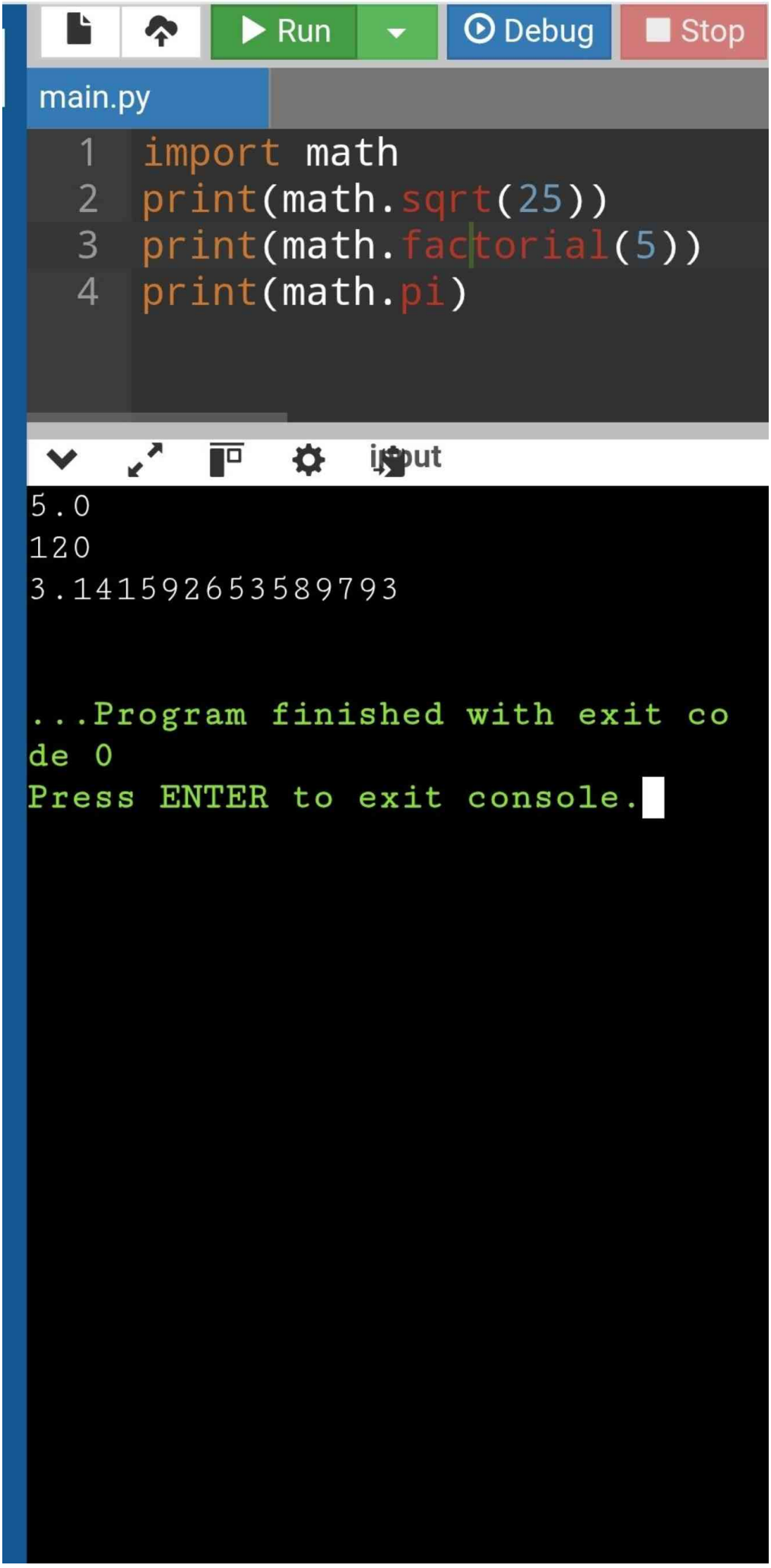
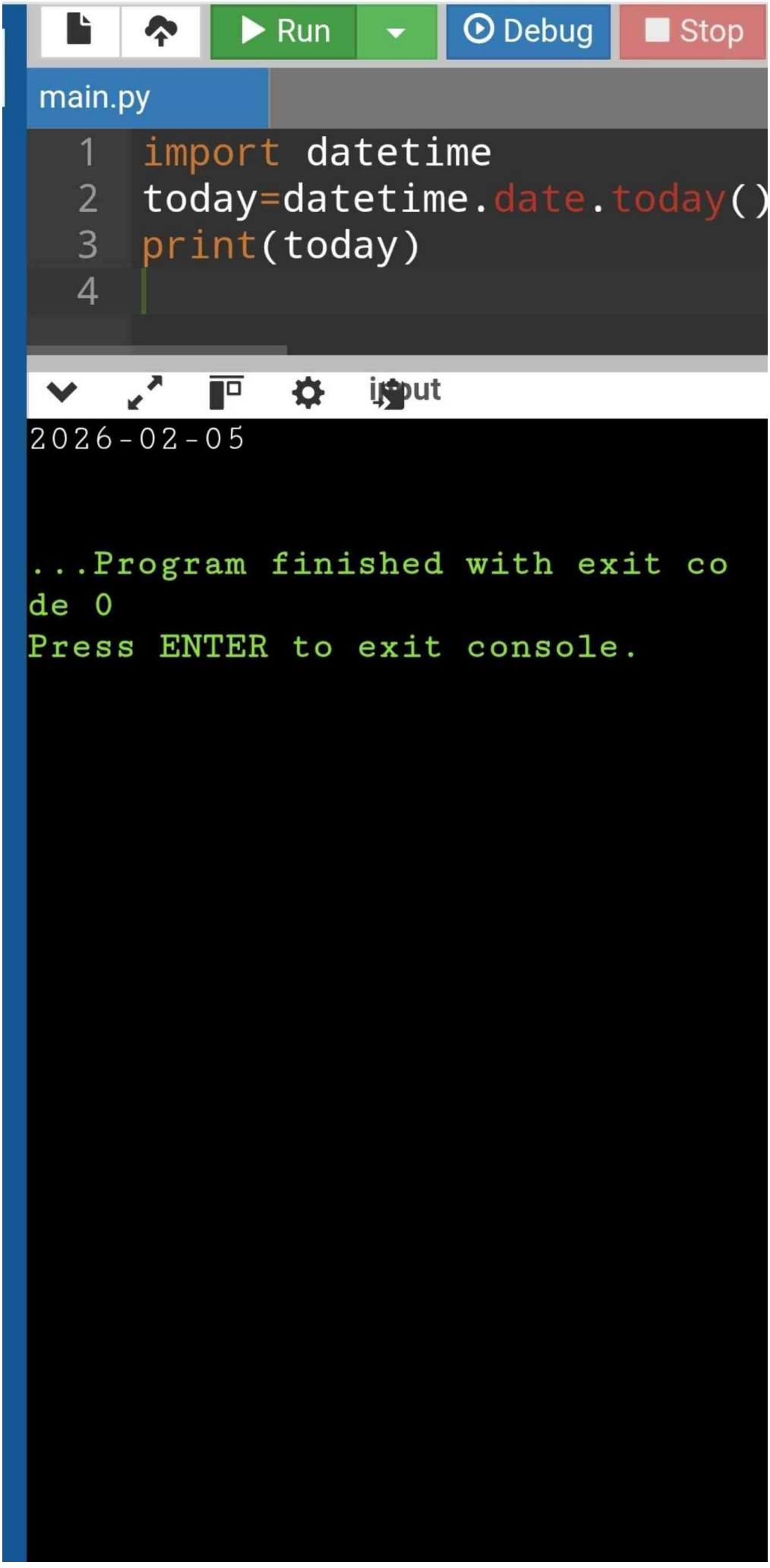
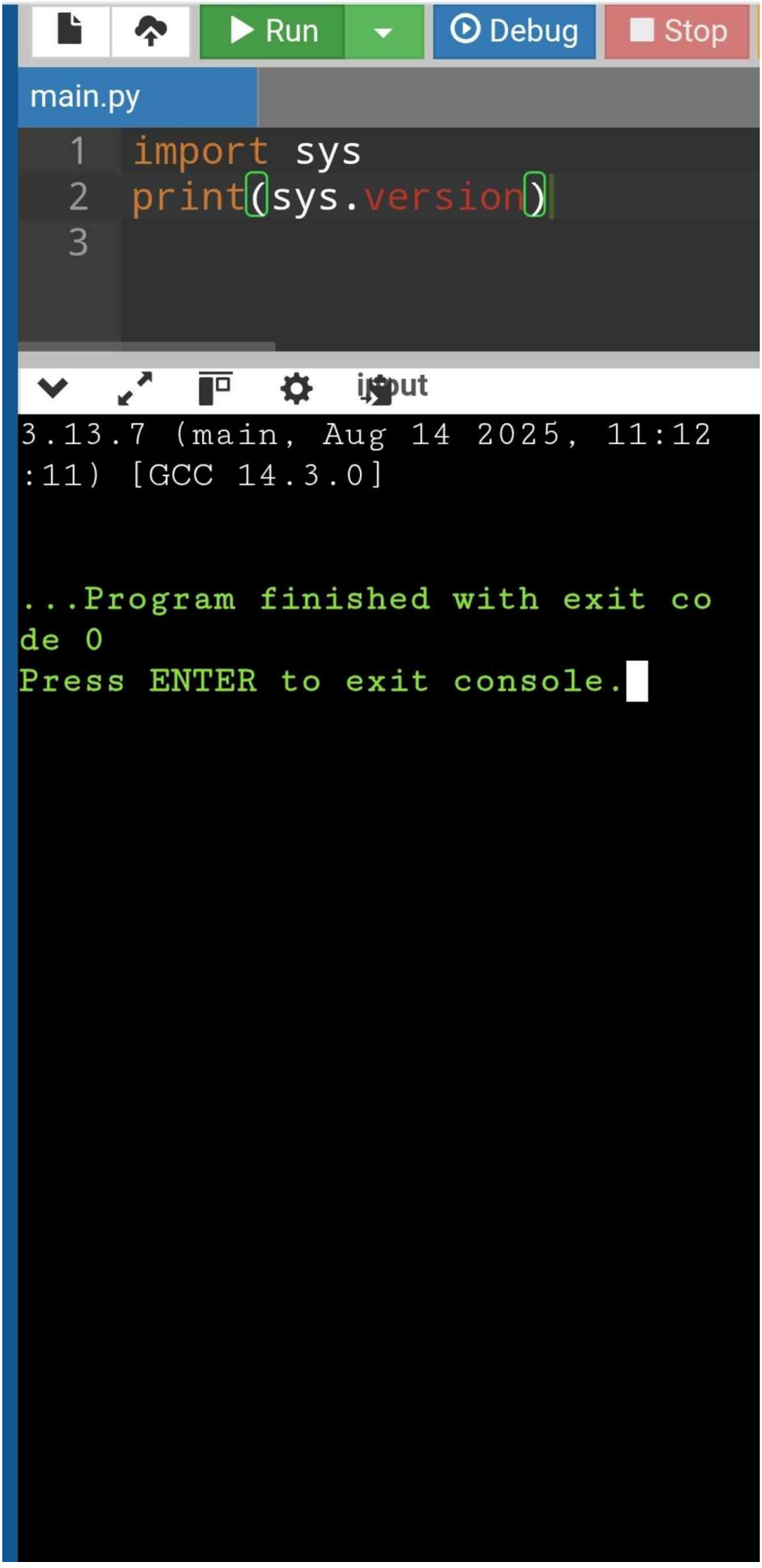
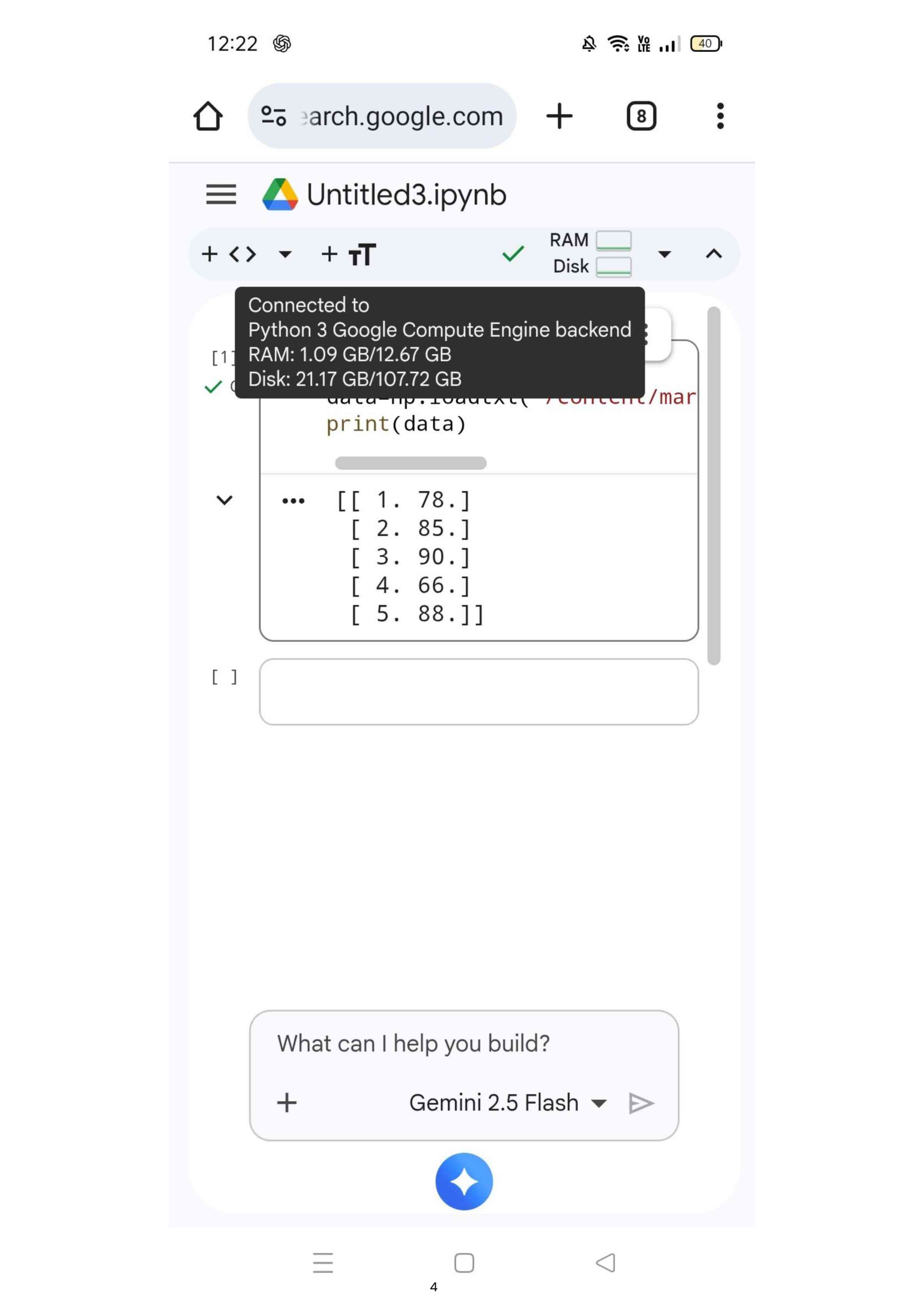
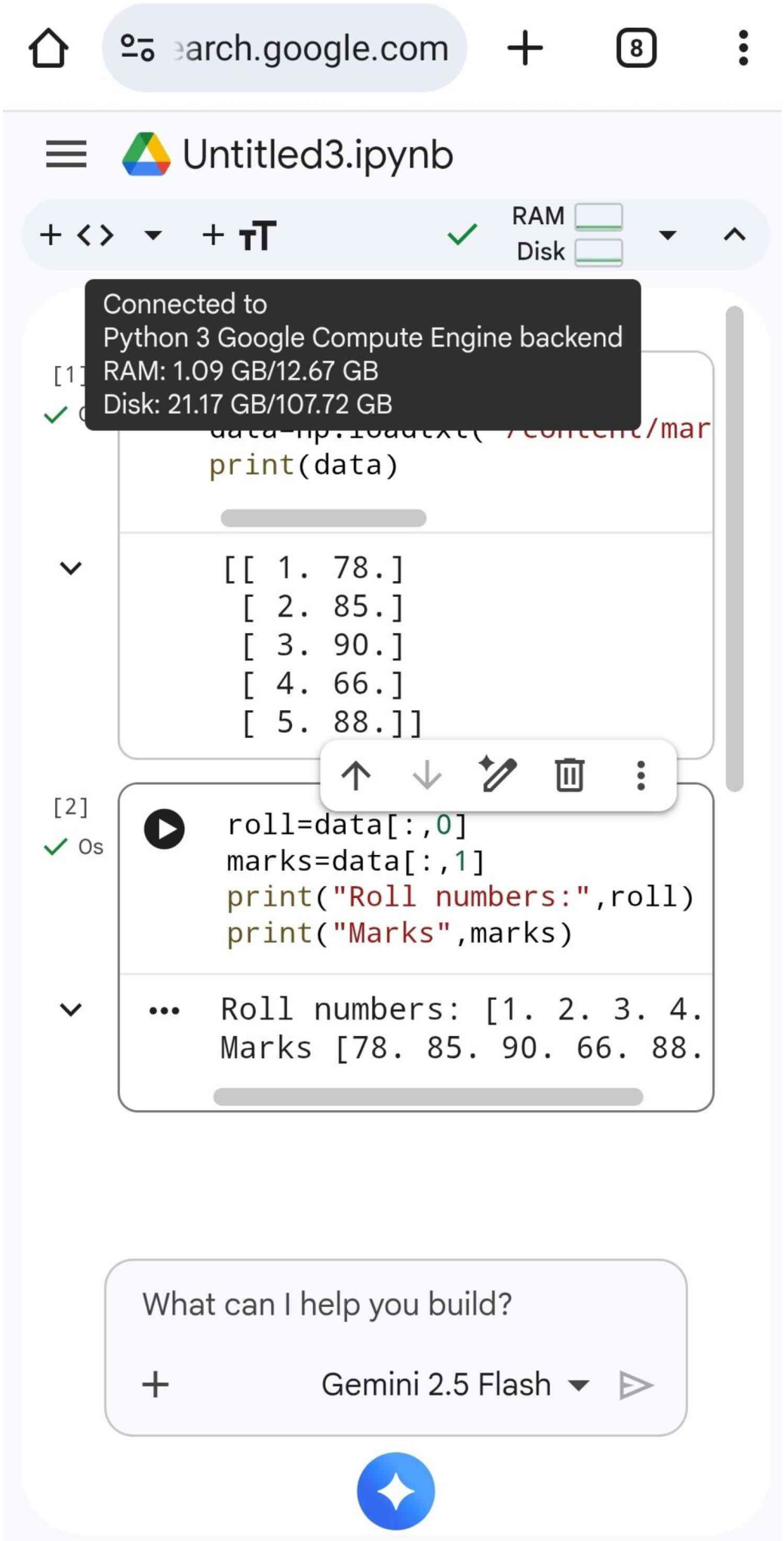
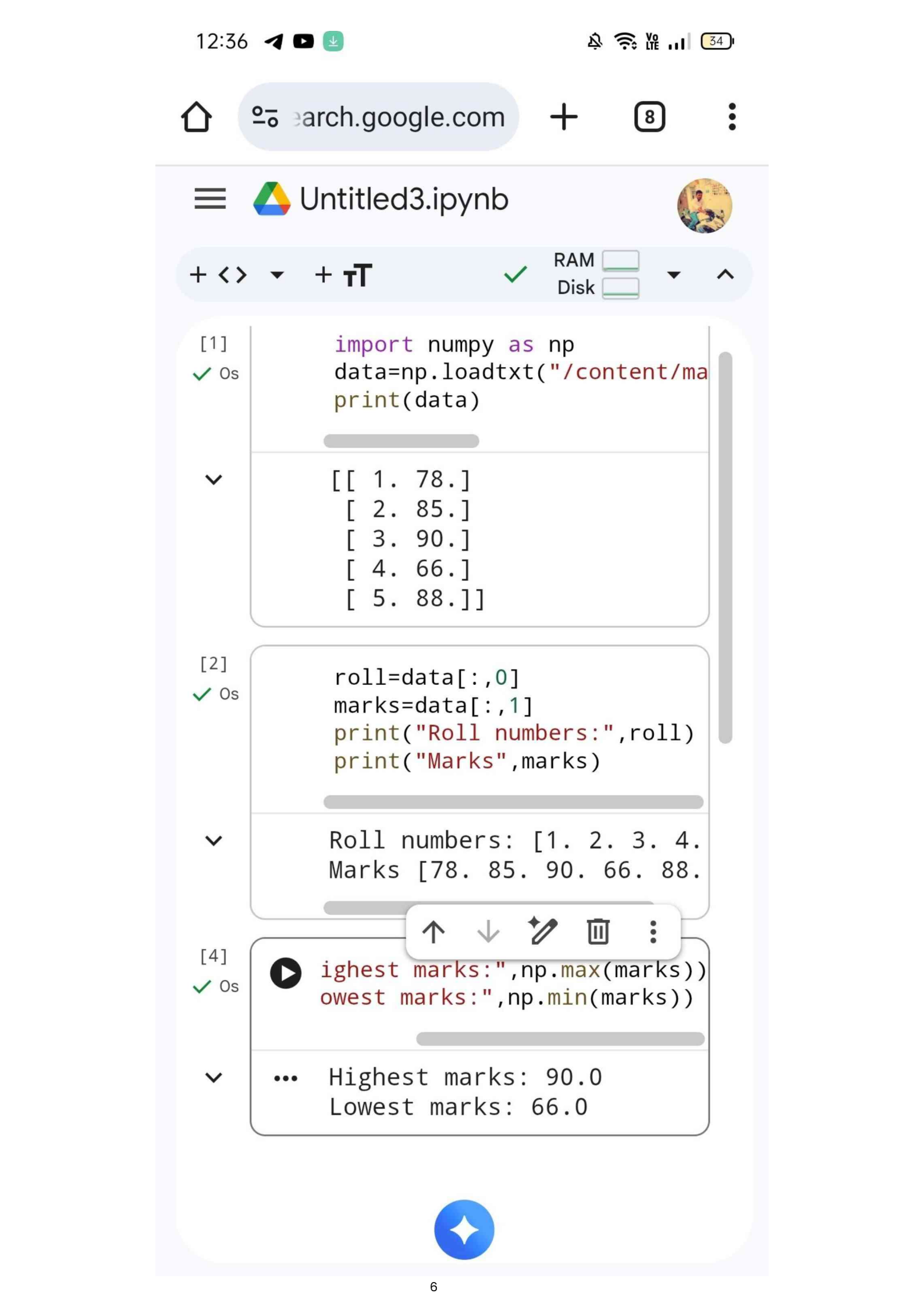

Run
Debug
Stop
main.py
1 import math
2 print(math.sqrt(25))
3 print(math.factorial(5))
4 print(math.pi)
input
5.0
120
3.141592653589793
...Program finished with exit co
de 0
Press ENTER to exit console.

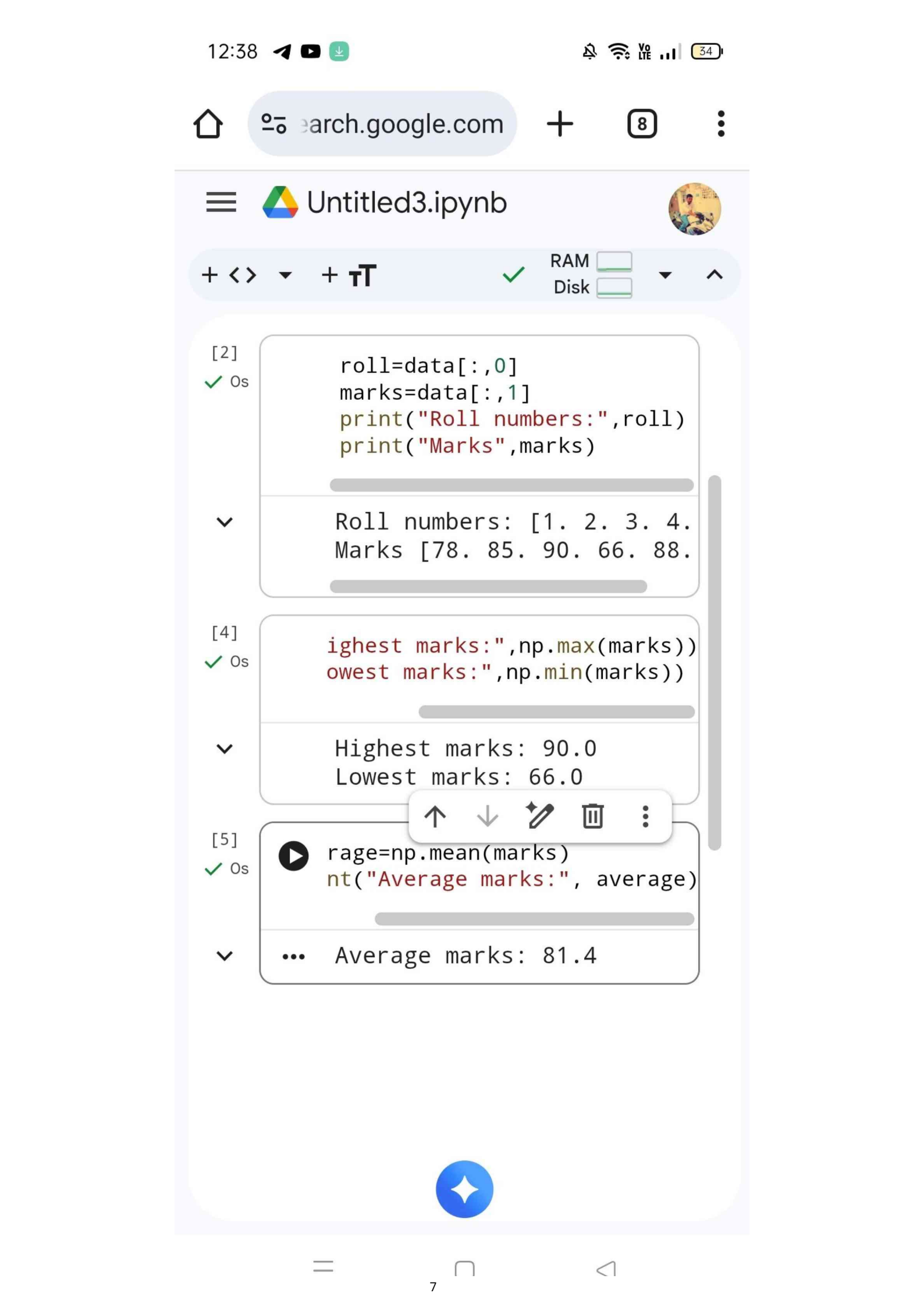

Run
Debug
Stop
main.py
1 import datetime
2 today=datetime.date.today()
3 print(today)
4
input
2026-02-05
...Program finished with exit co
de 0
Press ENTER to exit console.

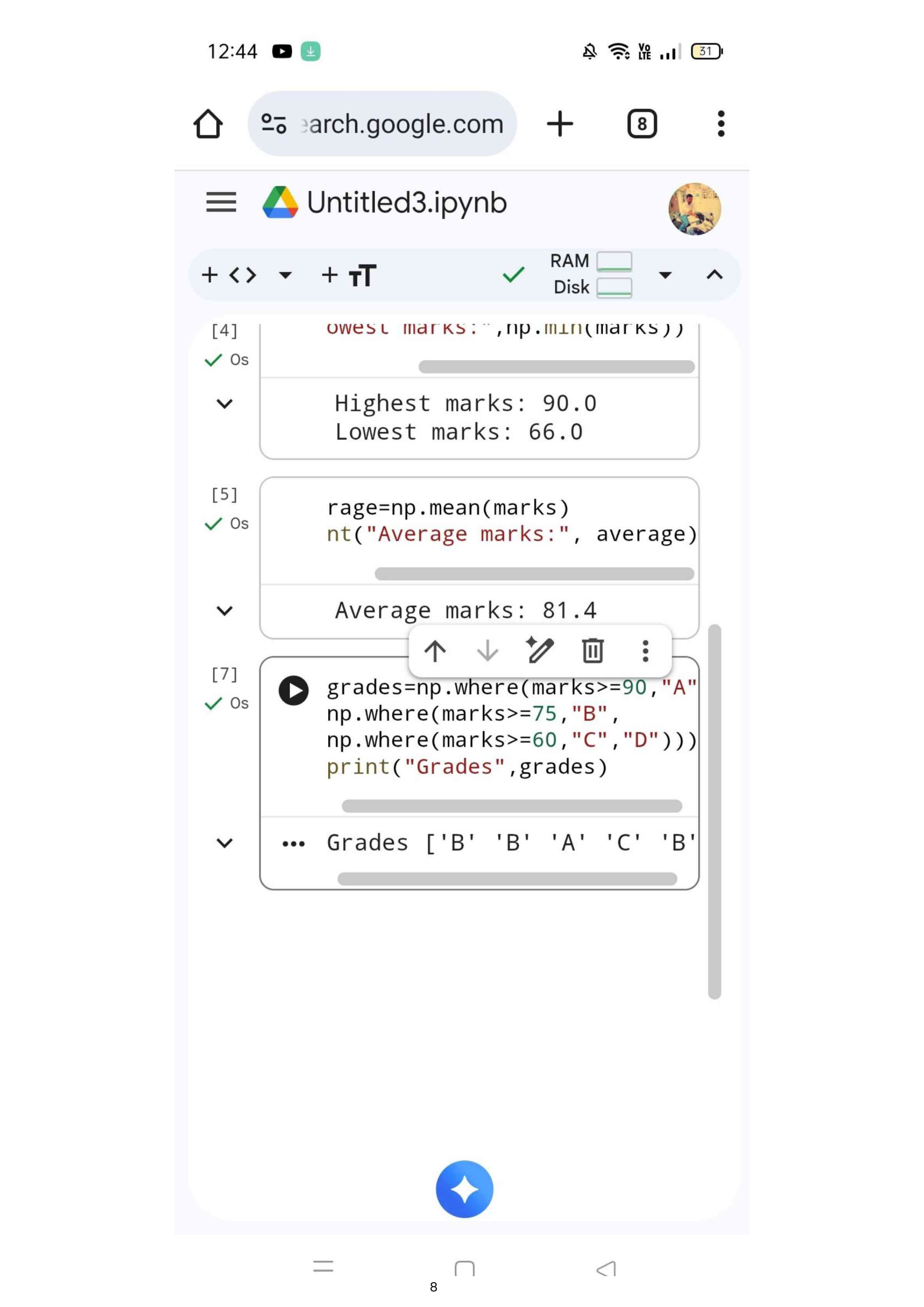

Run
Debug
Stop
main.py
1 import sys
2 print(sys.version)
3
input
3.13.7 (main, Aug 14 2025, 11:12
:11) [GCC 14.3.0]
...Program finished with exit co
de 0
Press ENTER to exit console.

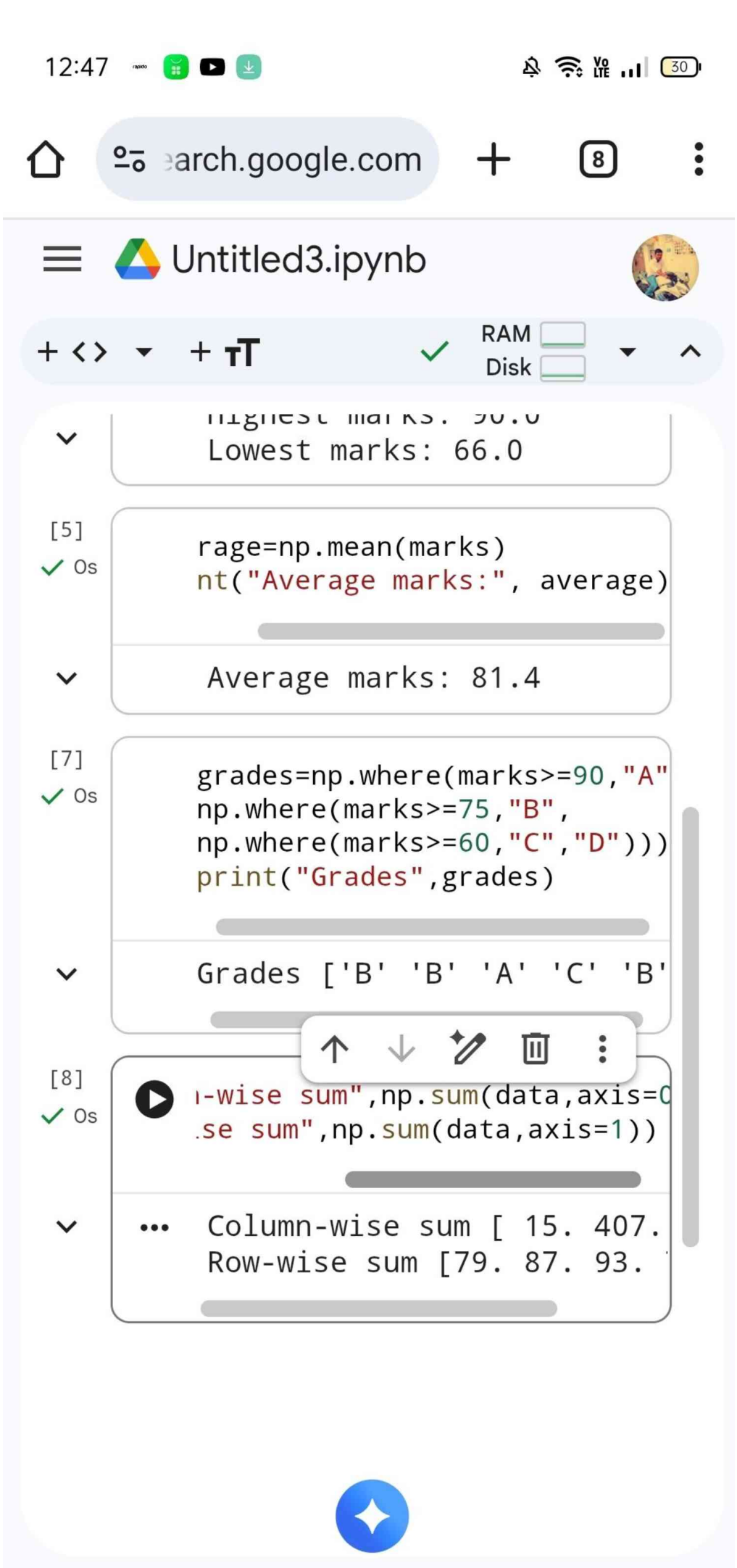

12:22
earch.google.com
Untitled3.ipynb
RAM
Disk
Connected to
Python 3 Google Compute Engine backend
RAM: 1.09 GB/12.67 GB
Disk: 21.17 GB/107.72 GB
[1]
print(data)
[[1. 78.]
[2. 85.]
[3. 90.]
[4. 66.]
[5. 88.]]
[ ]
What can I help you build?
Gemini 2.5 Flash


arch.google.com
8
Untitled3.ipynb
RAM
Disk
Connected to
Python 3 Google Compute Engine backend
RAM: 1.09 GB/12.67 GB
Disk: 21.17 GB/107.72 GB
[1]
print(data)
[[1. 78.]
[2. 85.]
[3. 90.]
[4. 66.]
[5. 88.]]
[2]
0s
roll=data[:,0]
marks=data[:,1]
print("Roll numbers:",roll)
print("Marks",marks)
Roll numbers: [1. 2. 3. 4.
Marks [78. 85. 90. 66. 88.
What can I help you build?
Gemini 2.5 Flash


12:36
arch.google.com
8
Untitled3.ipynb
RAM
Disk
[1]
0s
import numpy as np
data=np.loadtxt("/content/ma
print(data)
[[1. 78.]
[2. 85.]
[3. 90.]
[4. 66.]
[5. 88.]]
[2]
0s
roll=data[:,0]
marks=data[:,1]
print("Roll numbers:",roll)
print("Marks",marks)
Roll numbers: [1. 2. 3. 4.
Marks [78. 85. 90. 66. 88.
[4]
0s
ighest marks:",np.max(marks))
owest marks:",np.min(marks))
Highest marks: 90.0
Lowest marks: 66.0


12:38
arch.google.com
Untitled3.ipynb
RAM
Disk
[2]
0s
roll=data[:,0]
marks=data[:,1]
print("Roll numbers:",roll)
print("Marks",marks)
Roll numbers: [1. 2. 3. 4.
Marks [78. 85. 90. 66. 88.
[4]
0s
ighest marks:",np.max(marks))
owest marks:",np.min(marks))
Highest marks: 90.0
Lowest marks: 66.0
[5]
0s
rage=np.mean(marks)
nt("Average marks:", average)
Average marks: 81.4


12:44
arch.google.com
Untitled3.ipynb
RAM
Disk
[4]
0s
owest marks:",np.min(marks))
Highest marks: 90.0
Lowest marks: 66.0
[5]
0s
rage=np.mean(marks)
nt("Average marks:", average)
Average marks: 81.4
[7]
0s
grades=np.where(marks>=90,"A"
np.where(marks>=75,"B",
np.where(marks>=60,"C","D")))
print("Grades",grades)
Grades ['B' 'B' 'A' 'C' 'B'


12:47
arch.google.com
Untitled3.ipynb
RAM
Disk
Lowest marks: 66.0
[5]
Os
rage=np.mean(marks)
nt("Average marks:", average)
Average marks: 81.4
[7]
Os
grades=np.where(marks>=90,"A"
np.where(marks>=75,"B",
np.where(marks>=60,"C","D")))
print("Grades",grades)
Grades ['B' 'B' 'A' 'C' 'B'
[8]
Os
-wise sum",np.sum(data,axis=0
se sum",np.sum(data,axis=1))
Column-wise sum [15. 407.
Row-wise sum [79. 87. 93.

y

```
import numpy as np
arr = np.array([10, 15, 22, 33,
even=np.sum(arr % 2 == 0)
odd=np.sum(arr % 2 != 0)
print("Array:", arr)
print("Even numbers:", even)
print("Odd numbers:", odd)
```

input

```
Array: [10 15 22 33 40 55 60]
Even numbers: 4
Odd numbers: 3


...Program finished with exit co
de 0
Press ENTER to exit console.
```