

TUẦN 4

Họ & Tên: Phạm Nguyễn Phương Duy
MSSV: 19110290

▼ Bài 1

```
[ ] 1 import random
    2
    3 def find_element(A, x):
    4     # Kiểm tra xem x có nằm trong mảng A không
    5     if x in A:
    6         # Nếu x nằm trong mảng A, trả về chỉ số của nó
    7         return A.index(x)
    8     else:
    9         # Nếu x không nằm trong mảng A, trả về None
   10     return None
   11
```

```
[ ] 1 #Test dữ liệu:
    2 for i in range(10, 101, 10):
    3     #Đặt độ dài của mảng A
    4     N = i
    5
    6     # Tạo một mảng ngẫu nhiên đã sắp xếp A có độ dài N
    7     A = sorted([random.randint(0, 100) for _ in range(N)])
    8
    9     # Tạo một phần tử ngẫu nhiên x
   10     x = random.randint(0, 100)
   11
   12     # In mảng A có N phần tử được sắp xếp và phần tử x
   13     print(f"Đặt độ dài của mảng A: {i}")
   14     print(f"A: {A}")
   15     print(f"x: {x}")
   16     print(f"Result: {find_element(A, x)}\n")
```

```
D: Test dữ liệu của mảng A: 10
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
x: 10
Result: 0

Test dữ liệu của mảng A: 20
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200]
x: 10
Result: 0

Test dữ liệu của mảng A: 30
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300]
x: 10
Result: 0

Test dữ liệu của mảng A: 40
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400]
x: 10
Result: 0

Test dữ liệu của mảng A: 50
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500]
x: 10
Result: 0

Test dữ liệu của mảng A: 60
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510, 520, 530, 540, 550, 560, 570, 580, 590, 600]
x: 10
Result: 0

Test dữ liệu của mảng A: 70
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510, 520, 530, 540, 550, 560, 570, 580, 590, 600, 610, 620, 630, 640, 650, 660, 670, 680, 690, 700]
x: 10
Result: 0

Test dữ liệu của mảng A: 80
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510, 520, 530, 540, 550, 560, 570, 580, 590, 600, 610, 620, 630, 640, 650, 660, 670, 680, 690, 700, 710, 720, 730, 740, 750, 760, 770, 780, 790, 800]
x: 10
Result: 0

Test dữ liệu của mảng A: 90
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510, 520, 530, 540, 550, 560, 570, 580, 590, 600, 610, 620, 630, 640, 650, 660, 670, 680, 690, 700, 710, 720, 730, 740, 750, 760, 770, 780, 790, 800, 810, 820, 830, 840, 850, 860, 870, 880, 890, 900]
x: 10
Result: 0

Test dữ liệu của mảng A: 100
A: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510, 520, 530, 540, 550, 560, 570, 580, 590, 600, 610, 620, 630, 640, 650, 660, 670, 680, 690, 700, 710, 720, 730, 740, 750, 760, 770, 780, 790, 800, 810, 820, 830, 840, 850, 860, 870, 880, 890, 900, 910, 920, 930, 940, 950, 960, 970, 980, 990, 1000]
x: 10
Result: 0
```

▼ Bài 2

```
[ ] 1 import random
    2 import time
    3
    4 def find_smallest(S, k):
    5     # Sắp xếp tập hợp S theo thứ tự tăng dần
    6     S.sort()
    7     # Trả về phần tử nhỏ thứ k
    8     return S[k-1]
```

```
▶ 1 #Test Dữ liệu
  2 # Đặt các giá trị của N để kiểm tra
  3 N_values = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
  4 # Đặt giá trị của k
  5 k = 5
  6
  7 # Lặp lại cho mỗi giá trị của N
  8 for N in N_values:
  9     # Khởi tạo tổng thời gian bằng 0
 10     total_time = 0
 11
 12     # Tạo ngẫu nhiên tập hợp S có N phần tử khác nhau trong đó S(i) ∈ {1, 2, . . . , 1000}
 13     S = random.sample(range(1, 1001), N)
 14     # Ghi lại thời gian bắt đầu
 15     start_time = time.time()
 16     # Tìm phần tử nhỏ thứ k trong tập hợp S
 17     result = find_smallest(S, k)
 18     # Ghi lại thời gian kết thúc
 19     end_time = time.time()
 20     # Tính toán thời gian đã qua và cộng vào tổng thời gian
 21     total_time += (end_time - start_time)
 22     # Tính toán thời gian trung bình cho mỗi giá trị của N
 23     avg_time = total_time / 10
 24     # In ra thời gian trung bình cho mỗi giá trị của N
 25     print(f'Với N={N}, Thời gian trung bình: {avg_time:.6f} giây')
 26
```

```
☞ Với N=10, Thời gian trung bình: 0.000000 giây
  Với N=20, Thời gian trung bình: 0.000000 giây
  Với N=30, Thời gian trung bình: 0.000000 giây
  Với N=40, Thời gian trung bình: 0.000000 giây
  Với N=50, Thời gian trung bình: 0.000000 giây
  Với N=60, Thời gian trung bình: 0.000000 giây
  Với N=70, Thời gian trung bình: 0.000001 giây
  Với N=80, Thời gian trung bình: 0.000001 giây
  Với N=90, Thời gian trung bình: 0.000001 giây
  Với N=100, Thời gian trung bình: 0.000001 giây
```