

DATA STRUCTURE AND ALGORITHM



LAB REPORT

Name	AREEBA FAROOQ
Registration Number	200901058
Batch & Section	BSCS 01 (SECTION A)
Instructor's Name	Sir Nadeem

DATE:25-10-2021

TASK 1

QUEUE IMPLEMENTATION USING NUMPY

SOLUTION

```
print("Queue implementation using Numpy")

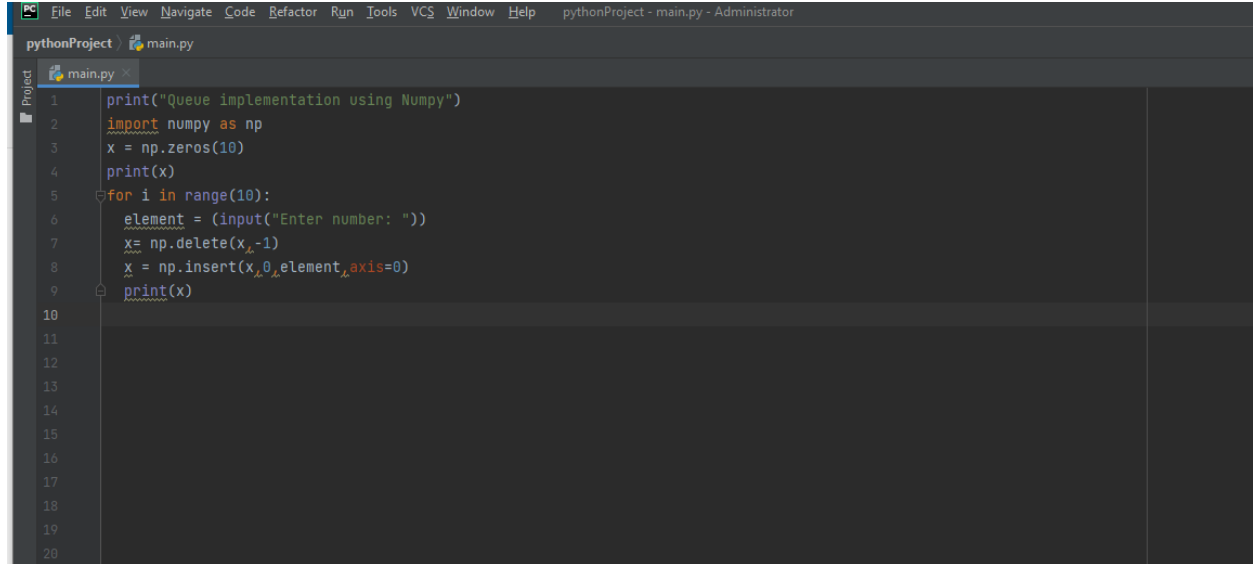
import numpy as np

x = np.zeros(5)

print(x)

for i in range(10):
    element = (input("Enter number: "))
    x= np.delete(x,-1)
    x = np.insert(x,0,element,axis=0)

print(x)
```

A screenshot of a code editor window titled 'pythonProject - main.py - Administrator'. The editor shows the same Python code as the previous block, with line numbers 1 through 20 on the left margin. The code implements a queue using NumPy arrays. The initial array 'x' is of size 5, but the loop runs 10 times, deleting the last element and inserting a new one at the beginning each time. The code is as follows:

```
1 print("Queue implementation using Numpy")
2 import numpy as np
3 x = np.zeros(10)
4 print(x)
5 for i in range(10):
6     element = (input("Enter number: "))
7     x= np.delete(x,-1)
8     x = np.insert(x,0,element,axis=0)
9     print(x)
10
11
12
13
14
15
16
17
18
19
20
```

```
Run: main
C:\Users\User\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/User/PycharmProjects/pythonProject/main.py
Queue implementation using Numpy
[0. 0. 0. 0. 0. 0. 0. 0. 0.]
Enter number: 1
[1. 0. 0. 0. 0. 0. 0. 0. 0.]
Enter number: 2
[2. 1. 0. 0. 0. 0. 0. 0. 0.]
Enter number: 3
[3. 2. 1. 0. 0. 0. 0. 0. 0.]
Enter number: 4
[4. 3. 2. 1. 0. 0. 0. 0. 0.]
Enter number: 5
[5. 4. 3. 2. 1. 0. 0. 0. 0.]
Enter number: 6
[6. 5. 4. 3. 2. 1. 0. 0. 0.]
Enter number: 7
[7. 6. 5. 4. 3. 2. 1. 0. 0.]
Enter number: 8
[8. 7. 6. 5. 4. 3. 2. 1. 0.]
Enter number: 9
[9. 8. 7. 6. 5. 4. 3. 2. 1.]
Enter number: 5
[5. 9. 8. 7. 6. 5. 4. 3. 2. 1.]
Process finished with exit code 0
```

QUEUE IMPLEMENTATION USING COLLECTION

SOLUTION

```
print("Queue implementation using Collections")

from collections import deque

x = deque()

print(x)

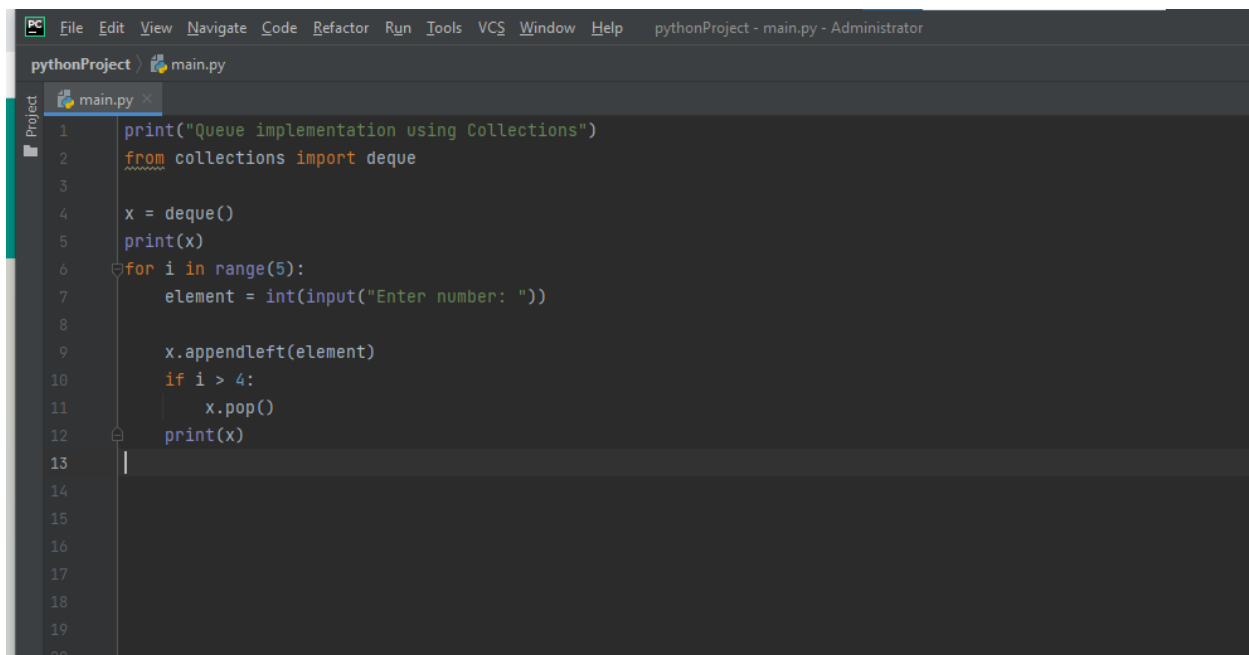
for i in range(5):
    element = int(input("Enter number: "))

    x.appendleft(element)

    if i > 4:
        x.pop()

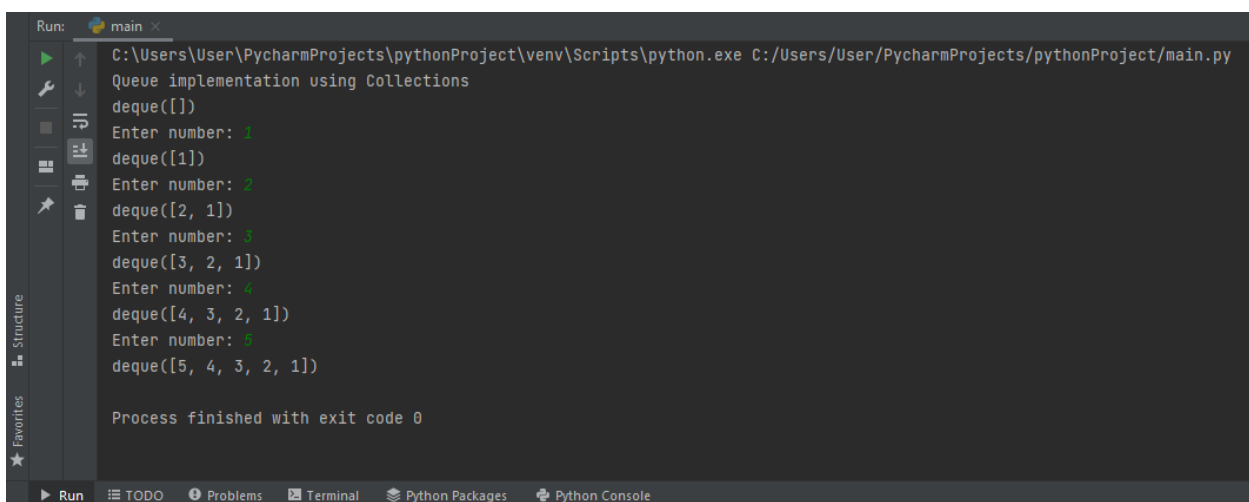
print(x)
```

OUTPUT



The screenshot shows the PyCharm IDE with a file named `main.py` open. The code implements a queue using `collections.deque`. It starts by printing a message, then imports `deque` from `collections`. A `deque` object `x` is created and printed. A `for` loop runs 5 times, each time prompting the user to enter a number. The number is added to the left of the deque (`x.appendleft(element)`). If the deque's length exceeds 4, the element at the right end is removed (`x.pop()`). The deque is printed after each iteration.

```
1 print("Queue implementation using Collections")
2 from collections import deque
3
4 x = deque()
5 print(x)
6 for i in range(5):
7     element = int(input("Enter number: "))
8
9     x.appendleft(element)
10    if i > 4:
11        x.pop()
12    print(x)
13
14
15
16
17
18
19
20
```



The screenshot shows the Run console in PyCharm. The command executed is `C:\Users\User\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/User/PycharmProjects/pythonProject/main.py`. The output shows the program's execution, including the initial deque state, user input, and the deque's state after each iteration. The program finishes with an exit code of 0.

```
Run: main
C:\Users\User\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/User/PycharmProjects/pythonProject/main.py
Queue implementation using Collections
deque([])
Enter number: 1
deque([1])
Enter number: 2
deque([2, 1])
Enter number: 3
deque([3, 2, 1])
Enter number: 4
deque([4, 3, 2, 1])
Enter number: 5
deque([5, 4, 3, 2, 1])

Process finished with exit code 0
```