EHB420E-ARTIFICIAL NEURAL NETWORKS

Mehmet Soner Korucu- 040190225

HOMEWORK #1

You can also reach the repo on my GitHub page: https://github.com/mkorucu/EHB420E.git

Problem 1:

```
userName = input("Enter name:")
userAge = input("Enter age:")
try:
    userAgeInt = int(userAge)
    print("Wellcome " + userName + ", you have " + str(100 - userAgeInt) + " years
to become 100 years old")
except ValueError:
    print("Invalid age input. Please enter a valid integer.")
```

Problem 2:

```
from datetime import datetime as dt

current_datetime = dt.now()
formatted_date = current_datetime.strftime("%d-%m-%Y %H:%M:%S")
print("Output1:\nTwinkle, twinkle, little star,\n\tHow I wonder what you
are!\n\t\tUp above the world so high,\n\t\tLike a diamond in the sky.\nTwinkle,
twinkle, little star,\n\tHow I wonder what you are!\nOutput2:\nCurrent date and
time :\n%s", formatted_date)
```

Problem 3:

```
import cmath

def quadratic_equation_calculator(a, b, c):
    delta = (b**2 - 4*a*c)
    root1 = (-b + cmath.sqrt(delta)) / (2*a)
    root2 = (-b - cmath.sqrt(delta)) / (2*a)
    return root1, root2

print("calculating roots of (x^2) - 5.86x + 8.5408")
roots = quadratic_equation_calculator(1, 5.86, 8.5408)
print("root 1: " + str(roots[0]))
print("root 2: " + str(roots[1]))
```

Problem 4:

```
'''
    Bitwise operations
'''
print("Bitwise operators are used to calculate bit-level operations. These
operations are useful when setting the registers of a microcontroller or its
peripherals.")
print("For example: & is bitwise AND operation. It ands every bit of 2 number and
returns the result.")
a = 5 #101
b = 6 #110
print("the result of a(5) & b(6) is equal to " + str(a & b)) # 101 & 110 => 1&1 ,
0&1 , 1&0 => 1,0,0 => 100 = 4
print("In the same manner, | means bitwise OR, ^ means bitwise XOR, ~ means bitwise
NOT, << shifts number to the left, >> shifts number to the right handside.")
print("a(5) | b(6) equals to " + str(a | b)) # 101 | 110 => 111 = 7
print("a(5) ^ b(6) equals to " + str(a ^ b)) # 101 ^ 110 => 011 = 3
print("bitwise not of a(5) equals to" + str(~a)) # 101 => 0101= 2
print("a(5) 2 times shifted to the left equals to " + str(a << 2)) # 101 << 2 =>
10100 = 20
print("b(6) 2 times shifted to the right equals to " + str(b >> 2)) # 110 >> 2 => 1
= 1


'''
Special operators
'''
print("Special operators are 'is' and 'is not' operators.")
print("These operators are used to compare the identity of two objects, which means
checking if the memory locations of two objects are same.")

c = [3,4,5]
d = 4
print("c and d are two different variable in memory. Therefore, their 'is'
operation is equal to " + str(c is d))


'''
Membership operators
'''
print("Membership operators are used to check if a value is inluded in a sequence
or collection, such as strings, lists, tuples, or sets. The membership operators
available in Python are \"in\" and \"not in\".")
text = "Hello, world!"
print("Hello" in text)  # True - "Hello" is present in the string
print("hi" not in text)  # True - "hi" is not present in the string
```