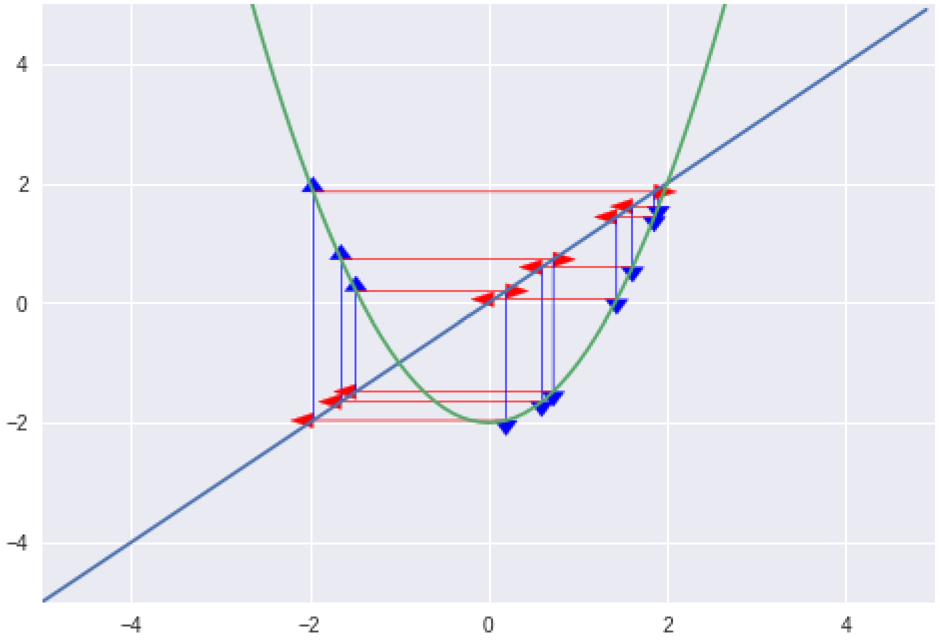
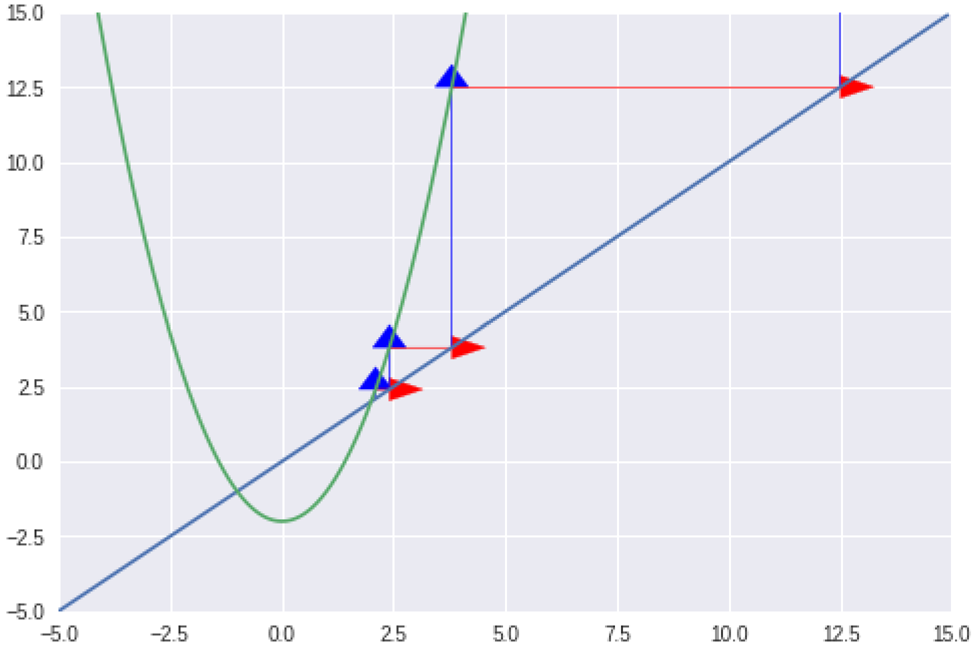
```
import numpy as np
import matplotlib.pyplot as plt


def fixed_point_iteration(f, x_0, num_it, a, b):
  plt.xlim(a, b)
  plt.ylim(a, b)

  xs = np.arange(a, b, 0.1)

  plt.plot(xs, xs)
  plt.plot(xs, list(map(f, xs)))

  x = x_0
  for _ in np.arange(num_it):
    plt.arrow(x,
              x,
              0,
              f(x) - x,
              head_width=(b / 20),
              head_length=(b / 20),
              fc='b',
              ec='b')

    plt.arrow(x,
              f(x),
              f(x) - x,
              0,
              head_width=(b / 20),
              head_length=(b / 20),
              fc='r',
              ec='r')

    x = f(x)

  plt.show()
```

```
fixed_point_iteration(lambda x: x**2 - 2, 1.9, 10, -5, 5)
```

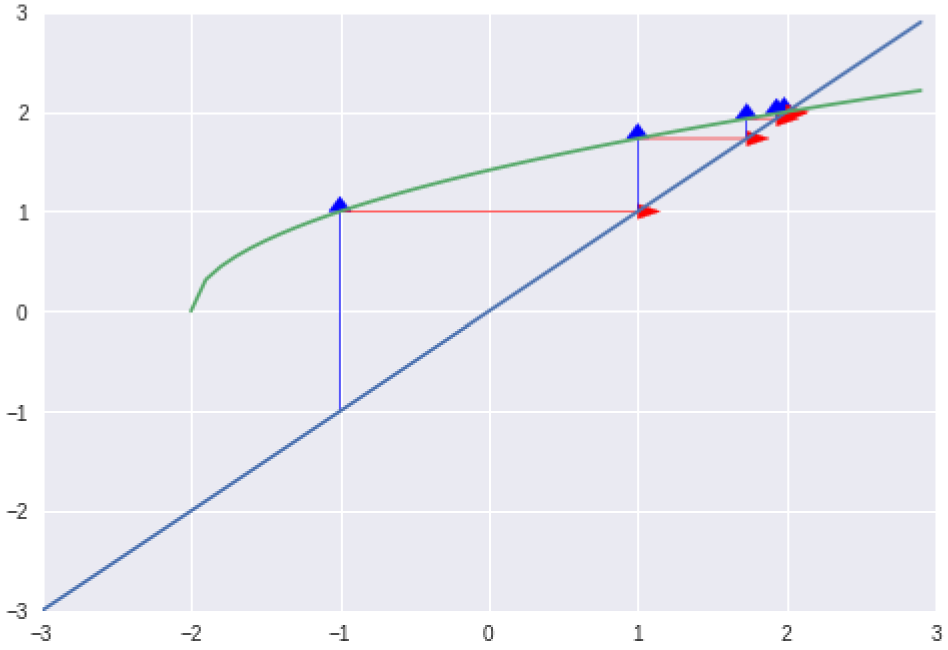


```
fixed_point_iteration(lambda x: x**2 - 2, 2.1, 5, -5, 15)
```

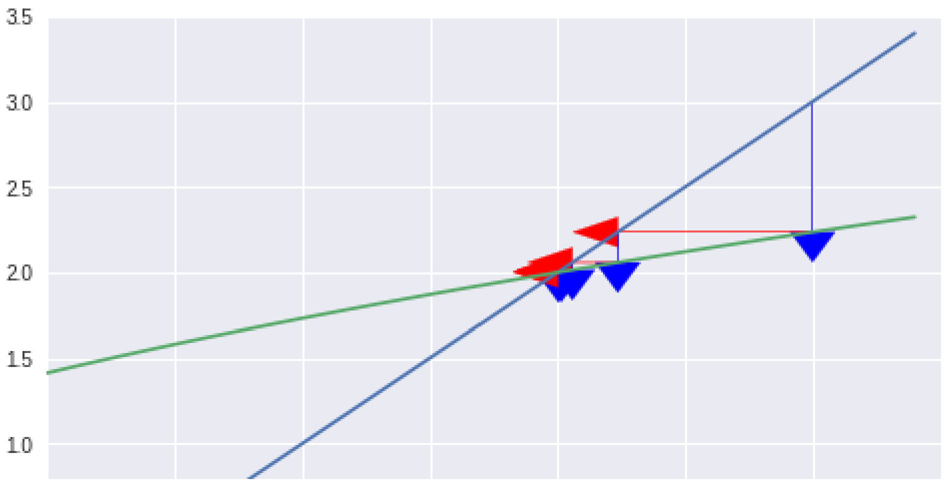


```
fixed_point_iteration(lambda x: (x + 2)**0.5, -1, 5, -3, 3)
```

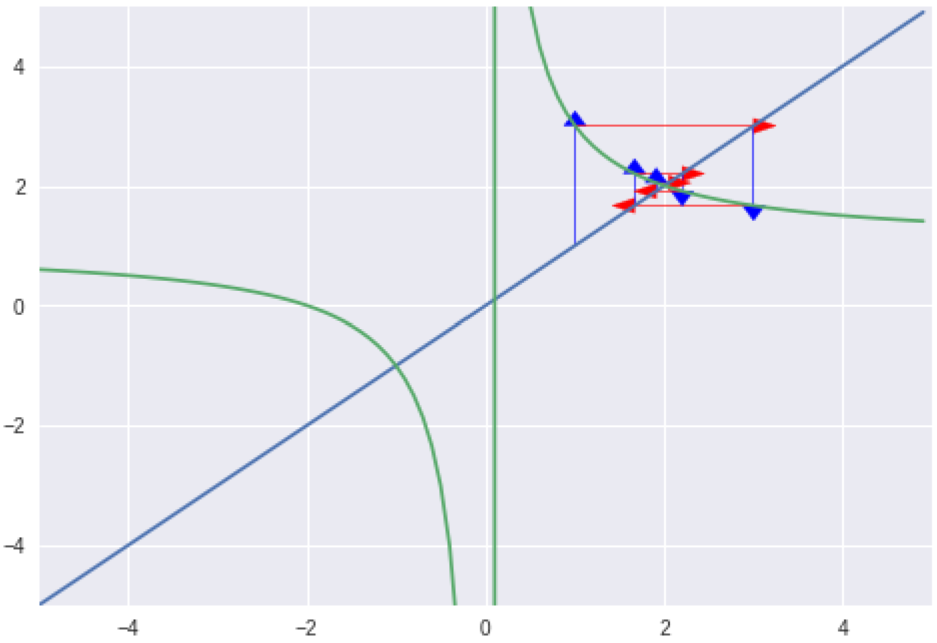
```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: RuntimeWarning: invali
  """Entry point for launching an IPython kernel.
```



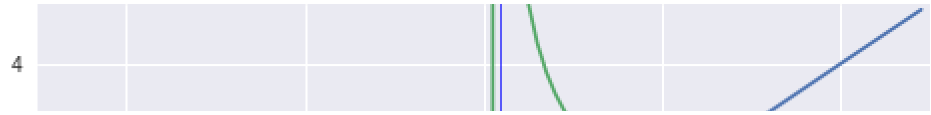
```
fixed_point_iteration(lambda x: (x + 2)**0.5, 3, 5, 0, 3.5)
```



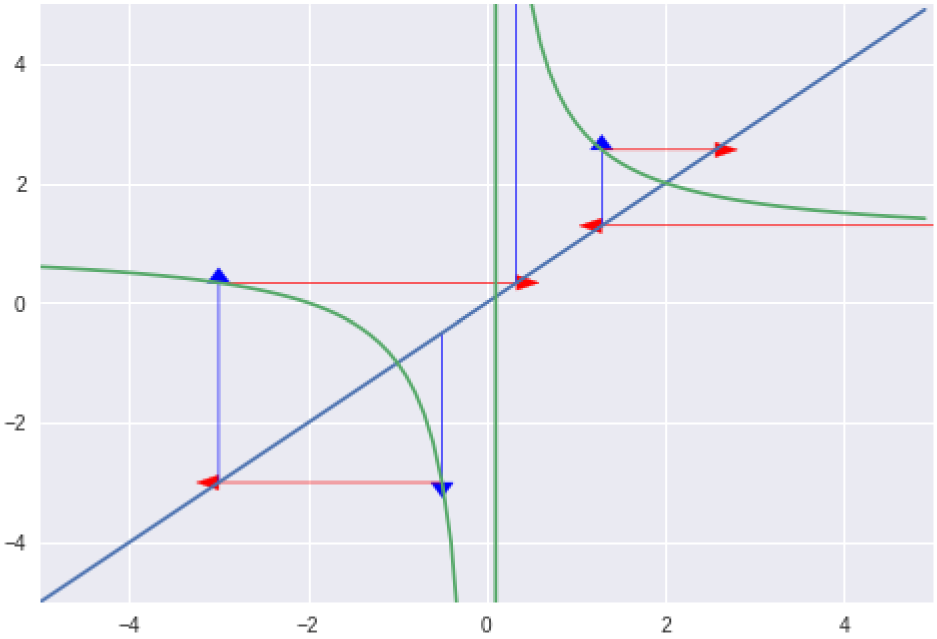
```
fixed_point_iteration(lambda x: 1 + 2/x, 1, 5, -5, 5)
```



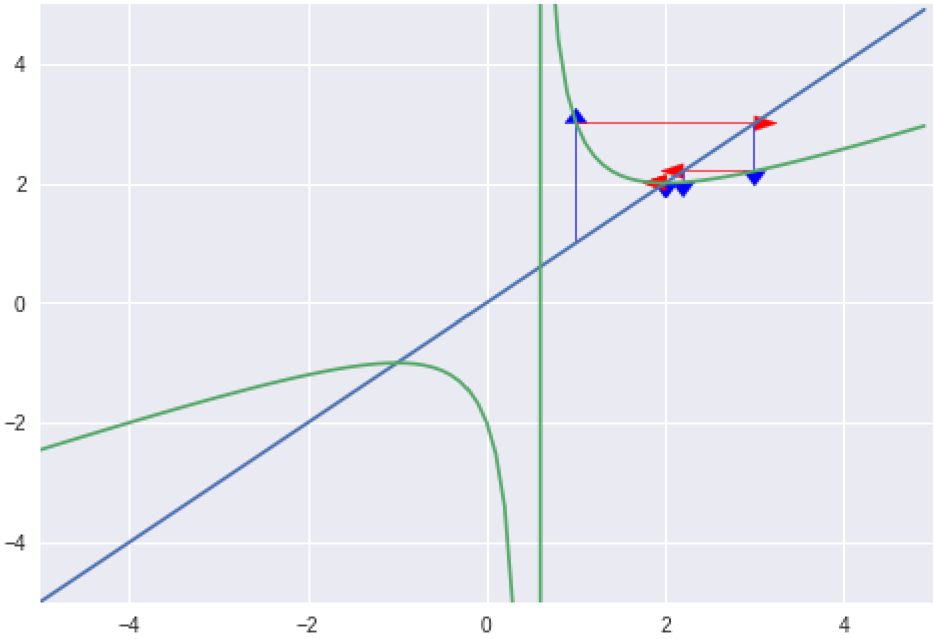
```
fixed_point_iteration(lambda x: 1 + 2/x, -2.5, 5, -5, 5)
```



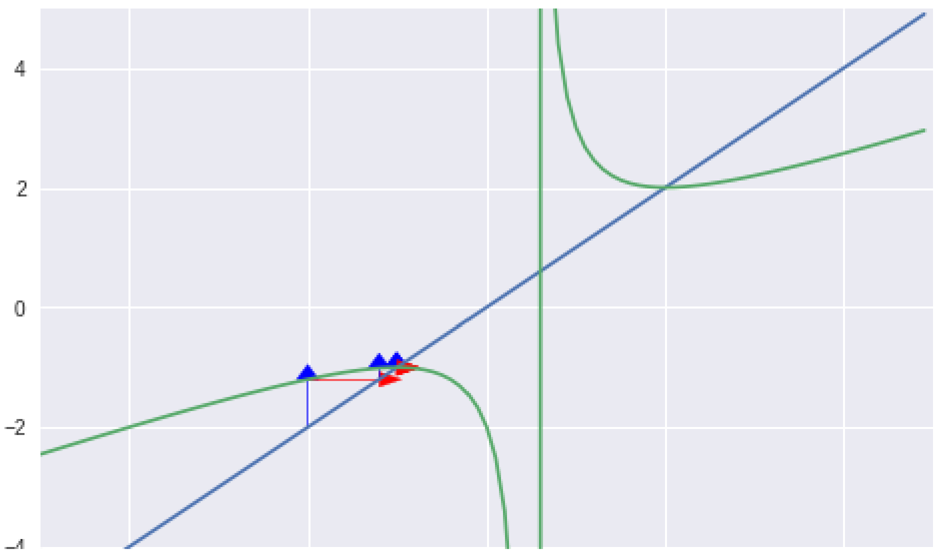
```
fixed_point_iteration(lambda x: 1 + 2/x, -0.5, 5, -5, 5)
```



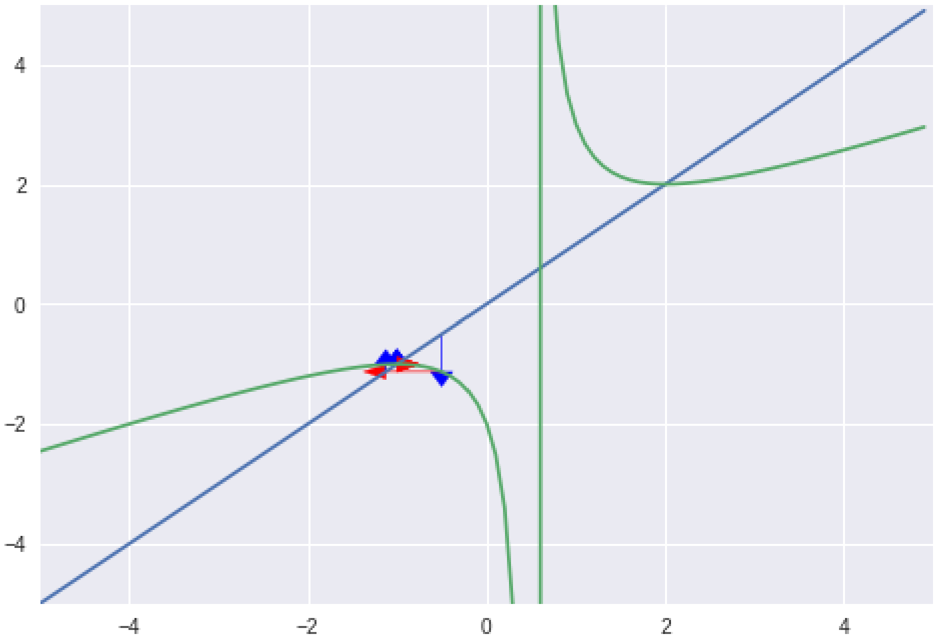
```
fixed_point_iteration(lambda x: (x**2 + 2) / (2*x - 1), 1, 5, -5, 5)
```



```
fixed_point_iteration(lambda x: (x**2 + 2) / (2*x - 1), -2, 5, -5, 5)
```



```
fixed_point_iteration(lambda x: (x**2 + 2) / (2*x - 1), -0.5, 5, -5, 5)
```


4
2
0
−2
−4
−4
−2
0
2
4