos 2\omega + 2\cos 3\omega + 2\cos 4\omega)$$

Формулы сглаживания:

$$y_5(nT) = \frac{1}{5} \sum_{k=n-2}^{n+2} x_k$$

$$y_9(nT) = \frac{1}{9} \sum_{k=n-4}^{n+4} x_k$$

График сглаженного полиномом первой степени дискретного сигнала приведен на рисунке 3.

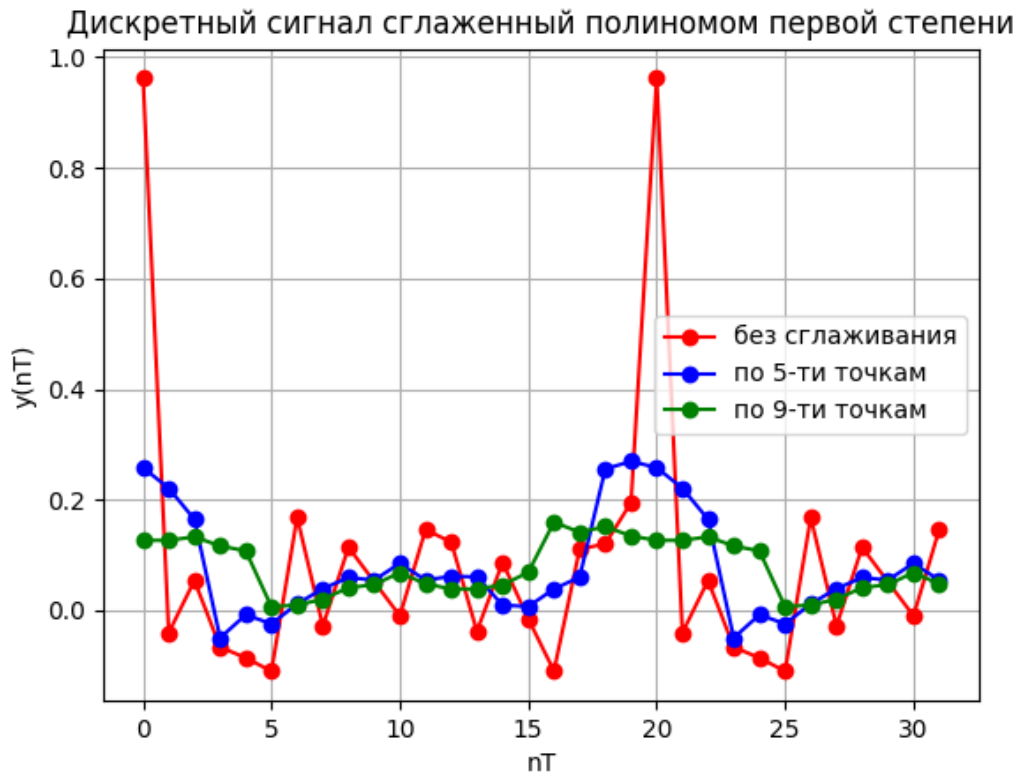


Рисунок 3 – Дискретный сигнал, сглаженный полиномом первой степени по 5-ти и 9-ти точкам

Пик в нуле был сглажен, визуально заметно, что сглаженный сигнал представляет собой линейное среднее.

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и построим отсчетов график совместно с отсчетами спектра исходного дискретного сигнала (см. рис. 4).

Дискретные отсчёты спектра после сглаживания полиномом 1 степени

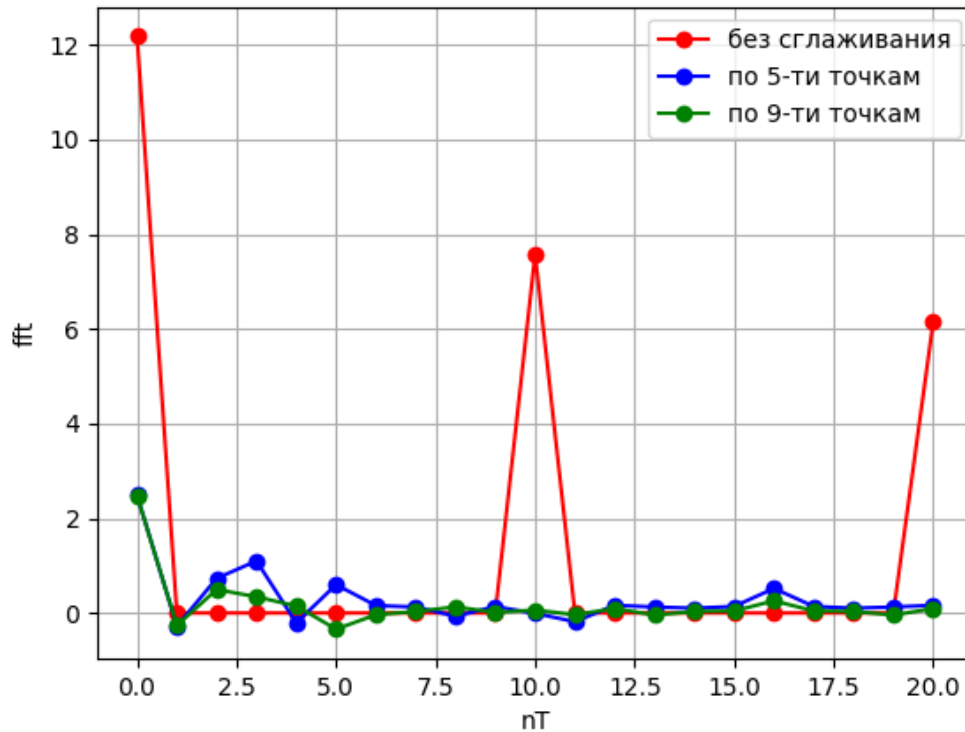


Рисунок 4 – Дискретные отсчёты спектра после фильтрации полиномом 1 степени

### Сглаживание полиномом второй степени.

Выполним сглаживание дискретного сигнала с помощью полинома 2 степени по 5 и 9 точкам.

Передаточные функции фильтра имеют вид:

$$H_5(\omega) = \frac{1}{35}(17 + 24\cos\omega - 6\cos 2\omega)$$

$$H_9(\omega) = \frac{1}{231}(59 + 108\cos\omega + 78\cos 2\omega + 28\cos 3\omega - 42\cos 4\omega)$$

Формулы сглаживания:

$$y_5(nT) = \frac{1}{35}(-3x_{n-2} + 12x_{n-1} + 17x_n + 12x_{n+1} - 3x_{n+2})$$



$$y_9(nT) = \frac{1}{231}(-21x_{n-4} + 14x_{n-3} + 39x_{n-2} + 54x_{n-1} + 59x_n + 54x_{n+1} + 39x_{n+2} + 14x_{n+3} - 21x_{n+4})$$

График сглаженного полиномом второй степени дискретного сигнала приведен на рисунке 5.



Рисунок 5 – Дискретный сигнал, сглаженный полиномом второй степени по 5-ти и 9-ти точкам

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и построим отсчетов график совместно с отсчетами спектра исходного дискретного сигнала (см. рис. 6).

Дискретные отсчёты спектра после сглаживания полиномом 2 степени

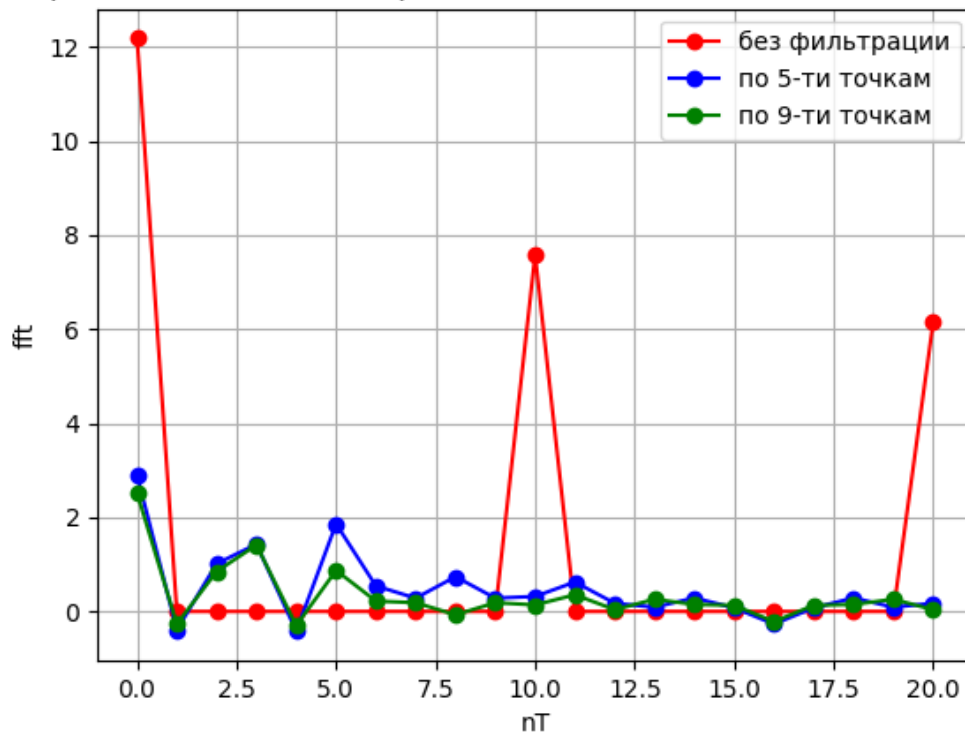


Рисунок 6 – Дискретные отсчёты спектра после фильтрации полиномом 2 степени

### Сглаживание полиномом четвертой степени.

Выполним сглаживание дискретного сигнала с помощью полинома 4 степени по 7 и 11 точкам.

Передаточные функции фильтра имеют вид:

$$H_7(\omega) = \frac{1}{231}(131 + 150\cos\omega - 60\cos 2\omega + 10\cos 3\omega)$$

$$H_9(\omega) = \frac{1}{429}(143 + 240\cos\omega + 120\cos 2\omega - 20\cos 3\omega - 90\cos 4\omega + 26\cos 5\omega)$$

Формулы сглаживания:

$$y_7(nT) = \frac{1}{231}(5x_{n-3} - 30x_{n-2} + 75x_{n-1} + 131x_n + 75x_{n+1} - 30x_{n+2} + 5x_{n+3})$$

$$y_{11}(nT) = \frac{1}{429}(13x_{n-5} - 45x_{n-4} - 10x_{n-3} + 60x_{n-2} + 120x_{n-1} + 143x_n + 120x_{n+1} + 60x_{n+2} - 10x_{n+3} - 45x_{n+4} + 13x_{n+5})$$

График сглаженного полиномом четвертой степени дискретного сигнала приведен на рисунке 7.



Рисунок 7 – Дискретный сигнал, сглаженный полиномом четвертой степени по 7-ти и 11-ти точкам

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сиг-

нала после его фильтрации и построим отсчетов график совместно с отсчетами спектра исходного дискретного сигнала (см. рис. 8).



Рисунок 8 – Дискретные отсчёты спектра после фильтрации полиномом 4 степени

**Дискретный фильтр, соответствующий численному дифференцированию 1-го порядка.**

Применим к сигналу дискретный фильтр, соответствующий численному дифференцированию 1-го порядка.

Формула для полученного сигнала:

$$y(nT) = \frac{1}{2}(x_{n+1} - x_{n-1})$$

Передаточная функция фильтра имеет вид:

$$H(\omega) = i \sin \omega$$

График полученного сигнала приведен на рисунке 9.



Рисунок 9 – Применение фильтра, соответствующего численному дифференцированию первого порядка.

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и построим отсчетов график совместно с отсчетами спектра исходного дискретного сигнала (см. рис. 10).



Рисунок 10 – Дискретные отсчёты спектра, фильтр численного дифференцирования первого порядка

**Дискретный фильтр, соответствующий численному интегрированию (прямоугольников, трапеций, Симпсона).**

Для дискретного сигнала применим дискретный фильтр, соответствующий численному интегрированию (средних прямоугольников  $y_1(n)$ , трапеций  $y_2(n)$ , Симпсона  $y_3(n)$ ):

$$y_1(n) = y_1(n-1) + x_{n+\frac{1}{2}}$$

$$y_2(n) = y_2(n-1) + \frac{1}{2}(x_n + x_{n+1})$$

$$y_3(n+1) = y_3(n-1) + \frac{1}{3}(x_{n-1} + 4x_n + x_{n+1})$$

Соответствующие им передаточные функции:

$$H_1(\omega) = \frac{1}{2i \sin \frac{\omega}{2}}$$

$$H_2(\omega) = \frac{\cos \frac{\omega}{2}}{2i \sin \frac{\omega}{2}}$$

$$H_3(\omega) = \frac{\cos \omega + 2}{3i \sin \omega}$$

Дискретный сигнал после применения к нему фильтров, соответствующих численному интегрированию предоставлен на рис. 11.



Рисунок 11 – Дискретный сигнал после применения к нему фильтров, соответствующих численному интегрированию

С помощью ДПФ найдём дискретные отсчеты спектра дискретного сигнала после его фильтрации и построим их графики (см. рис. 12).



Рисунок 12 – Дискретные отсчёты спектра, фильтры численного интегрирования



## **Выводы.**

В результате выполнения работы были получены практические навыки фильтрации входных дискретных сигналов с помощью нерекурсивных и рекурсивных фильтров. Были применены: сглаживание сигнала полиномом первой степени по 5 и 9 точкам, полиномом второй степени по 5 и 9 точкам, полиномом четвертой степени по 7 и 11 точкам, а также с помощью фильтра численного дифференцирования первого порядка и фильтров численного интегрирования методом прямоугольников, трапеций, Симпсона. Результаты применения каждого фильтра были отображены графически вместе с исходным сигналом. Для каждого фильтра с помощью ДПФ был построен график дискретных отсчетов спектра дискретного сигнала.

## ПРИЛОЖЕНИЕ А. ИСХОДНЫЙ КОД ПРОГРАММЫ.

```
import math

import matplotlib.pyplot as plt
import numpy as np
from scipy.fft import fft

np.random.seed(8382)
n = 31
d_omega = 0.1 * math.pi
A = np.random.randint(1, 11, 11)
omega = np.array([d_omega * i for i in range(0, 11)])
phi = np.random.rand(11) / 2
discrete_time = np.arange(0, n + 1)
time = np.linspace(0, n + 1, 1000)
d_time_period = np.arange(0, 21)

# signal
xT = np.array([np.sum(A * np.cos(omega * x + phi)) for x in time
               ]) / np.sum(A)
xnT = np.array([np.sum(A * np.cos(omega * x + phi)) for x in
               discrete_time]) / np.sum(A)
print("xnT: ", xnT)
plt.titleАналоговый(" и дискретный сигнал с шагом дискретизации T
               =1")
plt.xlabel("t")
plt.ylabel("x")
plt.grid(True)
plt.plot(time, xT, "b", labelАналоговый=" сигнал")
plt.plot(discrete_time, xnT, "ro-", labelДискретный=" сигнал")
plt.legend(loc="best")
plt.show()
```

```

# fft
d = 100
xnT = np.array([np.sum(A * np.cos(omega * x + phi)) for x in np.
    arange(-d, d)]) / np.sum(A)
plt.titleДискретные(" отсчетыспектрадискретногосигнала ")
plt.xlabel("nT")
plt.ylabel("fft")
plt.grid(True)
plt.plot(d_time_period, fft(xnT)[d:21 + d], "bo-")
plt.show()

# smooth 1 deg
shift = 2
smooth5_1 = np.array([np.sum(xnT[d + i - shift:d + i + shift +
    1]) for i in discrete_time]) / 5
shift = 4
smooth9_1 = np.array([np.sum(xnT[d + i - shift:d + i + shift +
    1]) for i in discrete_time]) / 9
plt.titleДискретный(" сигналсглаженныйполиномомпервойстепени
    ")
plt.xlabel("nT")
plt.ylabel("y(nT)")
plt.grid(True)
plt.plot(discrete_time, xnT[d:32 + d], "ro-", labelбез="
    сглаживания")
plt.plot(discrete_time, smooth5_1, "bo-", labelпо=" ти5- точкам")
plt.plot(discrete_time, smooth9_1, "go-", labelпо=" ти9- точкам")
plt.legend(loc="best")
plt.show()

plt.titleДискретные("

```

```

    отсчёты спектра после сглаживания полиномом 1 степени")
plt.xlabel("nT")
plt.ylabel("fft")
plt.grid(True)
plt.plot(d_time_period, fft(xnT)[d:21 + d], "ro-", label="без="
        "сглаживания")
plt.plot(d_time_period, fft(smooth5_1)[:21], "bo-", label="по="
        "ти5-точкам")
plt.plot(d_time_period, fft(smooth9_1)[:21], "go-", label="по="
        "ти9-точкам")
plt.legend(loc="best")
plt.show()

# smooth 2 deg
coef_5 = np.array([-3, 12, 17, 12, 3])
shift = 2
smooth5_2 = np.array([np.sum(coef_5 * xnT[d + i - shift:d + i +
        shift + 1]) for i in discrete_time]) / 35
coef_9 = np.array([-21, 14, 39, 54, 59, 54, 39, 14, -21])
shift = 4
smooth9_2 = np.array([np.sum(coef_9 * xnT[d + i - shift:d + i +
        shift + 1]) for i in discrete_time]) / 231
plt.title("Дискретный сигнал, сглаженный полиномом второй степени")
plt.xlabel("nT")
plt.ylabel("y(nT)")
plt.grid(True)
plt.plot(discrete_time, xnT[d:32 + d], "ro-", label="без="
        "сглаживания")
plt.plot(discrete_time, smooth5_2, "bo-", label="по=" "ти5-точкам")
plt.plot(discrete_time, smooth9_2, "go-", label="по=" "ти9-точкам")
plt.legend(loc="best")

```

```

plt.show()

# fft 2 deg
plt.titleДискретные("
    отсчётыспектрапослестглаживанияполиномом    2 степени")
plt.xlabel("nT")
plt.ylabel("fft")
plt.grid(True)
plt.plot(d_time_period, fft(xnT)[d:21 + d], "ro-", labelбез="
    фильтрации")
plt.plot(d_time_period, fft(smooth5_2)[:21], "bo-", labelпо="
    ти5- точкам")
plt.plot(d_time_period, fft(smooth9_2)[:21], "go-", labelпо="
    ти9- точкам")
plt.legend(loc="best")
plt.show()

# smooth 4 deg
coef_7 = np.array([5, -30, 75, 131, 75, -30, 5])
shift = 3
smooth_7_4 = np.array([np.sum(coef_7 * xnT[d + i - shift:d + i +
    shift + 1]) for i in discrete_time]) / 231
coef_11 = np.array([13, -45, -10, 60, 120, 143, 120, 60, -10,
    -45, 13])
shift = 5
smooth11_4 = np.array([np.sum(coef_11 * xnT[d + i - shift:d + i +
    shift + 1]) for i in discrete_time]) / 429
plt.titleДискретный("
    сигналсглаженныйполиномомчетвертойстепени    ")
plt.xlabel("nT")
plt.ylabel("y(nT)")
plt.grid(True)

```

```

plt.plot(discrete_time, xnT[d:32 + d], "ro-", label="без="
        "сглаживания")
plt.plot(discrete_time, smooth_7_4, "bo-", label="по=" "ти7-
        "точкам")
plt.plot(discrete_time, smooth11_4, "go-", label="по=" "ти11-
        "точкам")
plt.legend(loc="best")
plt.show()

# fft 4 deg
plt.title("Дискретные ("
        "отсчёты спектра после сглаживания полиномом 4 степени")
plt.xlabel("nT")
plt.ylabel("fft")
plt.grid(True)
plt.plot(d_time_period, fft(xnT)[d:21 + d], "ro-", label="без="
        "фильтрации")
plt.plot(d_time_period, fft(smooth_7_4)[:21], "bo-", label="по="
        "ти7- точкам")
plt.plot(d_time_period, fft(smooth11_4)[:21], "go-", label="по="
        "ти11- точкам")
plt.legend(loc="best")
plt.show()

# smooth deriv
smooth_deriv = np.array([xnT[d + i + 1] - xnT[d + i - 1] for i in
        discrete_time]) / 2
plt.title("Численное (" "дифференцирование 1- порядка.")
plt.xlabel("nT")
plt.ylabel("y(nT)")
plt.grid(True)
plt.plot(discrete_time, xnT[d:32 + d], "ro-", label="без="

```

```

    фильтра")
plt.plot(discrete_time, smooth_deriv, "bo-", labelc=" фильтром")
plt.legend(loc="best")
plt.show()

# fft deriv
plt.titleДискретные(" отсчётыспектра , диф. 1- порядка.")
plt.xlabel("nT")
plt.ylabel("fft")
plt.grid(True)
plt.plot(d_time_period, fft(xnT)[d:21 + d], "ro-", labelбез="
    фильтрации")
plt.plot(d_time_period, fft(smooth_deriv)[:21], "bo-", labelc="
    фильтром")
plt.legend(loc="best")
plt.show()

# smooth integral
rect = np.zeros(32)
rect[0] = xnT[1 + d]
for i in range(1, 32):
    rect[i] = rect[i - 1] + xnT[i + 1 + d]
trap = np.zeros(32)
trap[0] = (xnT[0 + d] + xnT[1 + d]) / 2
for i in range(1, 32):
    trap[i] = trap[i - 1] + (xnT[i + d] + xnT[i + 1 + d]) / 2
trap[31] = trap[30] + xnT[31]
simp = np.zeros(32)
simp[0] = (xnT[0 - 1 + d] + 4 * xnT[0 + d] + xnT[0 + 1 + d]) / 3
for i in range(1, 32):
    simp[i] = simp[i - 1] + (xnT[i - 1 + d] + 4 * xnT[i + d] +
        xnT[i + 1 + d]) / 3

```

```

simp[31] = simp[30] + xnT[30 + d]

plt.titleДискретный(" сигналпоследичисленногоинтегрирования .")
plt.xlabel("nT")
plt.ylabel("y(nT) ")
plt.grid(True)
plt.plot(discrete_time, xnT[d:32 + d], "ro-", labelбез="
    фильтра")
plt.plot(discrete_time, rect, "bo-", labelметод="
    прямоугольников")
plt.plot(discrete_time, trap, "go-", labelметод=" трапеций")
plt.plot(discrete_time, simp, "ko-", labelметод=" Симпсона")
plt.legend(loc="best")
plt.show()

# fft integral
plt.titleДискретные("
    отсчётыспектрапоследичисленногоинтегрирования ")
plt.xlabel("nT")
plt.ylabel("fft")
plt.grid(True)
plt.plot(d_time_period[1:], fft(xnT)[d:21 + d][1:], "ro-", label
    без=" фильтрации")
plt.plot(d_time_period[1:], fft(rect)[:21][1:], "bo-", label
    метод=" прямоугольников")
plt.plot(d_time_period[1:], fft(trap)[:21][1:], "go-", label
    метод=" трапеций")
plt.plot(d_time_period[1:], fft(simp)[:21][1:], "ko-", label
    метод=" Симпсона")
plt.legend(loc="best")
plt.show()

```