**Задача 3**. Цилиндрический сосуд высотой 6 метров и диаметром 4 метра имеет на дне круглое отверстие радиусом 0,1 метра. В начальный момент времени  сосуд доверху наполнен водой. Известно, что зависимость уровня воды  в сосуде от времени определяется дифференциальным уравнением [2]

,

где  – коэффициент скорости истечения жидкости из отверстия,  – площадь отверстия на дне сосуда,  – площадь горизонтального сечения сосуда, . Требуется определить время, в течение которого уровень воды в сосуде станет равным 0,1 метра.

**Код:**

# Вариант 3  
import math  
  
import matplotlib.pyplot as plt  
  
VESSEL\_HEIGHT = 6 # высота сосуда  
VESSEL\_DIAMETER = 4 # диаметр сосуда  
HOLE\_RADIUS = 0.1 # радиус круглого отверстия на дне  
k = 0.6 # коэффициент скорости истечения жидкости из отверстия  
g = 9.8 # ускорение свободного падения  
  
  
def circle\_area(r: float): # площадь круга  
 return math.pi \* r \*\* 2  
  
  
s = circle\_area(HOLE\_RADIUS) # площадь отверстия на дне сосуда  
S = circle\_area(VESSEL\_DIAMETER / 2) # площадь горизонтального сечения сосуда  
  
# Требуется определить время, в течение которого уровень воды в сосуде станет  
# равным 0.1 метра  
RES = 0.1  
  
h = 0.01 # временной шаг  
  
  
def f(x: float):  
 return -k \* s \* math.sqrt(2 \* g \* x) / S  
  
  
def euler(x, t):  
 x\_array.append(x)  
 t\_array.append(t)  
 while x > RES:  
 x = x + h \* f(x)  
 x\_array.append(x)  
 t = t + h  
 t\_array.append(t)  
 return t  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 print(f'Коэффициент скорости истечения жидкости из отверстия: {k}')  
 print(f'Ускорение свободного падения: {g} м/с^2\n')  
 print(f'Высота сосуда: {VESSEL\_HEIGHT} м')  
 print(f'Диаметр сосуда: {VESSEL\_DIAMETER} м')  
 print('В начальный момент времени сосуд доверху наполнен водой\n')  
 print(f'Площадь отверстия на дне сосуда: {s} м^2')  
 print(f'Площадь горизонтального сечения сосуда: {S} м^2\n')  
  
 x\_array = []  
 t\_array = []  
 res\_time = euler(VESSEL\_HEIGHT, 0)  
  
 print(f'Уровень воды достигнет {RES} м через {round(res\_time, 2)} с')  
  
 plt.xlabel('Время, с')  
 plt.ylabel('Уровень воды, м')  
 plt.grid()  
 plt.plot(t\_array, x\_array)  
 plt.show()

**Результат:**



