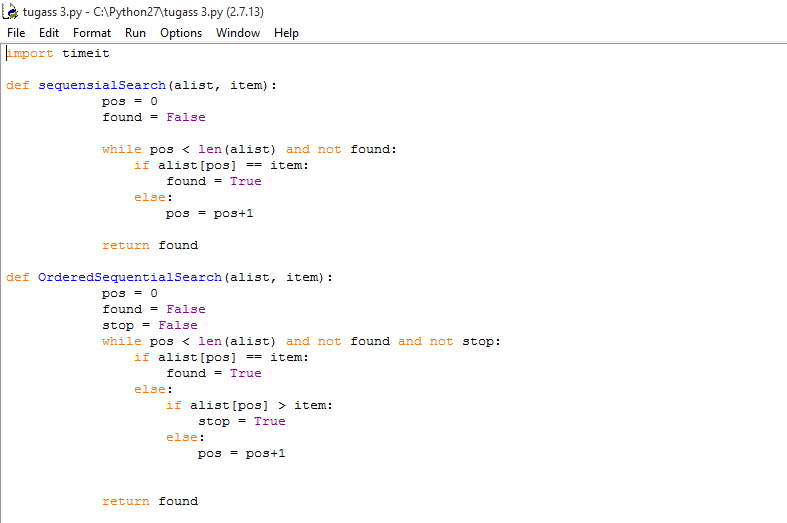
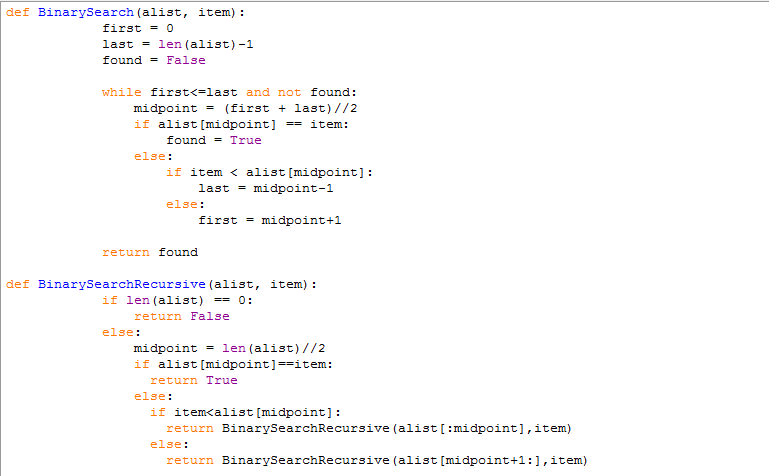
Nama : Bagas Akurna

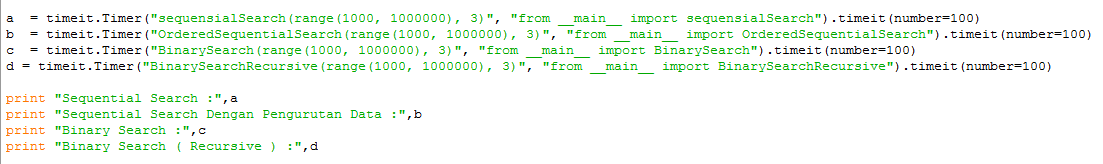
NIM : L200150035

Program Code:



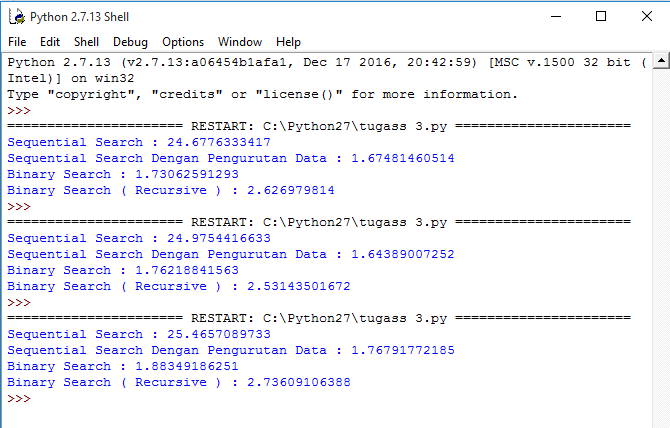


Perintah untuk mengeksekusi program di atas



Di awal program terlebih dahulu kita meng*import* module python yang bernama "timeit" untuk mengetahui berapa lama waktu yang dibutuhkan untuk meng-eksekusi program di masing - masing metode

Hasil dari program:



Kesimpulan : Sequential Search dan Binary Search ( recursive ) merupakan metode yang paling banyak memakan waktu. Sedangkan untuk yang paling singkat adalah metode Binary Search dan Sequential Search dengan pengurutan Data.

Kode program full :

import timeit

def sequensialSearch(alist, item):

pos = 0

found = False

while pos < len(alist) and not found:

if alist[pos] == item:

found = True

else:

pos = pos+1

return found

def OrderedSequentialSearch(alist, item):

pos = 0

found = False

stop = False

while pos < len(alist) and not found and not stop:

if alist[pos] == item:

found = True

else:

if alist[pos] > item:

stop = True

else:

pos = pos+1

return found

def BinarySearch(alist, item):

first = 0

last = len(alist)-1

found = False

while first<=last and not found:

midpoint = (first + last)//2

if alist[midpoint] == item:

found = True

else:

if item < alist[midpoint]:

last = midpoint-1

else:

first = midpoint+1

return found

def BinarySearchRecursive(alist, item):

if len(alist) == 0:

return False

else:

midpoint = len(alist)//2

if alist[midpoint]==item:

return True

else:

if item<alist[midpoint]:

return BinarySearchRecursive(alist[:midpoint],item)

else:

return BinarySearchRecursive(alist[midpoint+1:],item)

a = timeit.Timer("sequensialSearch(range(1000, 1000000), 3)", "from \_\_main\_\_ import sequensialSearch").timeit(number=100)

b = timeit.Timer("OrderedSequentialSearch(range(1000, 1000000), 3)", "from \_\_main\_\_ import OrderedSequentialSearch").timeit(number=100)

c = timeit.Timer("BinarySearch(range(1000, 1000000), 3)", "from \_\_main\_\_ import BinarySearch").timeit(number=100)

d = timeit.Timer("BinarySearchRecursive(range(1000, 1000000), 3)", "from \_\_main\_\_ import BinarySearchRecursive").timeit(number=100)

print "Sequential Search :",a

print "Sequential Search Dengan Pengurutan Data :",b

print "Binary Search :",c

print "Binary Search ( Recursive ) :",d