```
clc
clear
close all

x0 = [3, 2]';

N = eye(2);
grad = get_grad(x0);
s = get_s(grad);
alpha0 = 0;
count = 0;
x = x0;

fnext = get_f(x);

% s = get_s([5, 5]')
alpha = alpha0;
step_size = 0.001;

fprev = get_f(x);

% get astar by brute force
while 1
    alpha = alpha + step_size;
    fnext = get_f(x + alpha*s);

    if fnext > fprev
        astar = alpha - step_size;
        break
    end

    fprev = fnext;

end

% take minimizing step
x = x + astar*s;

x1 = x

% compute gamma
gamma = [get_grad(x1) - get_grad(x0)];
dx = x1 - x0;

% BFGS update
N1 = N + (1 + (gamma'*N*gamma)/(dx'*gamma))*((dx*dx')/(dx'*gamma)) -
 (((dx*gamma'*N) + (N*gamma*dx'))/(dx'*gamma))


% new search dir
s1 = N1*gamma;
```

```
x2 = x1 + -s1;

func = get_f(x1);
func = get_f(x2);
```

```
function s = get_s(gradf)
s = -gradf/norm(gradf);
end
```

```
function f = get_f(x)
f = (x(1))^2 - 2*x(1)*x(2) + 4*(x(2))^2;
end
```

```
function gradf = get_grad(x)
grad_1 = 2*x(1) - 2*x(2);
grad_2 = 8*x(2) - 2*x(1);
gradf = [grad_1, grad_2]';
end
```

```
x1 =

    2.7143
    0.5713
```

```
N1 =

    1.1389    0.2661
    0.2661    0.1876
```

Published with MATLAB® R2017a