

Exercise 1

30 min



1,3 27 459

PP

procédure saisir (@ X : entier)
P. formel

fonction generer-suite (X : entier) : chaine

fonction inverse (X) : entier

procédure saisir (@ $N : \text{entier}$)

P. formel / passage par
adresse
= variable

Debut-

Repete

Ecrire ("donner N")

lire (N)

$N \bmod 11 \neq 0$

jusqu'à $N \in [10..99]$ et $\underbrace{N \bmod 10}_{4} \neq \underbrace{N \div 10}_{5}$

Fin

fonction inverse ($X : \text{entier}$) : entier

P. formel / passage par valeur

Debut

retourner $(X \bmod 10 \times 10) + X \div 10$

Fin

res	18	63	27	45	9
-----	----	----	----	----	---

fonction generer. Date (\overline{X} : entier) : chaine

Debut

res ← CONVCH (X)

Repetier

X ← abs (X - inverse (X))

45 - 54
9

P. effectif

Appel fct

res ← res + "L" + CONVCH (X)

jusqu'à X = 9

retourner res

Fin

TDOL

①	T/N
res	chaine
inverse	fonction

PD

algorithme. Suite Neuf

Debut

P. effectif

Saisir (N)

Ecrire (generer_suite (N))

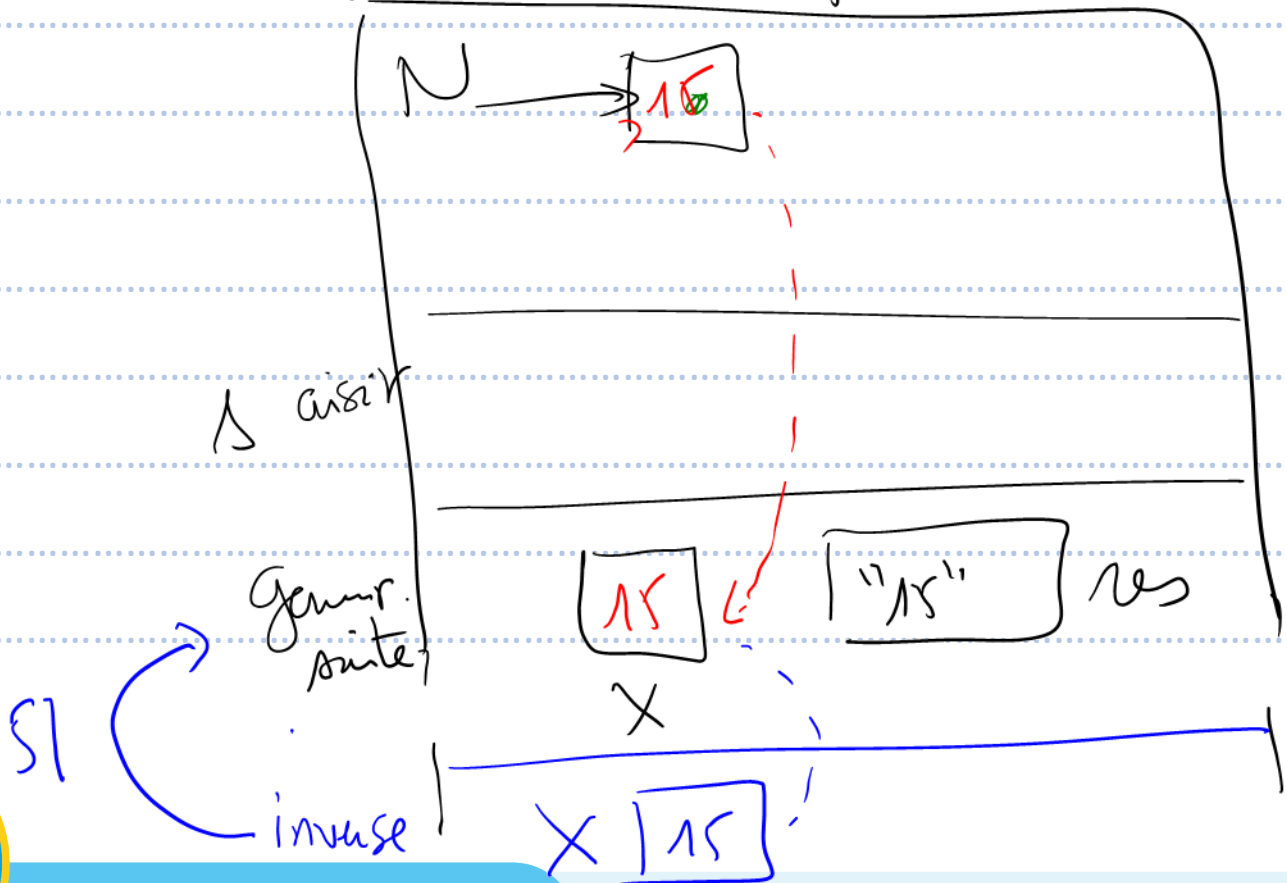
fin

Appel de la fct

TDBG

O	TW
N	entier
Saisir	procedur
gener-suite	fonction

MLC



Exercice 2

⌚ 20 min



tableau de 1 dimension = vecteur

procedure affiche($R: tab; X: val$)

PP

procedure saisir($@ n: entier$)

procedure remplir($n: entier; @ T: tab$)

procedure Inverse($T: tab; n: entier; @ V: tab$)

procedure saisir(@X:entier) (1 Resultat)

Debut

```

| Repeter
|   | Ecrire ("donner N")
|   | lire(X)
|   | jusqu'à X ≥ 2

```

Fin

Passage par valeur

Passage par adresse

procedure remplir(X:entier; @K:tab) (1 resultat)

Debut

```

| Pour i de 1 à X faire
|   | Ecrire ("T[", i, "] = ")
|   | lire (K[i])

```

Fin Pour

Fin

T D O L	
0	T N
i	entier

procedure affiche (R: tab; X: entier)

Resultat

P formal. Passage Par valeurs

Debut

Ecrire ("V = ")

Pour i de 1 à X faire

Ecrire (R[i])

FinPour

Fin

TOOL

O	T/n
i	entier

Procedure inverse (T: tab; N: entier; @V: tab)

Debut

Pour i de 1 à N faire

$V[N-i+1] \leftarrow T[i]$

FinPour

Fin

TOOL

O	T/n
i	entier

	1	2	3	4	5
T	-6	0	10	3	7

	1	2	3	4	5
V					-6

5
N

PP

algorithm. Inverse-TV
Debut

(

 Saisir (N)

 Remplir (N, T)

 inverse (T, N, V)

 affiche (V, N)

)

 Fin

TDOG

O	T N
Saisir, remplir	procedun
inverse, affiche	procedun
N	entier
T, V	Tab

T D N T

Tab = tableau de
~~100~~ entiers

B-

```

from numpy import *
def saisie():
    N=int(input("Saisir taille tableau "))
    while not(N>=2):
        N=int(input("Saisir taille tableau "))
    return N
"""
def saisie():
    valide=False
    while not(valide):
        N=int(input("Saisir taille tableau "))
        valide=N>=2
    return N
"""

def remplir(N):
    #global T erreur
    for i in range(N):
        T[i]=int(input("T["+str(i)+"]="))
    #return T erreur

def inverse(N):
    for i in range(N // 2):
        aux=T[i]
        T[i]=T[N-i-1]
        T[N-i-1]=aux

def affiche(N):
    for i in range(N):
        print(T[i])
#PP
N=saisie()
T=array([int]*N)
remplir(N)
inverse(N)
affiche(N)

```