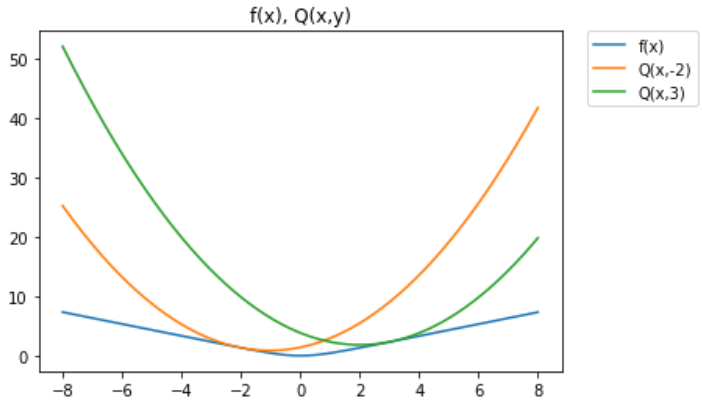
In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [3]:
```
def f(x):
    return np.log(np.cosh(x))

def df(x):
    return np.tanh(x)

def df2(x):
    return (1/np.cosh(x))**2

def Q(x,y):
    return f(y) + df(y)*(x-y) + ((x-y)**2)/2.0
```

6.4 (c)

In [14]:
```
x_arr = np.arange(-8-0.01, 8+0.01, 0.01)

plt.plot(x_arr, f(x_arr), label="f(x)")
plt.plot(x_arr, Q(x_arr,-2), label="Q(x,-2)")
plt.plot(x_arr, Q(x_arr,3), label="Q(x,3)")
plt.title('f(x), Q(x,y)')
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

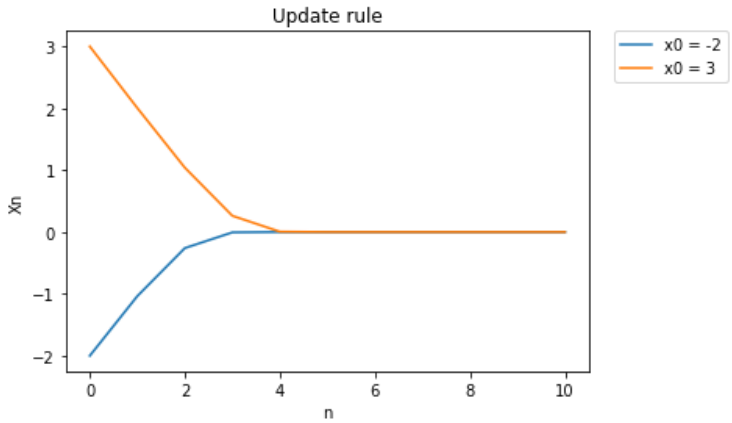
Out[14]: <matplotlib.legend.Legend at 0x118d69690>



6.4 (f)

In [30]:
```
def updateRule(xn):
    return xn - np.tanh(xn)

def getUpdatedXn(x0, n):
    arr_xn = [x0]
    for i in range(len(n)-1):
        arr_xn.append(updateRule(arr_xn[-1]))

    return arr_xn

n = np.arange(0, 11)
arr_xn_1 = getUpdatedXn(-2, n)
arr_xn_2 = getUpdatedXn(3, n)

plt.plot(n, arr_xn_1, label="x0 = -2")
plt.plot(n, arr_xn_2, label="x0 = 3")
plt.title('Update rule')
plt.ylabel('Xn')
plt.xlabel('n')
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[30]: <matplotlib.legend.Legend at 0x119515110>

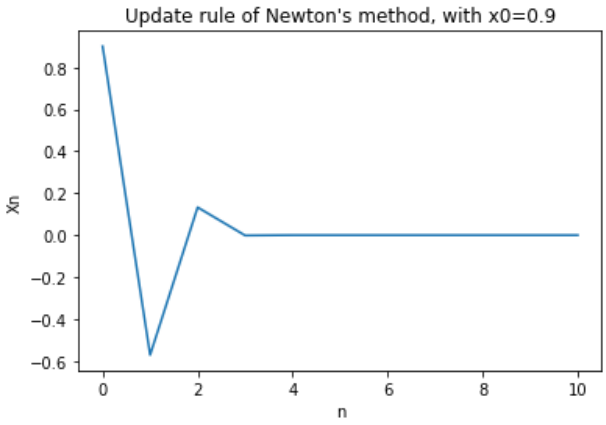


6.4 (g)

In [35]:
```
def updateRuleNewton(xn):
    return xn-np.sinh(xn)*np.cosh(xn)

def getUpdatedXnByNetwon(x0, n):
    arr_xn = [x0]
    for i in range(len(n)-1):
        arr_xn.append(updateRuleNewton(arr_xn[-1]))

    return arr_xn
```

In [78]:
```
# converging example
n = np.arange(0, 11)
x0_1 = 0.9
arr_xn_1 = getUpdatedXnByNetwon(x0_1, n)

plt.plot(n, arr_xn_1)
plt.ylabel('Xn')
plt.xlabel('n')
plt.title("Update rule of Newton's method, with x0=0.9")
```

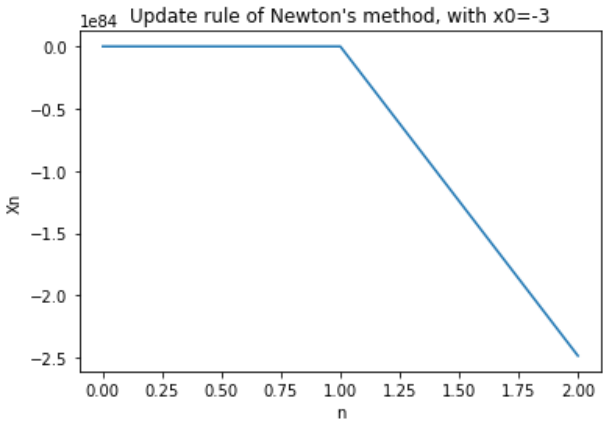
Out[78]: Text(0.5, 1.0, "Update rule of Newton's method, with x0=0.9")



In [79]:
```
# non-converging example
n = np.arange(0, 11)
x0_2 = -3
arr_xn_2 = getUpdatedXnByNetwon(x0_2, n)

plt.plot(n, arr_xn_2)
plt.ylabel('Xn')
plt.xlabel('n')
plt.title("Update rule of Newton's method, with x0=-3")
```

```
/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: overflow encount
ered in sinh

/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: overflow encount
ered in cosh

/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: invalid value en
countered in double_scalars
```

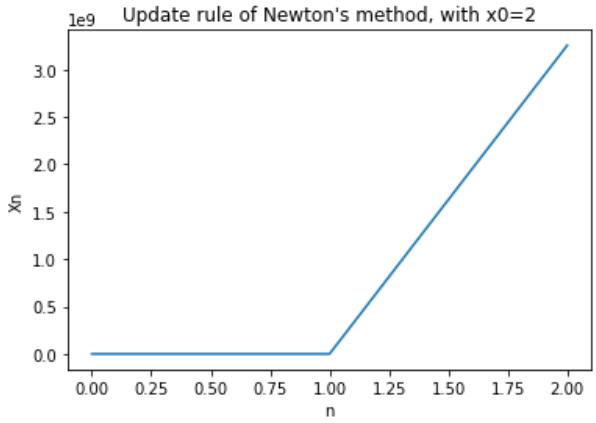
Out[79]: Text(0.5, 1.0, "Update rule of Newton's method, with x0=-3")



In [80]:
```
# non-converging example
n = np.arange(0, 11)
x0_3 = 2
arr_xn_3 = getUpdatedXnByNetwon(x0_3, n)

plt.plot(n, arr_xn_3)
plt.ylabel('Xn')
plt.xlabel('n')
plt.title("Update rule of Newton's method, with x0=2")
```

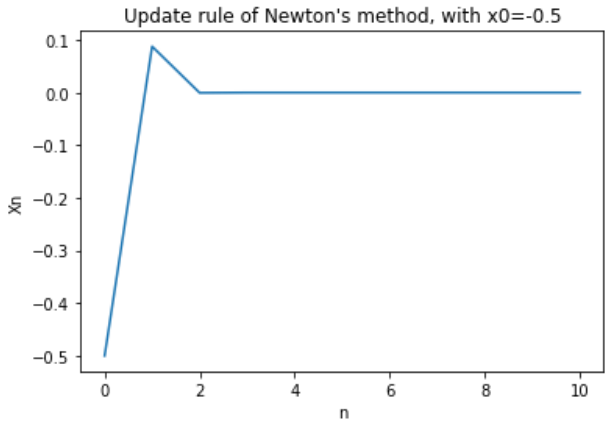
```
/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: overflow encount
ered in sinh

/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: overflow encount
ered in cosh

/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: invalid value en
countered in double_scalars
```

Out[80]: Text(0.5, 1.0, "Update rule of Newton's method, with x0=2")



In [81]:
```
# converging example
n = np.arange(0, 11)
x0_4 = -0.5
arr_xn_4 = getUpdatedXnByNetwon(x0_4, n)

plt.plot(n, arr_xn_4)
plt.ylabel('Xn')
plt.xlabel('n')
plt.title("Update rule of Newton's method, with x0=-0.5")
```

Out[81]: Text(0.5, 1.0, "Update rule of Newton's method, with x0=-0.5")



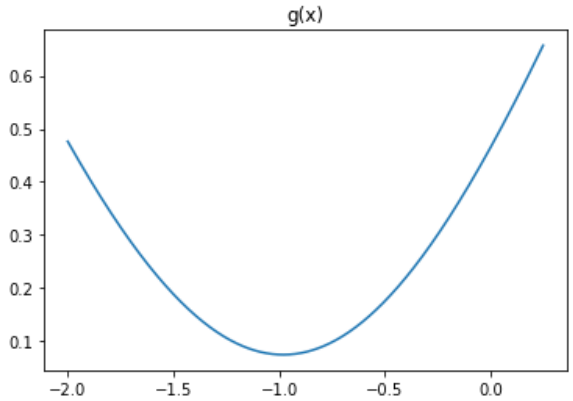
In [77]:
```
for x0 in np.arange(0, 1.5, 0.00001):
    n = np.arange(0, 11)
    arr_xn = getUpdatedXnByNetwon(x0, n)

    if 'nan' in str(arr_xn[-1]) or '-inf' in str(arr_xn[-1]):
    #if abs(arr_xn[-1]) < 0.000001:
        print("Didn't converge from x0 = %f" % x0)
        break

for x0 in np.arange(0, -1.5, -0.00001):
    n = np.arange(0, 11)
    arr_xn = getUpdatedXnByNetwon(x0, n)

    if 'nan' in str(arr_xn[-1]) or '-inf' in str(arr_xn[-1]):
        print("Didn't converge from x0 = %f" % x0)
        break
```

```
/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: overflow encount
ered in double_scalars

/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: overflow encount
ered in sinh

/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: overflow encount
ered in cosh

/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: invalid value en
countered in double_scalars


Didn't converge from x0 = 1.088870
Didn't converge from x0 = -1.088720
```

6.4 (h)

In [58]:
```
def g(x):
    arr_k = np.arange(1,11)
    return np.sum(np.log(np.cosh(x + 2 / (arr_k**0.5)))) / 10

arr_x = np.arange(-2,0.25,0.0001)
arr_g = []

for i in range(len(arr_x)):
    arr_g.append(g(arr_x[i]))

plt.plot(arr_x,arr_g)
plt.title('g(x)')
```

Out[58]:
```
Text(0.5, 1.0, 'g(x)')
```

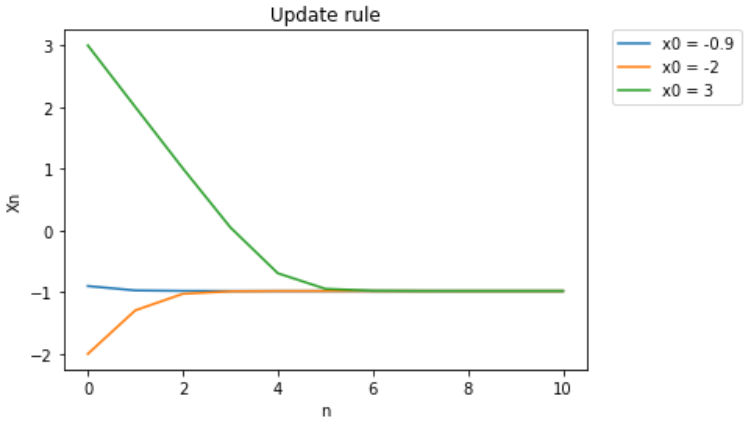


6.4 (k)

In [61]:
```
def updateR(xn):
    k_arr = np.arange(1, 11)
    return xn - np.sum(np.tanh(xn+2/k_arr**0.5)) / 10

def getUpdatedXnOfR(x0, n):
    arr_xn = [x0]
    for i in range(len(n)-1):
        arr_xn.append(updateR(arr_xn[-1]))

    return arr_xn
```

In [69]:
```
n = np.arange(0, 11)
x0_1 = -0.9
arr_xn_1 = getUpdatedXnOfR(x0_1, n)
x0_2 = -2
arr_xn_2 = getUpdatedXnOfR(x0_2, n)
x0_3 = 3
arr_xn_3 = getUpdatedXnOfR(x0_3, n)


plt.title("Update rule of R(x, Xn)")
plt.plot(n, arr_xn_1, label="x0 = -0.9")
plt.plot(n, arr_xn_2, label="x0 = -2")
plt.plot(n, arr_xn_3, label="x0 = 3")
plt.title('Update rule')
plt.ylabel('Xn')
plt.xlabel('n')
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[69]:
```
<matplotlib.legend.Legend at 0x11af49e10>
```



In [67]:
```
arr_xn_1
```

Out[67]:
```
[-0.9,
 -0.9709692353153697,
 -0.9789521820224213,
 -0.9798645110231493,
 -0.9799691070710898,
 -0.9799811032721009,
 -0.9799824791857176,
 -0.9799826369979926,
 -0.9799826550984956,
 -0.9799826571745586,
 -0.9799826574126755]
```