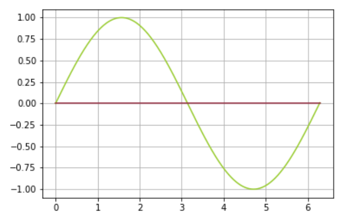
$$f(x) = sin(x), \quad x \in [0, 2\pi]$$

$$M_{n+1} = 1$$

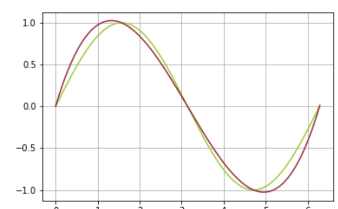
$$x = 0.1$$

Лагранж на равномерной сетке:

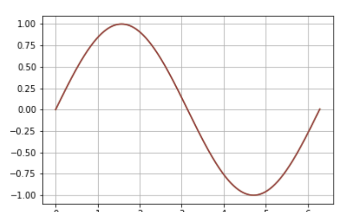
Кол-во узлов: 3
Значение интерполяции в точке: 1.1446350171091306e-17
Приближенная погрешность: -0.31344551676836296
Реальная погрешность: 0.09983341664682814



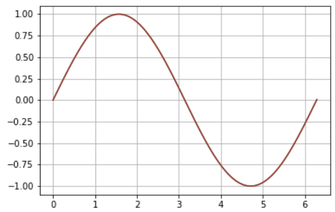
Кол-во узлов: 5
Значение интерполяции в точке: 0.16174558203178985
Приближенная погрешность: -0.10631891337303637
Реальная погрешность: 0.06191216538496169



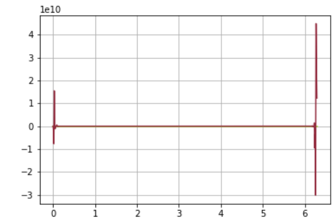
Кол-во узлов: 10
Значение интерполяции в точке: 0.09977432239477409
Приближенная погрешность: -0.00025824186005998865
Реальная погрешность: 5.909425205406771e-05



Кол-во узлов: 20
Значение интерполяции в точке: 0.09983341664676312
Приближенная погрешность: -1.1598809615938782e-12
Реальная погрешность: 6.503131366741854e-14



Кол-во узлов: 100
Значение интерполяции в точке: 49173091.561414436
Приближенная погрешность: 5.6218786500410605e-126
Реальная погрешность: 49173091.46158102

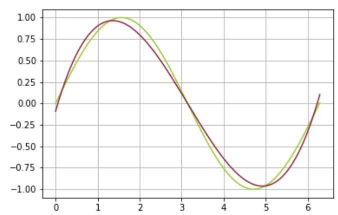


Кол-во узлов: 200
Значение интерполяции в точке: -3.726547082314121e+33
Приближенная погрешность: 0.0
Реальная погрешность: 3.726547082314121e+33

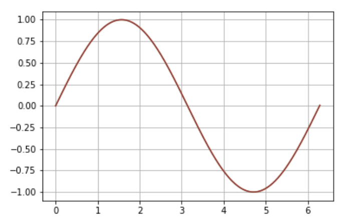
Лагранж в узлах Чебышева:

Кол-во узлов: 3
Значение интерполяции в точке: 0.45676587070096786
Приближенная погрешность: 0.937359326572923
Реальная погрешность: 0.3569324540541397

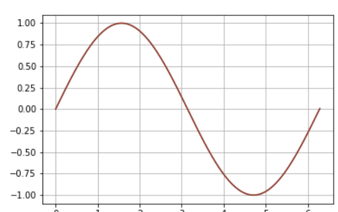
Кол-во узлов: 5
Значение интерполяции в точке: 0.0719564654489074
Приближенная погрешность: 0.047992948961276996
Реальная погрешность: 0.02787695119792076



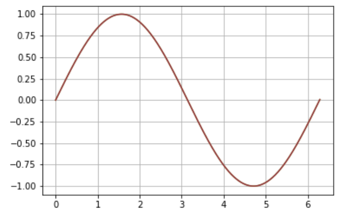
Кол-во узлов: 10
Значение интерполяции в точке: 0.099824255496215
Приближенная погрешность: -4.1263913515963566e-05
Реальная погрешность: 9.161150613154478e-06



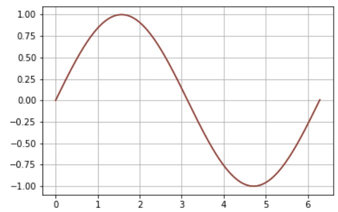
Кол-во узлов: 20
Значение интерполяции в точке: 0.09983341664682792
Приближенная погрешность: 2.340517650879222e-15
Реальная погрешность: 2.3592239273284576e-16



Кол-во узлов: 100
Значение интерполяции в точке: 0.09983341664682839
Приближенная погрешность: 8.649737630328156e-139
Реальная погрешность: 2.3592239273284576e-16



Кол-во узлов: 200
Значение интерполяции в точке: 0.09983341664682865
Приближенная погрешность: 0.0
Реальная погрешность: 4.996003610813204e-16



1 Лагранж на равномерной сетке

```
import numpy as np
import matplotlib.pyplot as plt

def f(x):
    return np.sin(x)

def factorial(k):
    j = 1
    for i in range(2, k+1):
        j = j*i
    return j

def lagranz(a, b, x):

    n = len(a)
    basic_polynomials = np.ones(n)

    inter_pol = 0

    for i in range(1, n-1):
        for j in range(i, n-1):
            basic_polynomials[i] = basic_polynomials[i] * \
            ((x - a[j + 1]) / (a[i] - a[j + 1]))

    for i in range(1, n):
        for j in range(i):
            basic_polynomials[i] = basic_polynomials[i] * \
            ((x - a[j]) / (a[i] - a[j]))

    for i in range(n-1):
        basic_polynomials[0] = basic_polynomials[0] * \
        ((x - a[i + 1]) / (a[0] - a[i + 1]))

    for i in range(n):
        inter_pol += b[i] * basic_polynomials[i]

    return inter_pol

def approximate_error(a, Mn, x):
    n = len(a)
    e = Mn/factorial(n)

    for i in range(n):
        e*=(a[i]-x)

    return e
```

```
Mn = 1
x_point = 0.1

A = 0
B = 2*np.pi

n_nod = 3
st = (np.abs(A) + np.abs(B))/(n_nod-1)
a = np.arange(A, B+st, st)
b = f(a)

print('Кол-во узлов: ', n_nod)
print('Значение интерполяции в точке: ', lagranz(a, b, x_point))
print('Приближенная погрешность:', approximate_error(a, Mn, x_point))
print('Реальная погрешность: ', np.abs(f(x_point) - lagranz(a, b, x_point)))


# графики
shag = 0.01
x = np.arange(A, B+shag, shag)
y = f(x)
fig = plt.figure()
plt.plot(x, y, color='#9acd32')
y_intro = []
for i in x:
    y_intro.append(lagranz(a, b, i))
plt.plot(x, y_intro, color='#922b3e')
plt.grid(True)
plt.show()
```

2 Лагранж в узлах Чебышева

```
import numpy as np
import matplotlib.pyplot as plt

def f(x):
    return np.sin(x)

def factorial(k):
    j = 1
    for i in range(2, k+1):
        j = j*i
    return j

def lagranz(a, b, x):

    n = len(a)
    basic_polynomials = np.ones(n)

    inter_pol = 0

    for i in range(1, n-1):
        for j in range(i, n-1):
            basic_polynomials[i] = basic_polynomials[i] * \
            ((x - a[j + 1]) / (a[i] - a[j + 1]))

    for i in range(1, n):
        for j in range(i):
            basic_polynomials[i] = basic_polynomials[i] * \
            ((x - a[j]) / (a[i] - a[j]))

    for i in range(n-1):
        basic_polynomials[0] = basic_polynomials[0] * \
        ((x - a[i + 1]) / (a[0] - a[i + 1]))

    for i in range(n):
        inter_pol += b[i] * basic_polynomials[i]

    return inter_pol

def approximate_error(a, Mn, x):
    n = len(a)
    e = Mn/factorial(n)

    for i in range(n):
        e=e*(a[i]-x)

    return e
```

```python
Mn = 1
x_point = 0.1

A = 0
B = 2*np.pi

n_nod = 3
k = np.arange(1, n_nod + 1)
a =  (A + B) / 2 + ((B - A) / 2) * np.cos((2*k - 1)*np.pi/(2 * len(k)))
b = f(a)

print('Кол-во узлов: ', n_nod)
print('Значение интерполяции в точке: ', lagranz(a, b, x_point))
print('Приближенная погрешность:', approximate_error(a, Mn, x_point))
print('Реальная погрешность: ', np.abs(f(x_point) - lagranz(a, b, x_point)))

# графики
shag = 0.01
x = np.arange(A, B+shag, shag)
y = f(x)
fig = plt.figure()
plt.plot(x, y, color='#9acd32')
y_intro = []
for i in x:
    y_intro.append(lagranz(a, b, i))
plt.plot(x, y_intro, color='#922b3e')
plt.grid(True)
plt.show()
```