**Задание 1, вариант 1**

Листинг программы:

# -\*- coding: utf-8 -\*-

from matplotlib import pyplot as plt

from numpy import arange

step = 0.1

start = -2.0

stop = 2.0

# Рассчитаем точки

x = arange(start, stop+step, step)

y = x\*\*2

# Построим график зеленого цвета сплошной линией

# и с круглыми маркерами

plt.plot(x, y, 'go-', label='line 1', linewidth=2)

plt.grid() # Сетка

plt.xlabel('x') # Подпись оси X

plt.ylabel('y') # Подпись оси Y

plt.xlim([start-step, stop+step]) # Ограничение по оси X

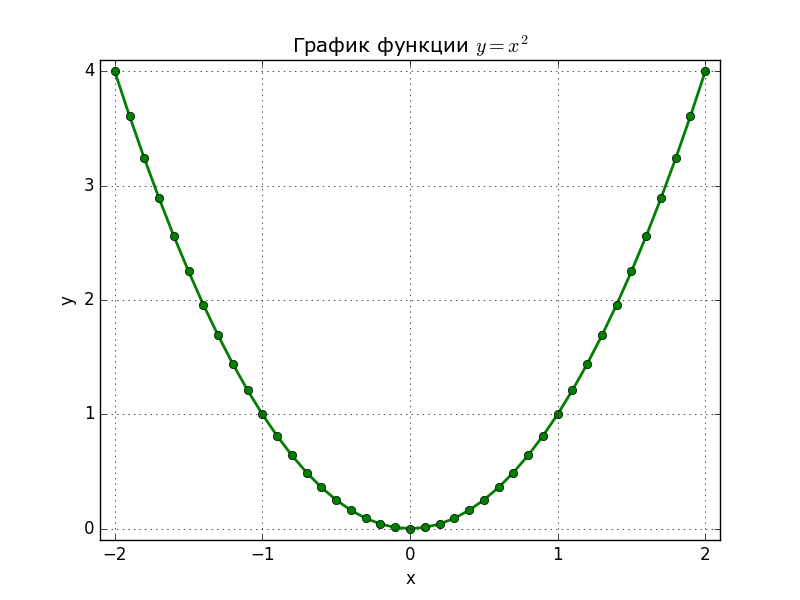
plt.ylim([min(y)-step, max(y)+step]) # Ограничение по оси Y

plt.title('График функции $y=x^2$', family='verdana') # Подпись

plt.savefig('fig1.png') # Сохраним график

plt.show() # Отобразим график

Результат работы:

****

**Задание 2, вариант 1**

Листинг программы:

# -\*- coding: utf-8 -\*-

from matplotlib import pyplot as plt

from numpy import arange, ones, zeros

from numpy.polynomial.polynomial import polyval

from matplotlib import rc

# Чтобы нормально отображался русский язык

rc('font', family='Verdana', weight='normal')

k\_list = range(1,7) # Степени многочлена 1, 2, ... , 6

step = 0.1

start = -1

stop = 1

# Рассчитаем точки

x = arange(start, stop+step, step)

y = zeros((len(k\_list), len(x))) # Заполним нулями

label\_list = ['$1$' for i in range(1,7)] # Заполним строками, $ нужен для отображение в LaTEX

plt.figure()

for i, k in enumerate(k\_list):

c = ones(k+1) # Задаем единичные коэффициенты

y[i, :] = polyval(x, c) # Полином по точкам x и коэффициентам c

label\_list[i] = '$'+label\_list[i-1][1:-1]+'+x^%d$'%(i+1) # Подпись для легенды

plt.plot(x, y[i, :], label=label\_list[i]) # Строим график

plt.grid() # Отобразим сетку

plt.legend(loc='best') # Отобразим легенду с наилучшим расположением

plt.xlabel('x') # Подпись оси X

plt.ylabel('y') # Подпись оси Y

plt.xlim([start-step, stop+step]) # Ограничение по оси X

plt.savefig('fig2.png') # Сохраним график

plt.show() # Отобразим график

# Построим кривые каждую в отдельном графике

# В 1 столбец

plt.figure(figsize=(10, 10))

for i in range(6):

plt.subplot(6, 1, i+1)

plt.plot(x, y[i, :]) # Строим график

plt.title(label\_list[i])

plt.grid()

plt.savefig('fig3.png') # Сохраним график

plt.show()

# В 2 столбца

plt.figure(figsize=(10, 10))

for i, k in enumerate(k\_list):

plt.subplot(3, 2, i+1)

plt.plot(x, y[i, :]) # Строим график

plt.title(label\_list[i])

plt.grid()

plt.savefig('fig4.png') # Сохраним график

plt.show()

# В 3 столбца

plt.figure(figsize=(10, 10))

for i, k in enumerate(k\_list):

plt.subplot(2, 3, i+1)

plt.plot(x, y[i, :]) # Строим график

plt.title(label\_list[i])

plt.grid()

plt.savefig('fig5.png') # Сохраним график

plt.show()

# В 1 строку

plt.figure(figsize=(20, 10))

for i, k in enumerate(k\_list):

plt.subplot(1, 6, i+1)

plt.plot(x, y[i, :]) # Строим график

plt.title(label\_list[i])

plt.grid()

plt.savefig('fig6.png') # Сохраним график

plt.show()

Результат работы:

