In [1]:
```python
import matplotlib.pyplot as plt
```

In [2]:
```python
# Define the cost function
def cost_function(x):
    return (x + 5) ** 2
```

In [3]:
```python
# Define the gradient of the function
def gradient(x):
    return 2 * (x + 5)
```

In [4]:
```python
# Hyperparameters
learning_rate = 0.1
initial_x = 3.0
num_iterations = 100
```

In [5]:
```python
# Lists to store values for plotting
x_values = []
y_values = []
```

In [6]:
```python
# Initialize x
x = initial_x
```

In [7]:
```python
# Gradient Descent loop
for i in range(num_iterations):
    x_values.append(x)
    y_values.append(cost_function(x))

    grad = gradient(x)
    x = x - learning_rate * grad  # Update x

    print(f"Iteration {i+1}: x = {x}, Cost = {cost_function(x)}")
```

```
Iteration 1: x = 1.4, Cost = 40.96000000000001
Iteration 2: x = 0.11999999999999966, Cost = 26.21439999999999
Iteration 3: x = -0.9040000000000001, Cost = 16.777216
Iteration 4: x = -1.7232000000000003, Cost = 10.737418239999998
Iteration 5: x = -2.3785600000000002, Cost = 6.8719476735999985
Iteration 6: x = -2.902848, Cost = 4.398046511104
Iteration 7: x = -3.3222784, Cost = 2.8147497671065596
Iteration 8: x = -3.65782272, Cost = 1.8014398509481986
Iteration 9: x = -3.926258176, Cost = 1.1529215046068466
Iteration 10: x = -4.1410065408, Cost = 0.7378697629483816
Iteration 11: x = -4.312805232640001, Cost = 0.4722366482869637
Iteration 12: x = -4.450244186112, Cost = 0.30223145490365716
Iteration 13: x = -4.5601953488896, Cost = 0.1934281311383406
Iteration 14: x = -4.64815627911168, Cost = 0.12379400392853822
Iteration 15: x = -4.718525023289343, Cost = 0.07922816251426466
Iteration 16: x = -4.774820018631475, Cost = 0.05070602400912922
Iteration 17: x = -4.81985601490518, Cost = 0.03245185536584277
Iteration 18: x = -4.855884811924144, Cost = 0.020769187434139472
Iteration 19: x = -4.884707849539315, Cost = 0.013292279957849222
```

In [8]:
```
# Final results
print(f"\nOptimal x: {x}")
print(f"Minimum value of y: {cost_function(x)}")
```

```
Optimal x: -4.999999998370371
Minimum value of y: 2.6556902744296365e-18
```

In [9]:
```
# Plotting the convergence
plt.plot(x_values, y_values, 'ro-')
plt.title('Gradient Descent Visualization for y = (x + 5)^2')
plt.xlabel('x')
plt.ylabel('y')
plt.show()
```