Spring 2024: CS5720 Neural Networks & Deep Learning – ICP3

Assignment-3

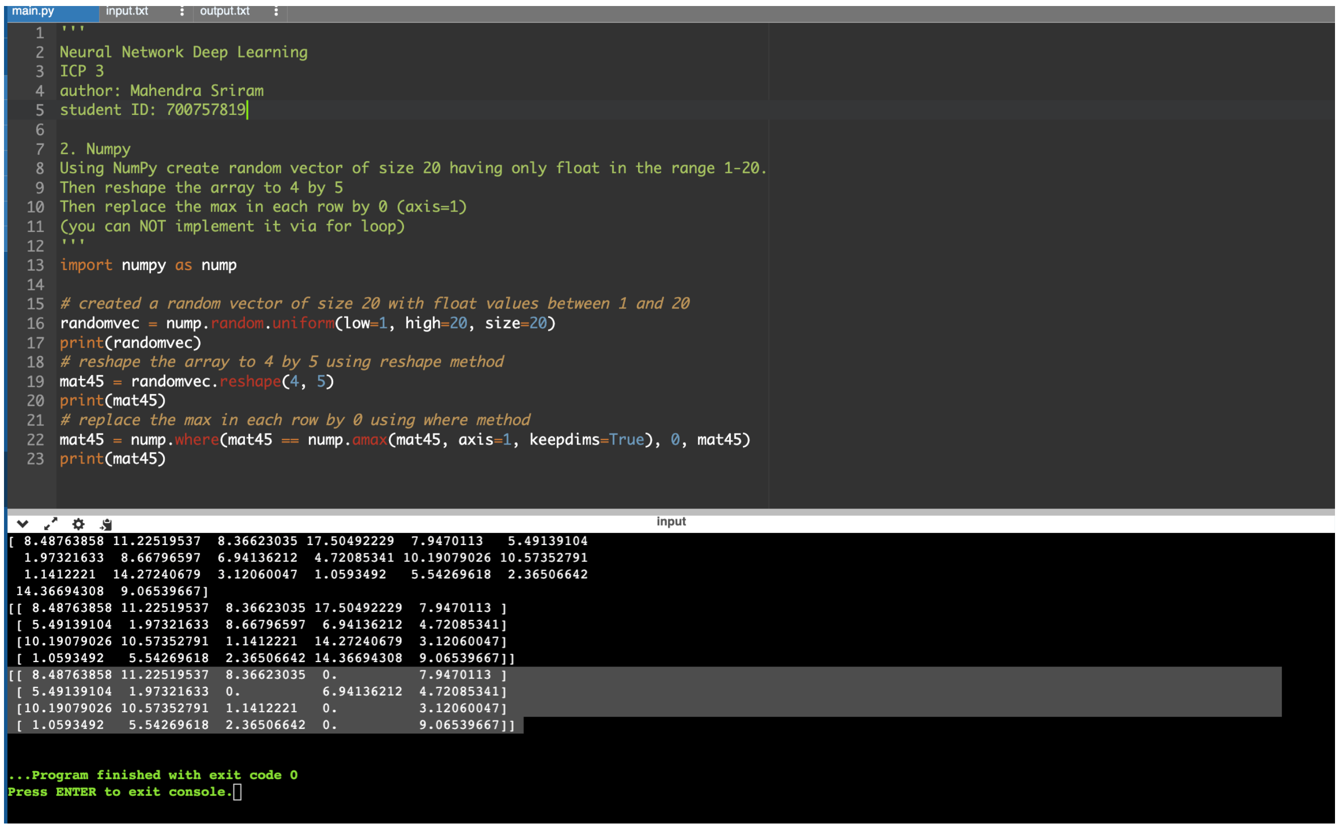
NAME: Mahendra Sriram Betha
STUDENT ID:700757819

Video Link:
https://drive.google.com/file/d/1oYx_GzD3E9qfh1Tkhi1x0MQtosOZLwpU/view?usp=drive_link

GitHub Link: https://github.com/mahendrasrirambetha/NN_Assignment3

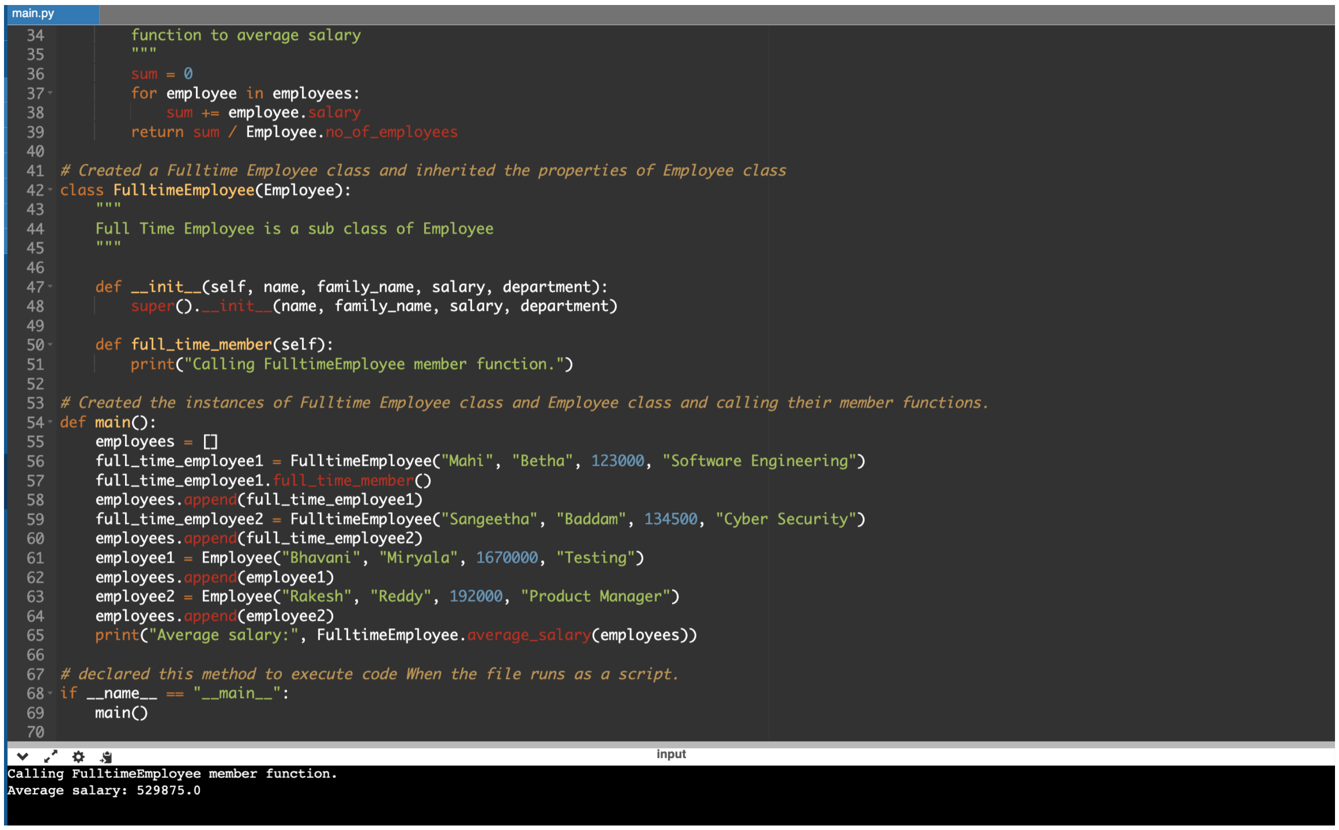
1. Create a class Employee and then do the following

• Create a data member to count the number of Employee
• Create a constructor to initialize name, family, salary, department
• Create a function to average salary
• Create a Fulltime Employee class and it should inherit the properties of Employee class
• Create the instances of Fulltime Employee class and Employee class and call their member functions.



```python
'''
Neural Network Deep Learning
ICP 3
author: Mahendra Sriram
student ID: 700757819

2. Numpy
Using NumPy create random vector of size 20 having only float in the range 1-20.
Then reshape the array to 4 by 5
Then replace the max in each row by 0 (axis=1)
(you can NOT implement it via for loop)
'''
import numpy as nump

# created a random vector of size 20 with float values between 1 and 20
randomvec = nump.random.uniform(low=1, high=20, size=20)
print(randomvec)
# reshape the array to 4 by 5 using reshape method
mat45 = randomvec.reshape(4, 5)
print(mat45)
# replace the max in each row by 0 using where method
mat45 = nump.where(mat45 == nump.amax(mat45, axis=1, keepdims=True), 0, mat45)
print(mat45)
```

```
[ 8.48763858 11.22519537  8.36623035 17.50492229  7.9470113   5.49139104
  1.97321633  8.66796597  6.94136212  4.72085341 10.19079026 10.57352791
  1.1412221  14.27240679  3.12060047  1.0593492   5.54269618  2.36506642
 14.36694308  9.06539667]
[[ 8.48763858 11.22519537  8.36623035 17.50492229  7.9470113 ]
 [ 5.49139104  1.97321633  8.66796597  6.94136212  4.72085341]
 [10.19079026 10.57352791  1.1412221  14.27240679  3.12060047]
 [ 1.0593492   5.54269618  2.36506642 14.36694308  9.06539667]]
[[ 8.48763858 11.22519537  8.36623035  0.          7.9470113 ]
 [ 5.49139104  1.97321633  0.          6.94136212  4.72085341]
 [10.19079026 10.57352791  1.1412221   0.          3.12060047]
 [ 1.0593492   5.54269618  2.36506642  0.          9.06539667]]


...Program finished with exit code 0
Press ENTER to exit console.
```



main.py

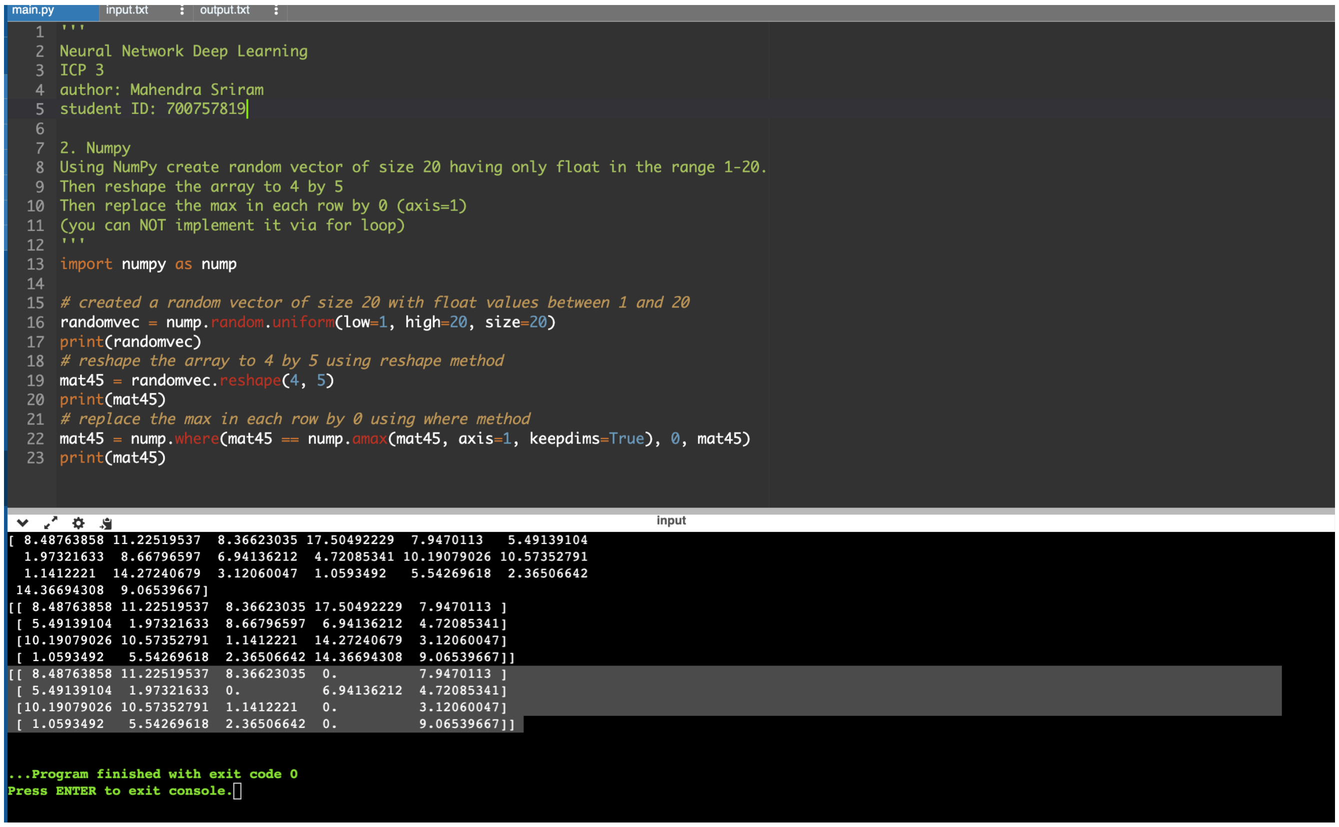
```python
        function to average salary
        """
        sum = 0
        for employee in employees:
            sum += employee.salary
        return sum / Employee.no_of_employees

# Created a Fulltime Employee class and inherited the properties of Employee class
class FulltimeEmployee(Employee):
    """
    Full Time Employee is a sub class of Employee
    """

    def __init__(self, name, family_name, salary, department):
        super().__init__(name, family_name, salary, department)

    def full_time_member(self):
        print("Calling FulltimeEmployee member function.")

# Created the instances of Fulltime Employee class and Employee class and calling their member functions.
def main():
    employees = []
    full_time_employee1 = FulltimeEmployee("Mahi", "Betha", 123000, "Software Engineering")
    full_time_employee1.full_time_member()
    employees.append(full_time_employee1)
    full_time_employee2 = FulltimeEmployee("Sangeetha", "Baddam", 134500, "Cyber Security")
    employees.append(full_time_employee2)
    employee1 = Employee("Bhavani", "Miryala", 1670000, "Testing")
    employees.append(employee1)
    employee2 = Employee("Rakesh", "Reddy", 192000, "Product Manager")
    employees.append(employee2)
    print("Average salary:", FulltimeEmployee.average_salary(employees))

# declared this method to execute code When the file runs as a script.
if __name__ == "__main__":
    main()

```

input

```
Calling FulltimeEmployee member function.
Average salary: 529875.0
```

2. NumPy

Using NumPy create random vector of size 20 having only float in the range 1-20. Then reshape the array to 4 by 5 Then replace the max in each row by 0 (axis=1) (you can NOT implement it via for loop)



```python
'''
Neural Network Deep Learning
ICP 3
author: Mahendra Sriram
student ID: 700757819

2. Numpy
Using NumPy create random vector of size 20 having only float in the range 1-20.
Then reshape the array to 4 by 5
Then replace the max in each row by 0 (axis=1)
(you can NOT implement it via for loop)
'''
import numpy as nump

# created a random vector of size 20 with float values between 1 and 20
randomvec = nump.random.uniform(low=1, high=20, size=20)
print(randomvec)
# reshape the array to 4 by 5 using reshape method
mat45 = randomvec.reshape(4, 5)
print(mat45)
# replace the max in each row by 0 using where method
mat45 = nump.where(mat45 == nump.amax(mat45, axis=1, keepdims=True), 0, mat45)
print(mat45)
```

```
[ 8.48763858 11.22519537  8.36623035 17.50492229  7.9470113   5.49139104
  1.97321633  8.66796597  6.94136212  4.72085341 10.19079026 10.57352791
  1.1412221  14.27240679  3.12060047  1.0593492   5.54269618  2.36506642
 14.36694308  9.06539667]
[[ 8.48763858 11.22519537  8.36623035 17.50492229  7.9470113 ]
 [ 5.49139104  1.97321633  8.66796597  6.94136212  4.72085341]
 [10.19079026 10.57352791  1.1412221  14.27240679  3.12060047]
 [ 1.0593492   5.54269618  2.36506642 14.36694308  9.06539667]]
[[ 8.48763858 11.22519537  8.36623035  0.          7.9470113 ]
 [ 5.49139104  1.97321633  0.          6.94136212  4.72085341]
 [10.19079026 10.57352791  1.1412221   0.          3.12060047]
 [ 1.0593492   5.54269618  2.36506642  0.          9.06539667]]


...Program finished with exit code 0
Press ENTER to exit console.
```