**ES215: Computer Organisation and Architecture**

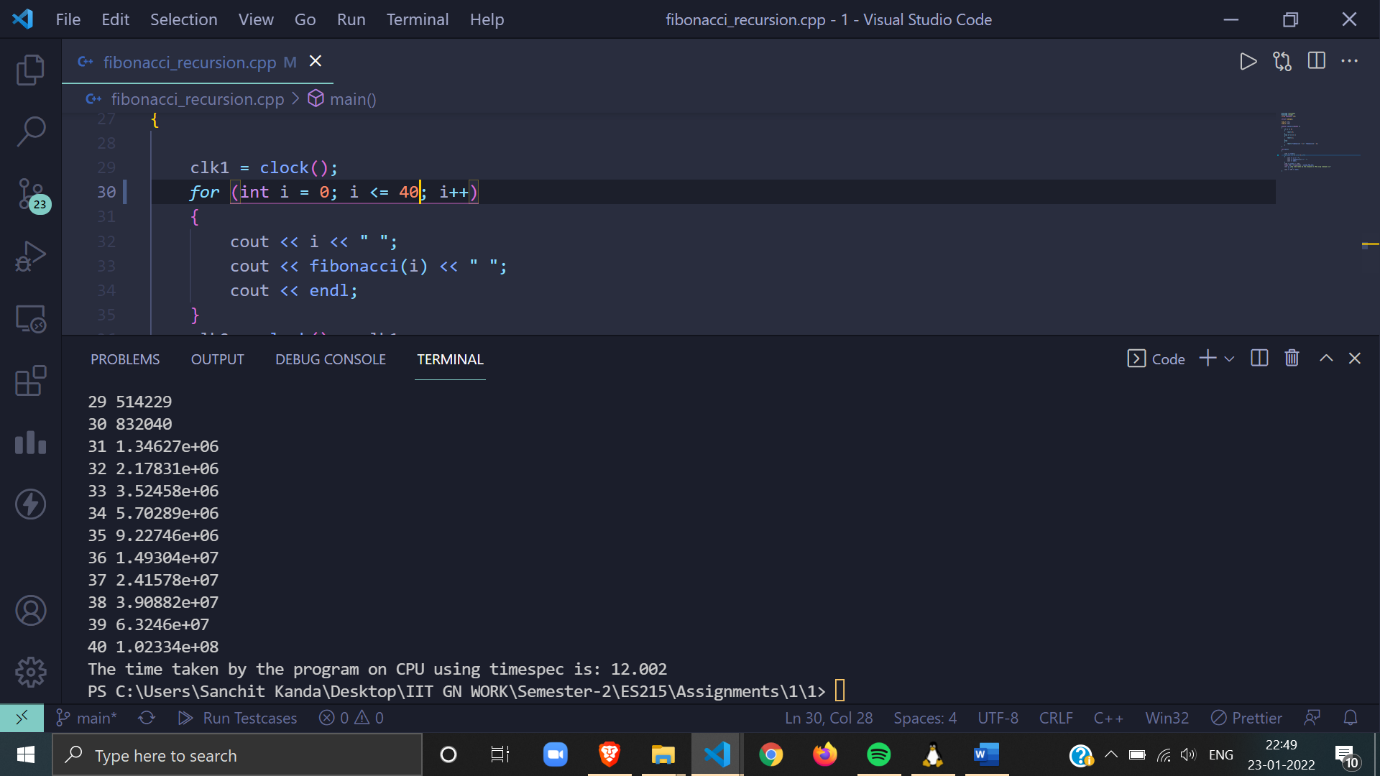
(https://github.com/Madhav-Kanda/ES215\_Assignment\_1)

**Q1.**

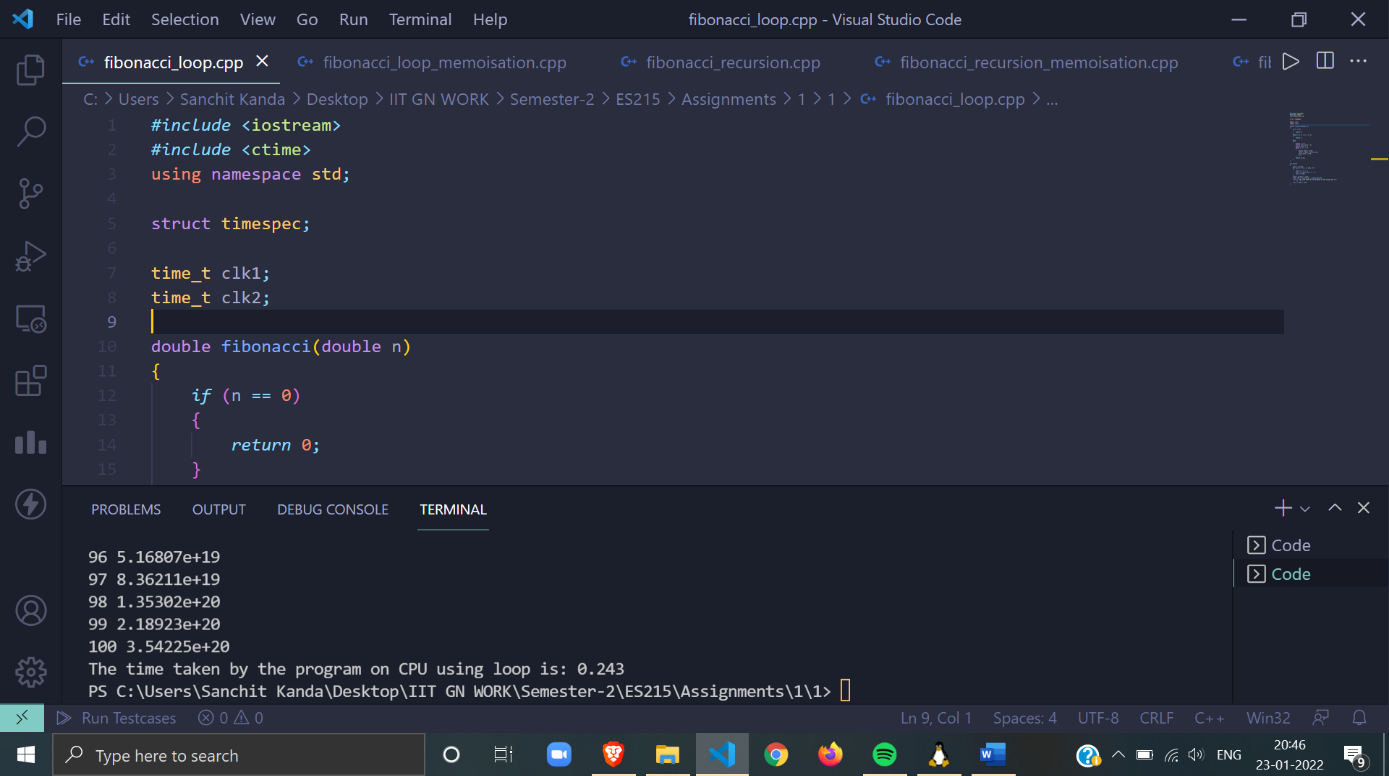
1. Time taken using recursion: **133,605.45 sec**

Time taken for first 40 fibonacci =**12.002 sec**

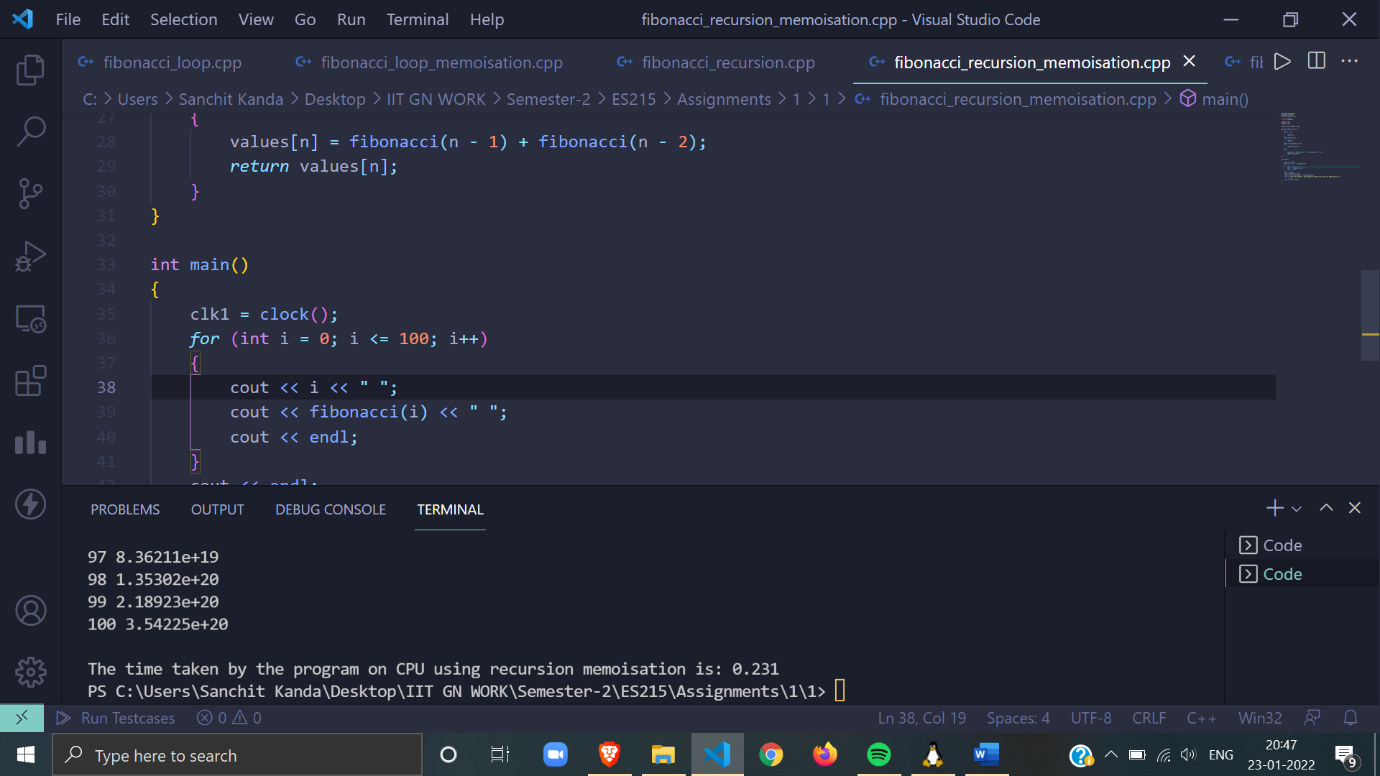
Formula to calculate time taken for 100 fibonacci = ((1.168)^(60)))\*(12.002) sec



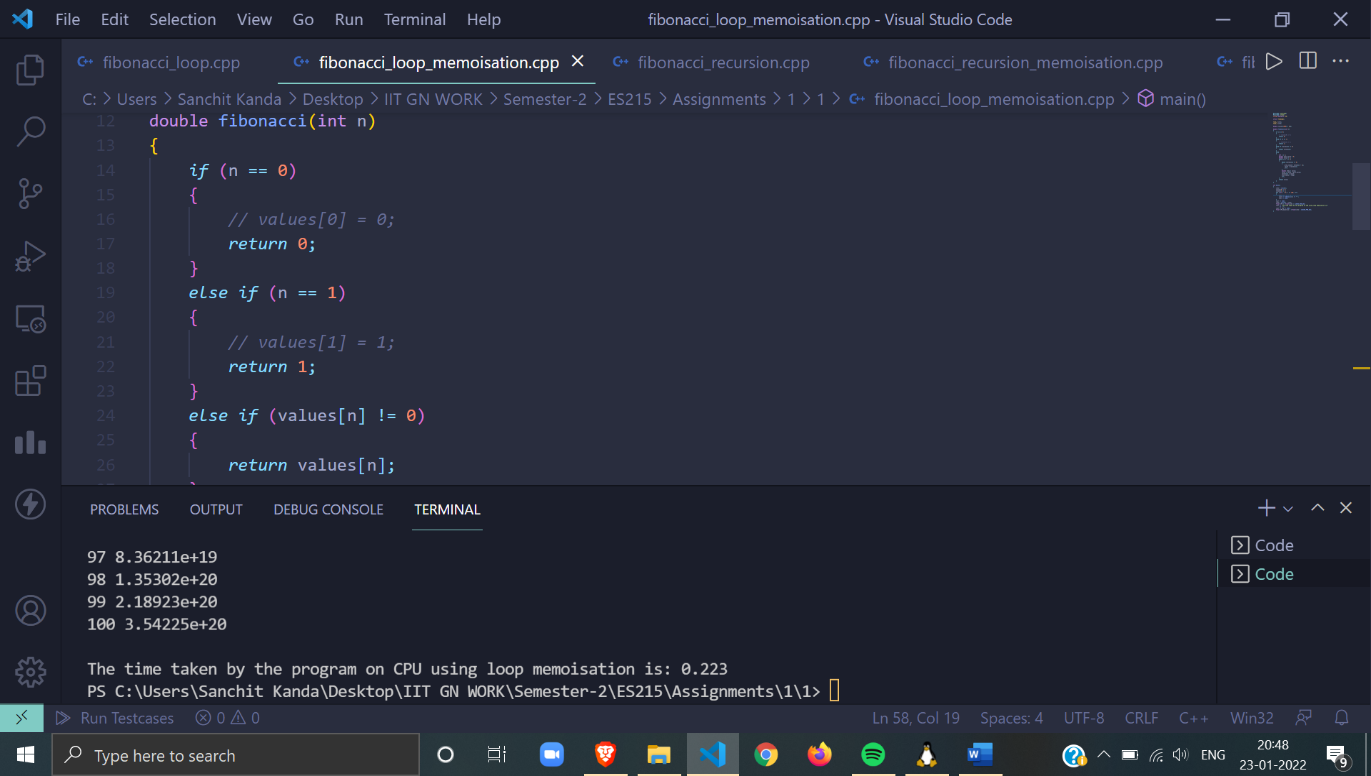
1. Time taken using loop: **0.243s**



1. Time taken using recursion and memoization: **0.231s**

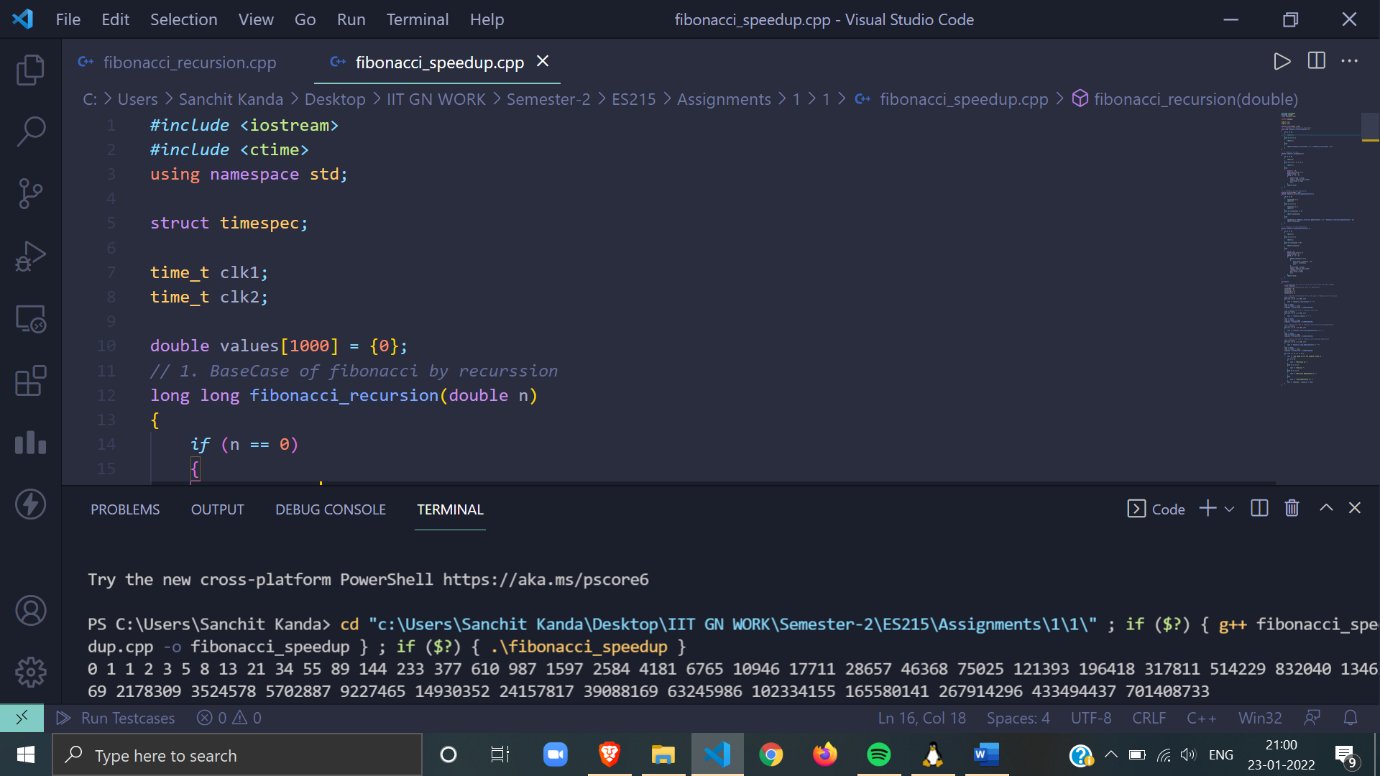


1. Time taken using loop memoization: **0.223s**



Speedup of all the program:

1. Speedup for recursion is: (Time taken using recursion/Time taken using recursion)= 1
2. Speedup for loop is: (Time taken using recursion/ Time taken using loop) =
3. Speedup for recursion and memoisation is:
4. Speedup for loop memoisation is:

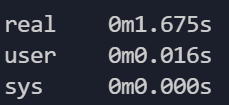


**Q2.**

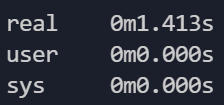
1. **Output Time**

**CPU Time=user +sys**

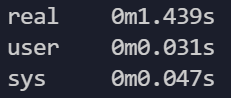
1. **For C++ using Double**
2. N=32



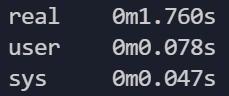
1. N=64



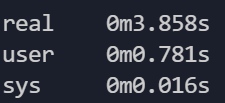
1. N=128



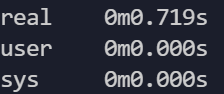
1. N=256



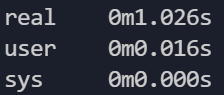
1. N=512



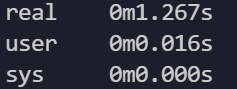
1. **For C++ using Integer**
2. N=32



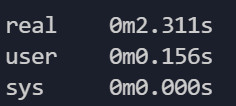
1. N=64



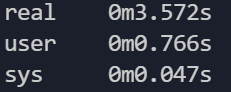
1. N=128



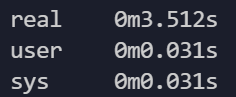
1. N=256



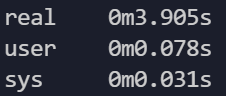
1. N=512



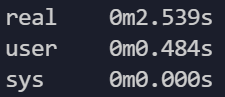
1. **For Python using integer**
2. N=32



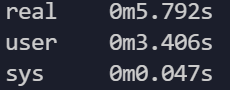
1. N=64



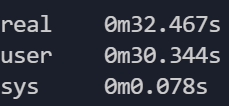
1. N=128



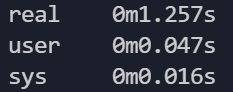
1. N=256



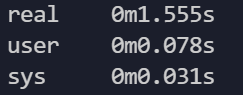
1. N=512



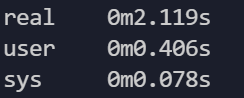
1. **For Python using Double**
2. N=32



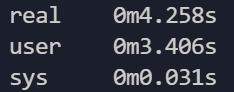
1. N=64



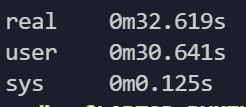
1. N=128



1. N=256



1. N=512



**B)**

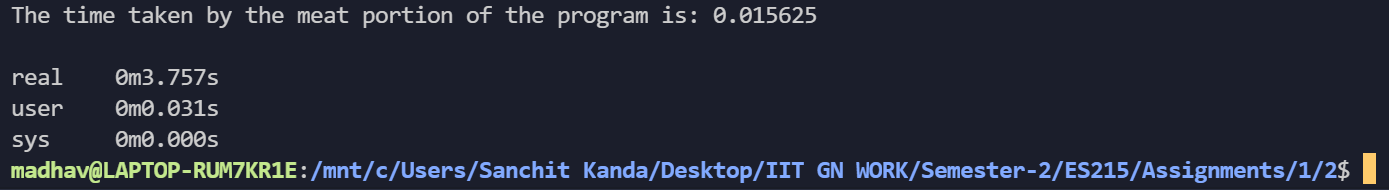
a) **C++ program for integer**

1. N=32

Execution Time: 3.757s

Meat Portion: 0.015625s

(Meat Portion/Execution Time): 0.00415

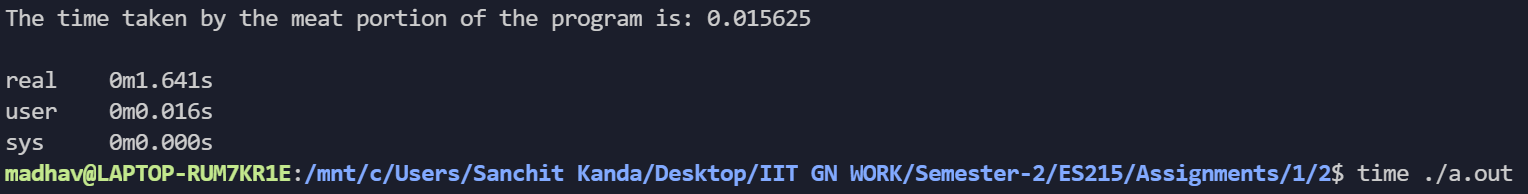


2. N=64

Execution Time: 1.641s

Meat Portion: 0.015625s

(Meat Portion/Execution Time): 0.0.009521

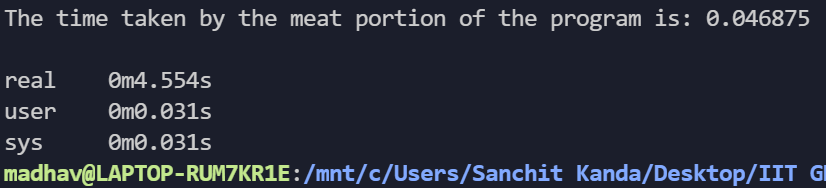


3. N=128

Execution Time: 4.554s

Meat Portion: 0.046875s

(Meat Portion/Execution Time): 0.0102

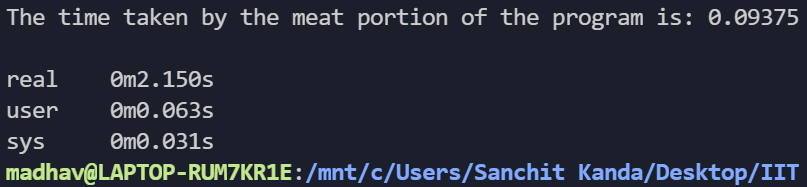


4. N=256

Execution Time: 2.150s

Meat Portion: 0.09375s

(Meat Portion/Execution Time): 0.043

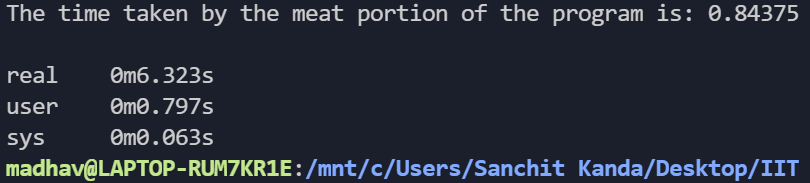


5. N=512

Execution Time: 6.323s

Meat Portion: 0.84375s

(Meat Portion/Execution Time): 0.133



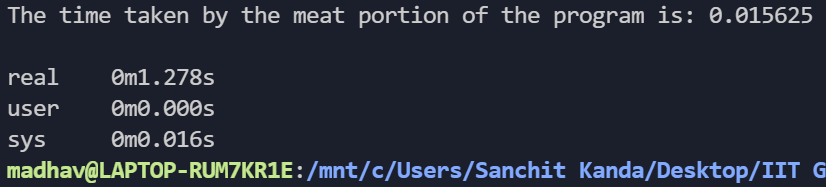
b) **C++ program for double**

1. N=32

Execution Time: 1.278s

Meat Portion: 0.015625s

(Meat Portion/Execution Time): 0.1222

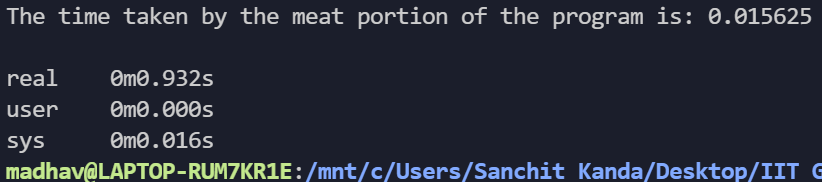


2. N=64

Execution Time: 0.932s

Meat Portion: 0.015625s

(Meat Portion/Execution Time): 0.0167

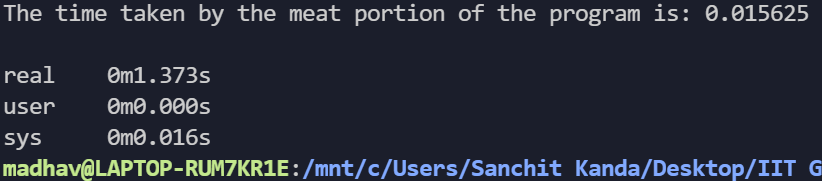


3. N=128

Execution Time: 1.373s

Meat Portion: 0.015625s

(Meat Portion/Execution Time): 0.0113

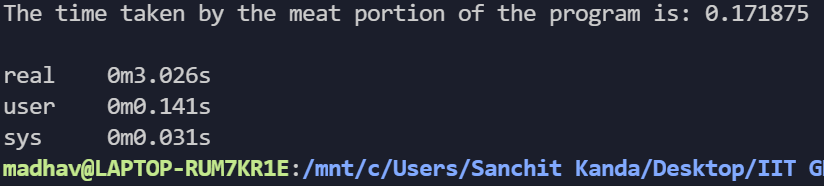


4. N=256

Execution Time: 3.026s

Meat Portion: 0.171875s

(Meat Portion/Execution Time): 0.056799

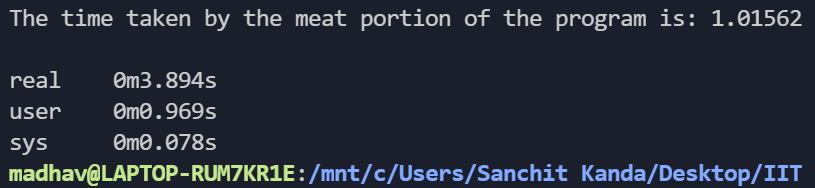


5. N=512

Execution Time: 3.894s

Meat Portion: 1.0156s

(Meat Portion/Execution Time): 0.2608



**a) Python program for Double**

1. N=32

Execution Time: 1.126s

Meat Portion: 0.0110s

(Meat Portion/Execution Time): 0.00976

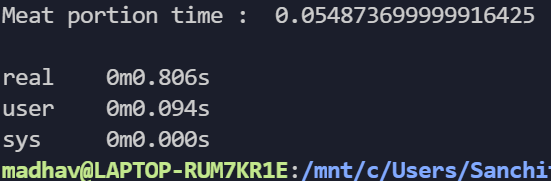


2. N=64

Execution Time: 0.806s

Meat Portion: 0.05487s

(Meat Portion/Execution Time): 0.067

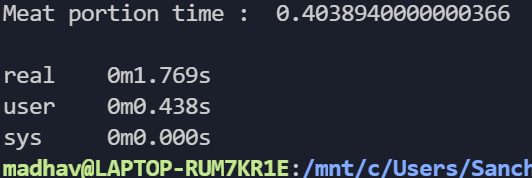


3. N=128

Execution Time: 1.769s

Meat Portion: 0.403s

(Meat Portion/Execution Time): 0.2278

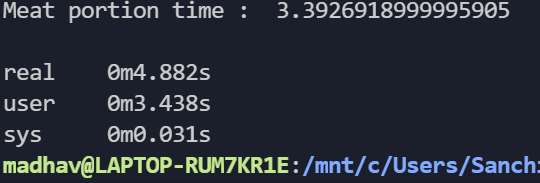


4. N=256

Execution Time: 4.882s

Meat Portion: 3.392s

(Meat Portion/Execution Time): 0.69

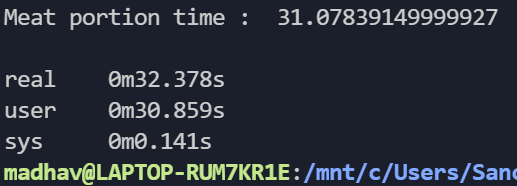


5. N=512

Execution Time: 32.378s

Meat Portion: 31.07s

(Meat Portion/Execution Time): 0.95



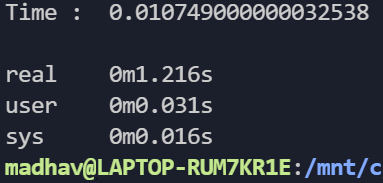
**b)Python program for Integer**

1. N=32

Execution Time: 1.216s

Meat Portion: 0.0107s

(Meat Portion/Execution Time): 0.087

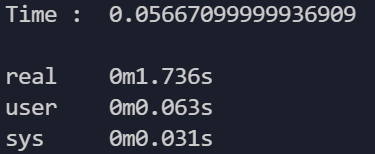


2. N=64

Execution Time: 1.736s

Meat Portion: 0.0566s

(Meat Portion/Execution Time): 0.032

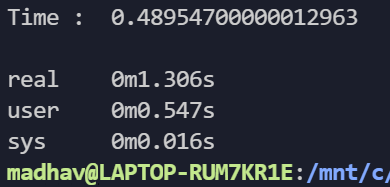


3. N=128

Execution Time: 1.306s

Meat Portion: 0.489s

(Meat Portion/Execution Time): 0.37

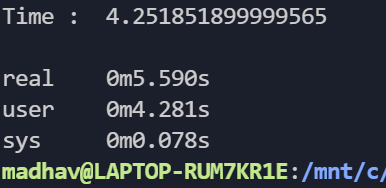


4. N=256

Execution Time: 5.590s

Meat Portion: 4.251s

(Meat Portion/Execution Time): 0.76

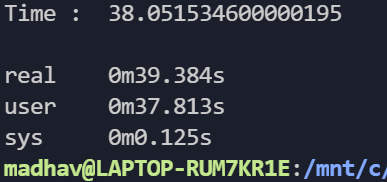


5. N=512

Execution Time: 39.384s

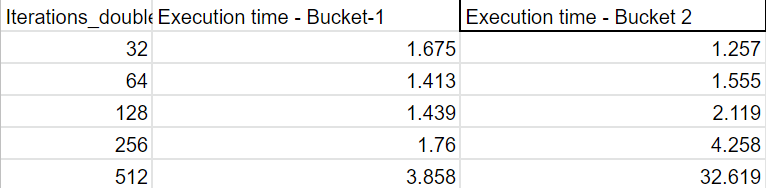
Meat Portion: 38.05s

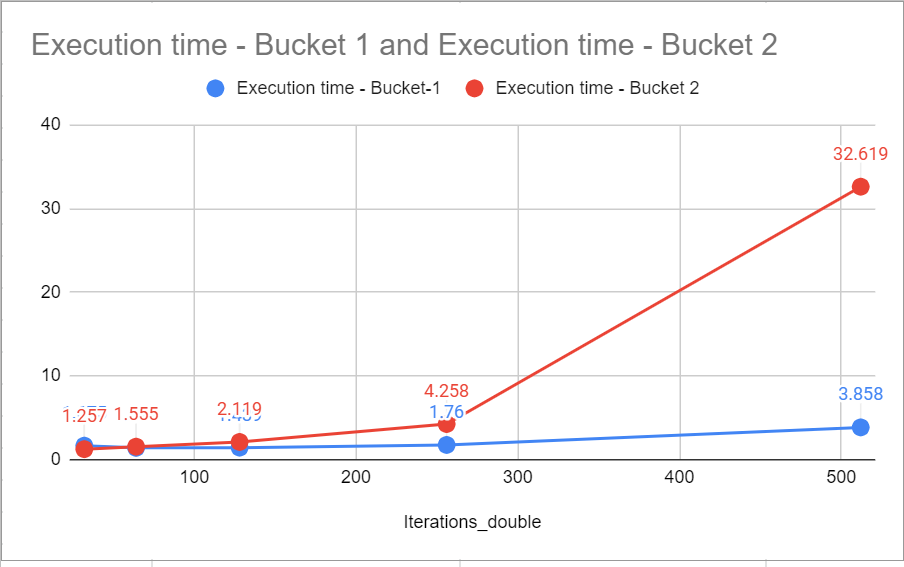
(Meat Portion/Execution Time): 0.96



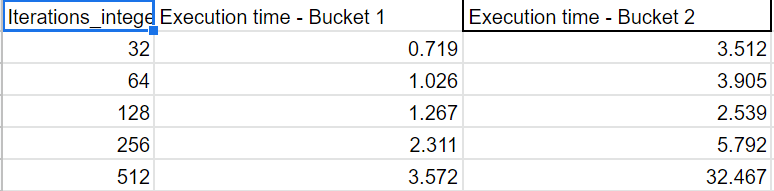
**c)**

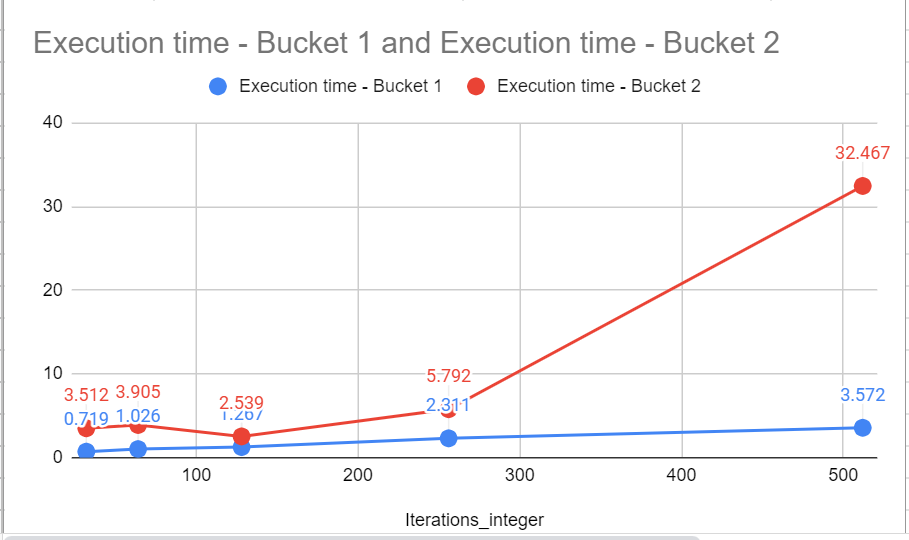
**Execution Time for Double of the Two programming languages**

****

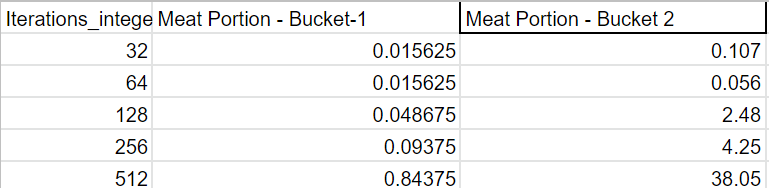
****

**Execution Time for Integer of the two program languages**

****

****

**Meat Portion Time for Integer of the two program languages**

****

