ZAD 1

```
def y_(sum):
    if sum > 0:
        return 1
    else:
        return -1

def suma(xi, xi2, w1, w2, x0, Theta):
    sum = xi * w1 + xi2 * w2 + Theta * x0
    return sum

x1 = [2, 2, 0, -2, -2, 0, 4]
x2 = [1, 2, 6, 10, 0, 0, -20]
d = [1, 1, 1, -1, -1, -1, -1]
w = [0, 0, 0]
#Theta = 0
iterator = 0
x0 = 1

while True:
    z = True
    for i in range(7):
        y = y_(suma(x1[i], x2[i], w[0], w[1], x0, w[2]))
        if y != d[i]:
            w[0] = w[0] + d[i] * x1[i]
            w[1] = w[1] + d[i] * x2[i]
            w[2] = w[2] + d[i] * x0
            z = False
        print(y , d[i])
    iterator += 1
    print()
    print(iterator)
    print()
    if z:
        break

print("w 0 =", w[2] , "w 1 =", w[0] , "w 2 =", w[1])
```

```
1 -1
-1 -1
-1 -1
-1 -1

34

1 1
1 1
1 1
1 -1
-1 -1
-1 -1
1 -1
```

ZAD 2

```
import math

WIELKOSC_TAB = 2
WIELKOSC_TAB2 = 13
beta = 0.5
ni = 0.35

def f(suma):
    y = (1 - math.exp(-1*suma*beta))/(1+math.exp(-1*suma*beta))
    return y

jeden = [-1, -1, 1, -1, -1, 1, -1, -1, 1, -1, -1, 1, 1]
cztery = [1, -1, 1, 1, 1, 1, -1, -1, 1, -1, -1, 1, 1]
d = [-1, 1]
w = [0]*WIELKOSC_TAB2
w2 = [0]*WIELKOSC_TAB2
iterator = 0
d1=-1
d4=1

while True:
    E = 0
    for i in range(12):
        s = jeden[i] * w[i] + w[12] * jeden[12]
        y = f(s)
        w[i] += d[0] * jeden[i] * ni
        w[12] += d[0] * jeden[12] * ni
        print(y,d[0])
    print(y,"<-- y koncowy dla 1",d[0],"<--d oczekiwana")
    E += 0.5*(d1 - y)*(d1 - y)
    for i in range(12):
        s2 = cztery[i] * w2[i] + w2[12] * cztery[12]
        y2 = f(s2)
        w2[i] += d[1] * cztery[i] * ni
        w2[12] += d[1] * cztery[12] * ni
        print(y2,d[1])
    print(y2,"<-- y koncowy dla 4",d[1],"<--d oczekiwana")
    iterator += 1
    print("\n",iterator,"<-- przebieg funkcji\n")
    E += 0.5*(d4 - y2)*(d4 - y2)
    print(E,"blad sumaryczny")
    if E <= 0.0000000005:
        break
```

```
 -0.9995503664595334 -1
 -0.9996225383440512 -1
 -0.9996831275617949 -1
 -0.9996831275617949 <-- y koncowy dla 1 -1 <--d oczekiwana
 0.9978298055470949 1
 0.9981778976111988 1
 0.9984701996041594 1
 0.9987156405988281 1
 0.9989217243160301 1
 0.999094755553519 1
 0.9992400309704962 1
 0.9993619997421463 1
 0.999464398774423 1
 0.9995503664595333 1
 0.9996225383440513 1
 0.9996831275617949 1
 0.9996831275617949 <-- y koncowy dla 4 1 <--d oczekiwana

  4 <-- przebieg funkcji

 1.0040814209404929e-07 blad sumaryczny
 -0.9997766933187409 -1
 -0.9998125402783868 -1
 -0.9998426332525774 -1
 -0.9998678957102907 -1
 -0.9998891029505544 -1
 -0.9999069058645483 -1
 -0.9999218508904095 -1
 -0.9999343967697488 -1
 -0.9999449286177708 -1
 -0.9999537697371759 -1
 -0.9999611915372831 -1
 -0.9999674218620193 -1
 -0.9999674218620193 <-- y koncowy dla 1 -1 <--d oczekiwana
 0.999776693318741 1
```

Zadanie 3

```
import math

WIELKOSC_TAB = 2
WIELKOSC_TAB2 = 13
beta = 1
ni = 1

def f(suma):
    return 1 / (1 + math.exp(-beta * suma))

def fprim(x):
    y = beta * f(x)* (1-f(x))
    return y

w = [0]*13
jeden = [1,-1,-1,1,-1,-1,1,-1,1,1,-1,-1,1]
cztery = [1,1,1,1,1,1,1,-1,-1,1,-1,-1,1]
cosjeden = [1,-1,-1,1,-1,-1,1,-1,1,1,-1,-1,1]
cosdwa = [1, 1, 1, 1, 1, 1, 1, -1, -1, 1, -1, -1, 1]
d = [-1, 1]
iterator = 0
d4 = 0
d2 = 0
d1 = 1
s2=0

while True:
    E=0
    s=0
    for i in range(13):
      s = s + jeden[i] * w[i]
    y = f(s)
    for i in range(13):
        w[i] = w[i] + ni * beta*(1-y)*y*(d1-y)*jeden[i]
    E = 0.5*(d1 - y)*(d1 - y)
    s=0
    for i in range(13):
     s = s + cztery[i] * w[i]
    y2 = f(s)
    for i in range(13):
        w[i] = w[i] + ni * beta*(1-y2)*y2*(d4-y2)*cztery[i]
    iterator += 1
    E += 0.5*(d4 - y2)*(d4 - y2)
    print(E,"blad sumaryczny")
    if E <= 0.0005:
        break
```

```
s = sum([jeden[i] * w[i] for i in range(13)])
print("wyjscie dla 1",f(s))
s = sum([cztery[i] * w[i] for i in range(13)])
print("wyjscie dla  4",f(s))
s = sum([cosjeden[i] * w[i] for i in range(13)])
print("wyjscie dla niby 1",f(s))
s = sum([cosdwa[i] * w[i] for i in range(13)])
print("wyjscie dla niby 4",f(s))
```

```
0.0010091370038741961 blad sumaryczny
0.000990167117426475 blad sumaryczny
0.0009718902629434627 blad sumaryczny
0.0009542692998583558 blad sumaryczny
0.0009372696891328926 blad sumaryczny
0.0009208592696910731 blad sumaryczny
0.0009050080574993233 blad sumaryczny
0.0008896880646626945 blad sumaryczny
0.0008748731362499181 blad sumaryczny
0.0008605388028540927 blad sumaryczny
0.0008466621471476034 blad sumaryczny
0.000833221682906579 blad sumaryczny
0.0008201972451672632 blad sumaryczny
0.0008075698903378218 blad sumaryczny
0.0007953218052293503 blad sumaryczny
0.0007834362240909671 blad sumaryczny
0.0007718973528398348 blad sumaryczny
0.000760690299768867 blad sumaryczny
0.0007498010012095598 blad sumaryczny
0.0007392162177859437 blad sumaryczny
0.00072892337214866 blad sumaryczny
0.0007188910608750453 blad sumaryczny
0.0007091666942496062 blad sumaryczny
0.0006996809867882203 blad sumaryczny
0.0006904433976203323 blad sumaryczny
0.000681444355686343 blad sumaryczny
0.0006726747748732572 blad sumaryczny
0.000664126023726262 blad sumaryczny
0.0006557898974003987 blad sumaryczny
0.0006476585916613905 blad sumaryczny
0.0006397246787632428 blad sumaryczny
0.0006319810850465588 blad sumaryczny
0.00062442107011608 blad sumaryczny
0.0006170382074692125 blad sumaryczny
0.000609826366459004 blad sumaryczny
0.0006027796954855618 blad sumaryczny
0.0005958926063195146 blad sumaryczny
0.0005891597594696054 blad sumaryczny
0.0005825760505142019 blad sumaryczny
0.0005761365973234731 blad sumaryczny
0.0005698367281054153 blad sumaryczny
0.0005636719702142447 blad sumaryczny
0.0005576380396653019 blad sumaryczny
0.0005517308313048312 blad sumaryczny
0.0005459464095875035 blad sumaryczny
0.0005402809999183715 blad sumaryczny
0.0005347309805192988 blad sumaryczny
0.0005292928747832265 blad sumaryczny
0.0005239633440825465 blad sumaryczny
0.0005187391810003849 blad sumaryczny
0.0005136173029561004 blad sumaryczny
0.0005085947461985 blad sumaryczny
0.0005036686601422595 blad sumaryczny
0.0004988363020249013 blad sumaryczny
wyjscie dla 1 0.977746282304652
wyjscie dla  4 0.022171235875895486
wyjscie dla niby 1 0.977746282304652
wyjscie dla niby 4 0.022171235875895486
```