

```
# Исследуйте функции:
# 1. Определить корни
# 2. Найти интервалы, на которых функция возрастает
# 3. Найти интервалы, на которых функция убывает
# 4. Построить график
# 5. Вычислить вершину
# 6. Определить промежутки, на которых  $fx > 0$ 
# 7. Определить промежутки, на которых  $fx < 0$ 
```

```
# Функция 1:
#  $f = -18 * x ** 3 + 5 * x ** 2 + 10 * x - 30$ 
```

```
# 1. Определить корни

from sympy import *

x = Symbol('x')
fx1 = -18 * x ** 3 + 5 * x ** 2 + 10 * x - 30

#Корни уравнения
solveset(fx1, x, Reals).evalf(3)
```

$\{-1.24\}$

```

# 2. Найти интервалы, на которых функция возрастает
# 3. Найти интервалы, на которых функция убывает

diff_list = [-oo, oo]
diff_list[1:1] = solveset(diff(fx1), x, Reals).evalf(3)
inc_list, dec_list = [], []
for i in range(1, len(diff_list)):
    demeanor = is_increasing(fx1, Interval.open(diff_list[i - 1], diff_list[i]))
    if demeanor:
        inc_list.append([diff_list[i - 1], diff_list[i]])
    else:
        dec_list.append([diff_list[i - 1], diff_list[i]])

print(f'Возрастает на участке: ', *inc_list)
print(f'Убывает на участке: ', *dec_list)

```

```

Возрастает на участке: [-0.348, 0.533]
Убывает на участке: [-oo, -0.348] [0.533, oo]

```

4. Построить график

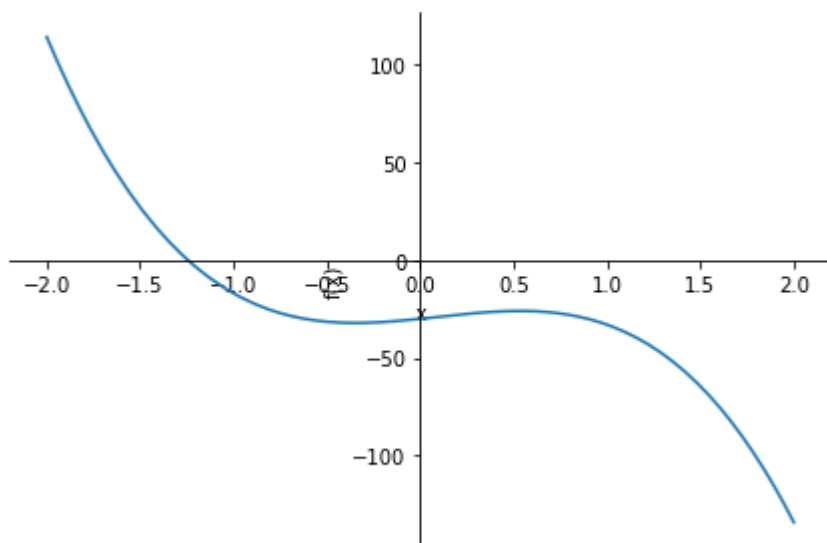
```

from sympy.plotting import plot
init_printing()

plot(fx1, (x, -2, 2))

```

[Download](#)



<sympy.plotting.plot.Plot at 0x7f1acce214f0>

```
# 5. Вычислить вершину
from random import uniform

dataset = sorted(solveset(diff(fx1), x, Reals).evalf(3))
dataset.insert(0, dataset[0] - 1)

dif = diff(fx1)
my_list = []

for index, value in enumerate(dataset):
    my_list.append(dif.subs(x, uniform(value, dataset[index] + 1)))
    if index:
        if my_list[index - 1] < 0 < my_list[index]:
            print(f'Минимум: ({value}, {fx1.subs(x, value).evalf(3)})')
        elif my_list[index - 1] > 0 > my_list[index]:
            print(f'Максимум: ({value}, {fx1.subs(x, value).evalf(3)})')
```

Минимум: (-0.348, -32.1)

Максимум: (0.533, -26.0)

6. Определить промежутки, на которых $fx > 0$

```
print('fx1 > 0 в интервале:')
solveset(fx1 > 0, x, Reals).evalf(3)
```

fx1 > 0 в интервале:

[↓ Download](#)

$(-\infty, -1.24)$

7. Определить промежутки, на которых $fx < 0$

```
print('fx1 < 0 в интервале:')
solveset(fx1 < 0, x, Reals).evalf(3)
```

fx1 < 0 в интервале:

[↓ Download](#)

$(-1.24, \infty)$

```
# Функция 2:
#  $f = 2 * x^3 + 2 * x^2 - 18 * x - 18$ 
```

```
# 1. Определить корни
```

```
from sympy import *
```

```
x = Symbol('x')
fx2 = 2 * x ** 3 + 2 * x ** 2 - 18 * x - 18
```

```
#Корни уравнения
solveset(fx2, x, Reals).evalf(3)
```

[Download](#)

```
{-3.0, -1.0, 3.0}
```

```
# 2. Найти интервалы, на которых функция возрастает
```

```
# 3. Найти интервалы, на которых функция убывает
```

```
diff_list = [-oo, oo]
diff_list[1:1] = solveset(diff(fx2), x, Reals).evalf(3)
inc_list, dec_list = [], []
for i in range(1, len(diff_list)):
    demeanor = is_increasing(fx2, Interval.open(diff_list[i - 1], diff_list[i]))
    if demeanor:
        inc_list.append([diff_list[i - 1], diff_list[i]])
    else:
        dec_list.append([diff_list[i - 1], diff_list[i]])

print(f'Возрастает на участке: ', *inc_list)
print(f'Убывает на участке: ', *dec_list)
```

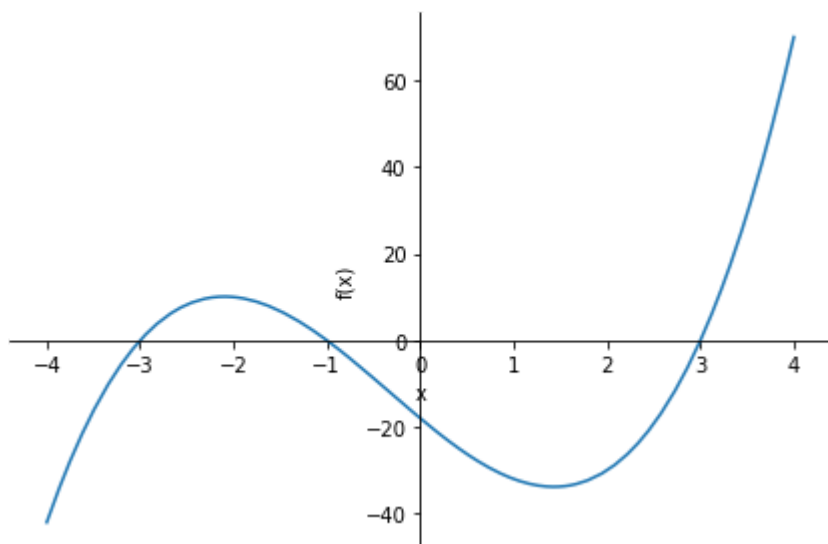
```
Возрастает на участке: [-oo, -2.10]
Убывает на участке: [-2.10, 1.43] [1.43, oo]
```

```
# 4. Построить график
```

```
from sympy.plotting import plot
init_printing()
```

```
plot(fx2, (x, -4, 4))
```

[Download](#)



<sympy.plotting.plot.Plot at 0x7f1ab56edfd0>

5. Вычислить вершину

```
from random import uniform
```

```
dataset = sorted(solveset(diff(fx2), x, Reals).evalf(3))
```

```
dataset.insert(0, dataset[0] - 1)
```

```
dif = diff(fx2)
```

```
my_list = []
```

```
for index, value in enumerate(dataset):
```

```
    my_list.append(dif.subs(x, uniform(value, dataset[index] + 1)))
```

```
    if index:
```

```
        if my_list[index - 1] < 0 < my_list[index]:
```

```
            print(f'Минимум: ({value}, {fx2.subs(x, value).evalf(3)})')
```

```
        elif my_list[index - 1] > 0 > my_list[index]:
```

```
            print(f'Максимум: ({value}, {fx2.subs(x, value).evalf(3)})')
```

```
Максимум: (-2.10, 10.1)
```

```
Минимум: (1.43, -33.8)
```

6. Определить промежутки, на которых $fx > 0$

```
print('fx2 > 0 в интервале:')
```

```
solveset(fx2 > 0, x, Reals).evalf(3)
```

```
fx2 > 0 в интервале:
```

[Download](#)

$$(-3.0, -1.0) \cup (3.0, \infty)$$

7. Определить промежутки, на которых $fx < 0$

```
print('fx2 < 0 в интервале:')
solveset(fx2 < 0, x, Reals).evalf(3)
```

fx2 < 0 в интервале:

[Download](#)

$$(-\infty, -3.0) \cup (-1.0, 3.0)$$

Функция 3:

```
# f = (x ** 2 + 3) / (3 * (x + 1))
```

1. Определить корни

```
from sympy import *
```

```
x = Symbol('x')
```

```
fx3 = 2 * x ** 3 + 2 * x ** 2 - 18 * x - 18
```

#Корни уравнения

```
solveset(fx3, x, Reals).evalf(3)
```

[Download](#)

$$\{-3.0, -1.0, 3.0\}$$

2. Найти интервалы, на которых функция возрастает

3. Найти интервалы, на которых функция убывает

```
diff_list = [-oo, oo]
diff_list[1:1] = solveset(diff(fx3), x, Reals).evalf(3)
inc_list, dec_list = [], []
for i in range(1, len(diff_list)):
    demeanor = is_increasing(fx3, Interval.open(diff_list[i - 1], diff_list[i]))
    if demeanor:
        inc_list.append([diff_list[i - 1], diff_list[i]])
    else:
        dec_list.append([diff_list[i - 1], diff_list[i]])

print(f'Возрастает на участке: ', *inc_list)
print(f'Убывает на участке: ', *dec_list)
```

Возрастает на участке: $[-\infty, -2.10]$

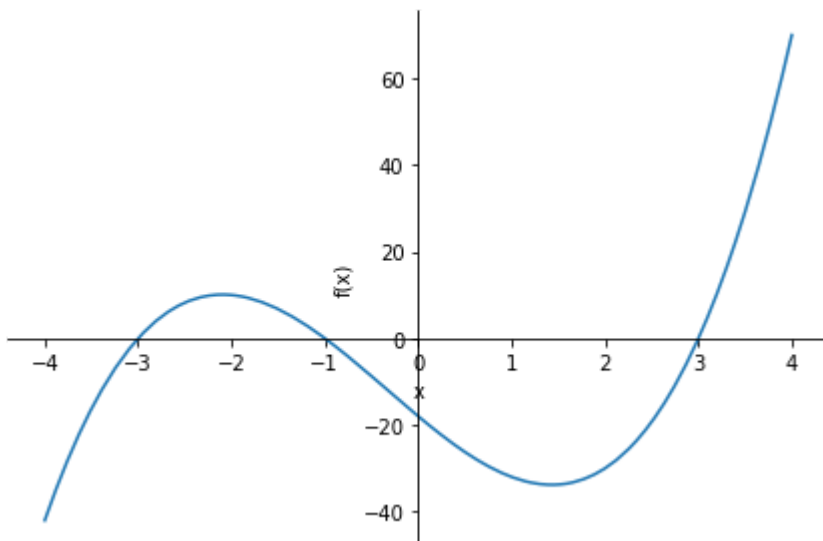
Убывает на участке: $[-2.10, 1.43]$ $[1.43, \infty]$

4. Построить график

```
from sympy.plotting import plot
init_printing()

plot(fx3, (x, -4, 4))
```

[Download](#)



<sympy.plotting.plot.Plot at 0x7f1accbd2850>

```
# 5. Вычислить вершину
from random import uniform

dataset = sorted(solveset(diff(fx3), x, Reals).evalf(3))
dataset.insert(0, dataset[0] - 1)

dif = diff(fx3)
my_list = []

for index, value in enumerate(dataset):
    my_list.append(dif.subs(x, uniform(value, dataset[index] + 1)))
    if index:
        if my_list[index - 1] < 0 < my_list[index]:
            print(f'Минимум: ({value}, {fx3.subs(x, value).evalf(3)})')
        elif my_list[index - 1] > 0 > my_list[index]:
            print(f'Максимум: ({value}, {fx3.subs(x, value).evalf(3)})')
```

Максимум: (-2.10, 10.1)
 Минимум: (1.43, -33.8)

6. Определить промежутки, на которых $fx > 0$

```
print('fx3 > 0 в интервале:')
solveset(fx3 > 0, x, Reals).evalf(3)
```

$fx3 > 0$ в интервале:

[Download](#)

$(-3.0, -1.0) \cup (3.0, \infty)$

7. Определить промежутки, на которых $fx < 0$

```
print('fx3 < 0 в интервале:')
solveset(fx3 < 0, x, Reals).evalf(3)
```

$fx3 < 0$ в интервале:

[Download](#)

$(-\infty, -3.0) \cup (-1.0, 3.0)$