

Nama : Rahmat Beny Susanto

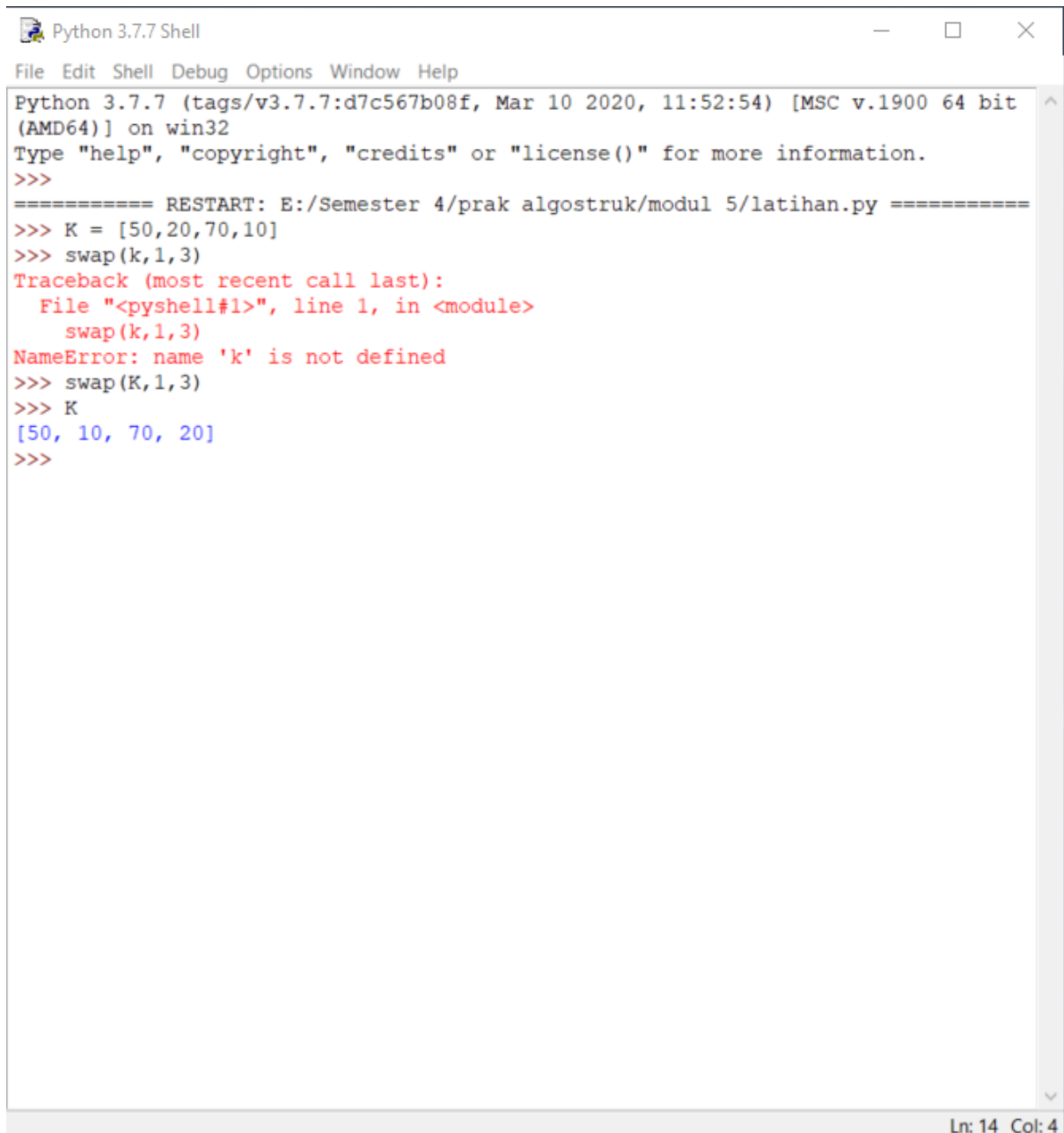
NIM : L200180079

Kelas : C

PRAK ASD

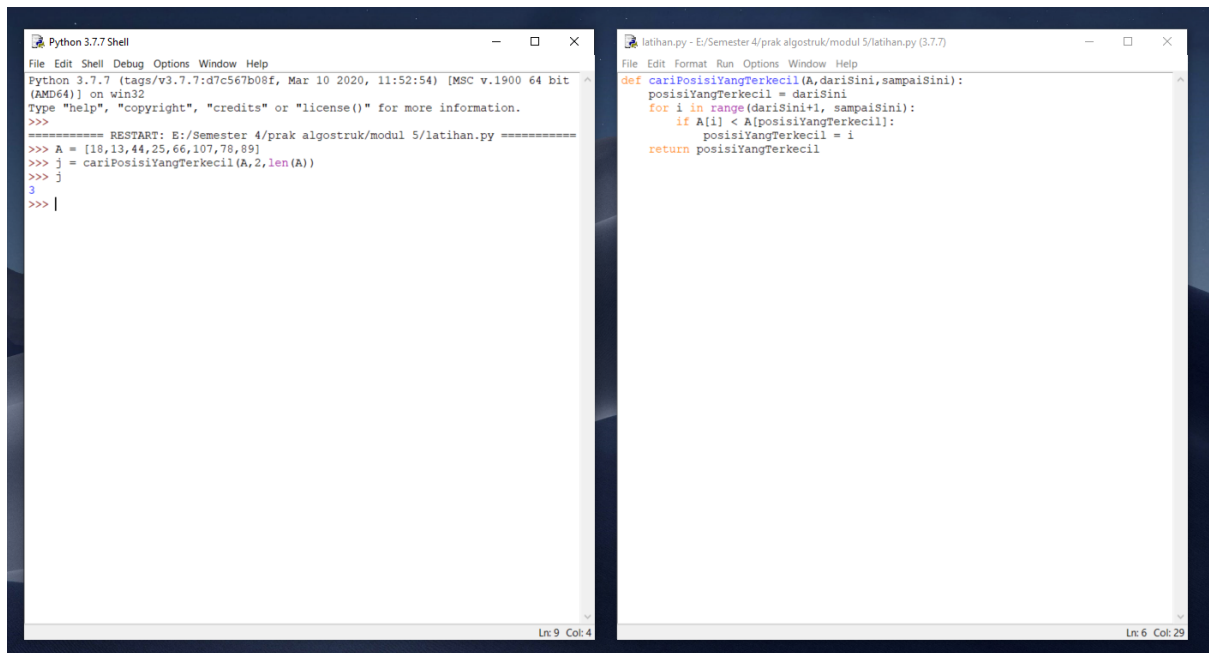
MODUL 5

LATIHAN

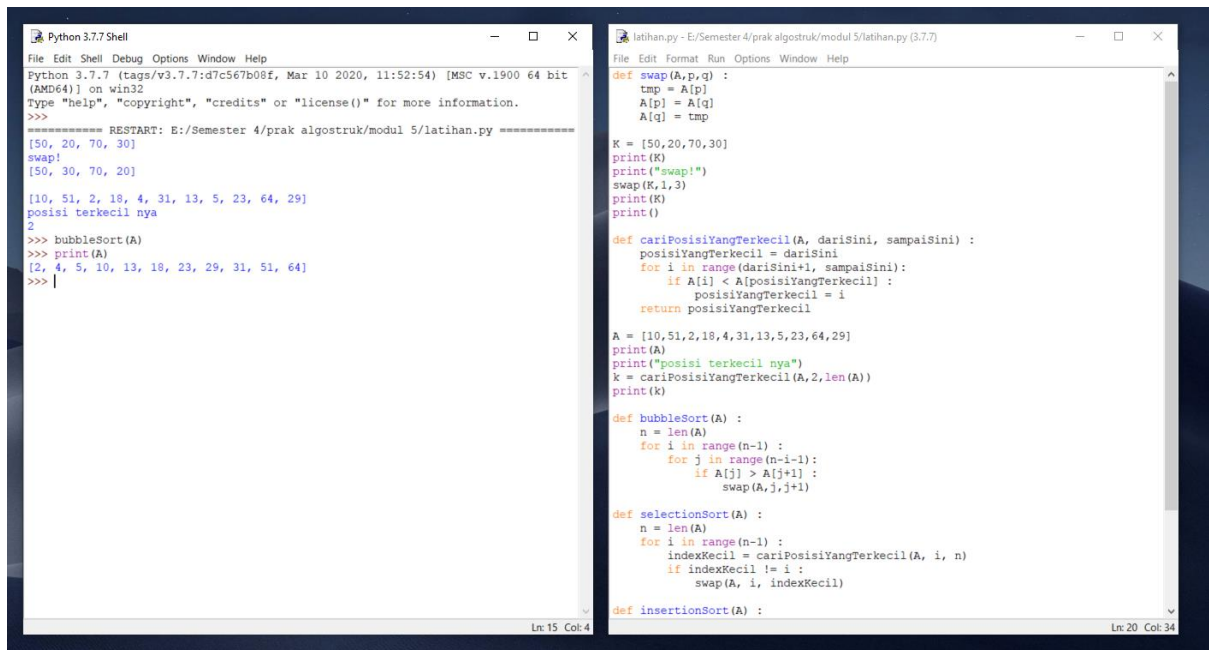
A screenshot of a Python 3.7.7 Shell window. The window has a title bar with the text "Python 3.7.7 Shell" and standard window controls. Below the title bar is a menu bar with "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The main text area shows the following content:

```
Python 3.7.7 (tags/v3.7.7:d7c567b08f, Mar 10 2020, 11:52:54) [MSC v.1900 64 bit  
(AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
===== RESTART: E:/Semester 4/prak algostruk/modul 5/latihan.py =====  
>>> K = [50,20,70,10]  
>>> swap(k,1,3)  
Traceback (most recent call last):  
  File "<pyshell#1>", line 1, in <module>  
    swap(k,1,3)  
NameError: name 'k' is not defined  
>>> swap(K,1,3)  
>>> K  
[50, 10, 70, 20]  
>>>
```

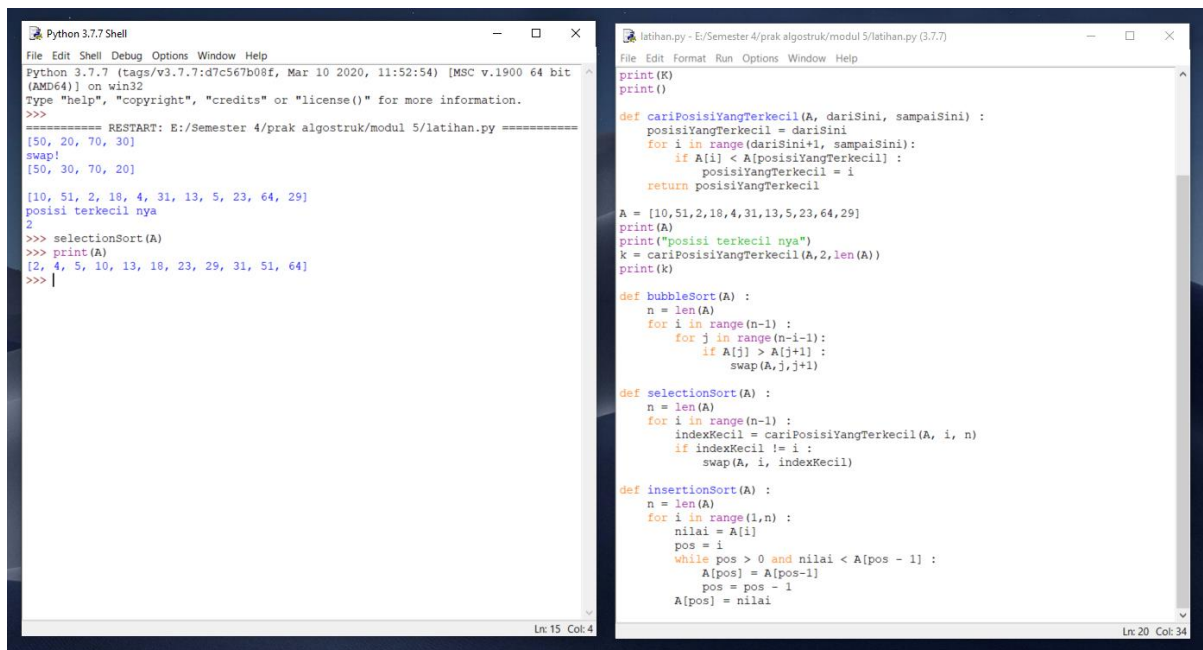
The error message "NameError: name 'k' is not defined" is highlighted in red. The list K is shown as [50, 10, 70, 20] in blue. At the bottom right of the window, the status bar shows "Ln: 14 Col: 4".



Bubble sort



SelectionSort



The image shows two side-by-side Python 3.7.7 Shell windows. The left window displays the execution of a Selection Sort program. It starts with a list [50, 20, 70, 30], swaps the first two elements to get [50, 30, 70, 20], then sorts a larger list [10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29] to get [2, 4, 5, 10, 13, 18, 23, 29, 31, 51, 64]. The right window shows the source code for the Selection Sort implementation, including a helper function `cariPosisiYangTerkecil` and the `selectionSort` function.

```
Python 3.7.7 Shell
File Edit Shell Debug Options Window Help
Python 3.7.7 (tags/v3.7.7:d7c567b08f, Mar 10 2020, 11:52:54) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:/Semester 4/prak algostruk/modul 5/latihan.py =====
[50, 20, 70, 30]
swap!
[50, 30, 70, 20]

[10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29]
posisi terkecil nya
2
>>> selectionSort(A)
>>> print(A)
[2, 4, 5, 10, 13, 18, 23, 29, 31, 51, 64]
>>>

latihan.py - E:/Semester 4/prak algostruk/modul 5/latihan.py (3.7.7)
File Edit Format Run Options Window Help
print(K)
print()

def cariPosisiYangTerkecil(A, dariSini, sampaiSini):
    posisiYangTerkecil = dariSini
    for i in range(dariSini+1, sampaiSini):
        if A[i] < A[posisiYangTerkecil]:
            posisiYangTerkecil = i
    return posisiYangTerkecil

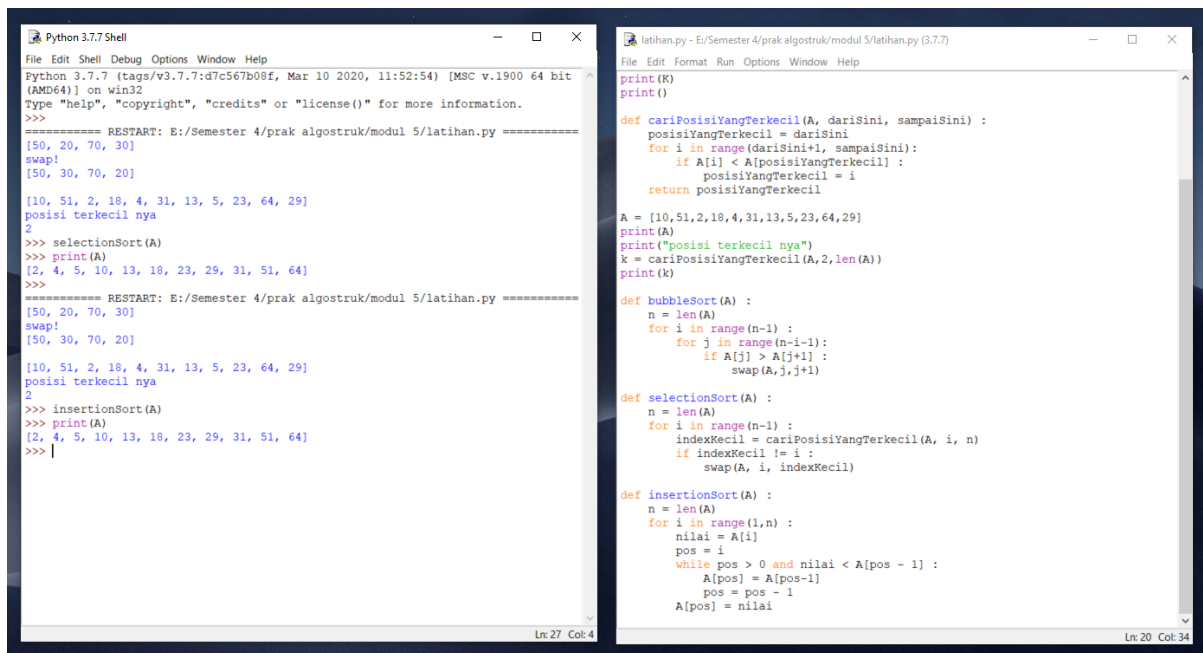
A = [10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29]
print(A)
print("posisi terkecil nya")
k = cariPosisiYangTerkecil(A, 2, len(A))
print(k)

def bubbleSort(A):
    n = len(A)
    for i in range(n-1):
        for j in range(n-i-1):
            if A[j] > A[j+1]:
                swap(A, j, j+1)

def selectionSort(A):
    n = len(A)
    for i in range(n-1):
        indexKecil = cariPosisiYangTerkecil(A, i, n)
        if indexKecil != i:
            swap(A, i, indexKecil)

def insertionSort(A):
    n = len(A)
    for i in range(1, n):
        nilai = A[i]
        pos = i
        while pos > 0 and nilai < A[pos - 1]:
            A[pos] = A[pos-1]
            pos = pos - 1
        A[pos] = nilai
```

insertionSort



The image shows two side-by-side Python 3.7.7 Shell windows. The left window displays the execution of an Insertion Sort program. It starts with a list [50, 20, 70, 30], swaps the first two elements to get [50, 30, 70, 20], then sorts a larger list [10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29] to get [2, 4, 5, 10, 13, 18, 23, 29, 31, 51, 64]. The right window shows the source code for the Insertion Sort implementation, including a helper function `cariPosisiYangTerkecil` and the `insertionSort` function.

```
Python 3.7.7 Shell
File Edit Shell Debug Options Window Help
Python 3.7.7 (tags/v3.7.7:d7c567b08f, Mar 10 2020, 11:52:54) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:/Semester 4/prak algostruk/modul 5/latihan.py =====
[50, 20, 70, 30]
swap!
[50, 30, 70, 20]

[10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29]
posisi terkecil nya
2
>>> insertionSort(A)
>>> print(A)
[2, 4, 5, 10, 13, 18, 23, 29, 31, 51, 64]
>>>

latihan.py - E:/Semester 4/prak algostruk/modul 5/latihan.py (3.7.7)
File Edit Format Run Options Window Help
print(K)
print()

def cariPosisiYangTerkecil(A, dariSini, sampaiSini):
    posisiYangTerkecil = dariSini
    for i in range(dariSini+1, sampaiSini):
        if A[i] < A[posisiYangTerkecil]:
            posisiYangTerkecil = i
    return posisiYangTerkecil

A = [10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29]
print(A)
print("posisi terkecil nya")
k = cariPosisiYangTerkecil(A, 2, len(A))
print(k)

def bubbleSort(A):
    n = len(A)
    for i in range(n-1):
        for j in range(n-i-1):
            if A[j] > A[j+1]:
                swap(A, j, j+1)

def selectionSort(A):
    n = len(A)
    for i in range(n-1):
        indexKecil = cariPosisiYangTerkecil(A, i, n)
        if indexKecil != i:
            swap(A, i, indexKecil)

def insertionSort(A):
    n = len(A)
    for i in range(1, n):
        nilai = A[i]
        pos = i
        while pos > 0 and nilai < A[pos - 1]:
            A[pos] = A[pos-1]
            pos = pos - 1
        A[pos] = nilai
```