Correction des exercices

**I les nombres**

Exercice 1

p==q donne « vrai » car 10 et 10.0 ont la même valeur

p is q est faux car ils ne sont pas exactement la même chose, ils n’ont pas le même type, p est un entier et q est un réel.

Exercice 2

A=sqrt(3)/4 b=135./124. C=(5./3.)\*\*3

**II les textes**

Questions

1. BonjourBonjour
2. bonjour bonjour les gens
3. on ne peut multiplier les chaines de caractères que par des entiers
4. len(chaine) donne le nombre de caractères d’une chaine de caractère autrement dit sa longueur.
5. texte.count(chain) donne le nombre de fois où la chaine de caractère stockée dans la variable chain est présente dans la chaine s’appelée texte.
6. texte.count("gens") nous donne comme résultat 4
7. texte.count(texte[43]) donne comme réponse 30 (texte[43] correspond à la lettre ‘r’
8. chaine[i:j] donne la partie de la chaine allant du caractère de rang i (compris) jusqu’au caractère de rang j non compris.

chaine[i:] ] donne la partie de la chaine partant du caractère de rang i (compris) (jusqu’à la fin)

chaine[:j] donne la partie de la chaine allant jusqu’au caractère de rang j non compris.

**III les listes**

Exercice

1. maliste[0] donne le premier élément et maliste[-2] donnera l’avant dernière.
2. La fonction count (dans maliste.count(elt) ne compte plus le nombre d’occurrence de l’élément dans la liste, mais le nombre de fois ou elt est exactement un élément de la liste)
3. La fonction append sert à rajouter un élément à une liste. Elle rajoute à la fin et pas au milieu donc elle ne peut insérer « coucou » entre le deuxième et le troisième élément.
4. On écrira liste.insert(2,"coucou") et liste.insert(0,"coucou")
5. liste.remove('coucou') ne fonctionne pas car il retire le premier coucou et non le second, on préferera dans ce cas récupérer l’indice du second « coucou » (il valait 3 avant la tentative ratée) et donc on tape del liste[3]
6. on a copié le contenu de c dans d puis on a modifié d puis on a affiché les valeurs des deux variables. Puis on essaye de faire quelque chose d’analogue avec deux listes et là on se rend compte que toute modification d’une liste est repercuté dans toutes ses copies.

On voit alors une autre manière de faire la copie de la liste : d=c[ :] on copie en quelque sorte le contenu de la liste, avant ce qu’on faisait revenait à donner plusieurs nom au même objet.

1. list(range(7)) donnera [0,1,2,3,4,5,6]
2. range(i,j) va faire une liste allant de I à j non compris. range(i,j,k) va faire une liste commençant à i et ne dépassant jamais j et allant de k en k.
3. [-5,-4.8,-4.6,-4.4, … , 17.6]=arange(-5,17.8,0.2)
4. arange(13,956)
5. arange(147,3201.1,0.1)

**IV Les programmes**

**Exercice 1**

nom = input(“ quel est ton nom ?  “)

prenom = input(“ quel est ton prénom ?  “)

parfum = input(“ quel est ton parfum de glace préféré ?  “)

phrase = prenom+’ ‘+nom+’ ‘ +’ton parfum de glace préféré est ‘+parfum

**Exercice 2**

On obtient respectivement 1 et 3

Il a compté à chaque fois le nombre de x

Il faut remplacer la ligne 10 par n=texte.count(x)

**Exercice 3**

Int(247/22) nous donne le quotient et 247%22 le reste (de la division euclidienne de 247 par 22)

diviseur = int(input(“ quel est le diviseur ? “))

dividende = int(input(“ quel est le dividende ? “))

quotient = int(dividende / diviseur)

reste = dividende% diviseur

print(‘le quotient est ‘+quotient+’ et le reste est : ‘+reste)

**Exercice 4**

vitesse = int(input(“ quel est la vitesse du véhicule ? “))

distance = int(input(“ à quelle distance se situe l’obstacle  “))

if (0.3\*vitesse+0,0075\*v\*v>distance) :

 print(‘désolé mais ton véhicule a emboutit l’obstacle’)

else :

 print(‘tu as réussi à éviter l’obstacle mais reste vigilant!’)

**Exercice 5**

from random import randint

val=randint(1,6)

if val==1 or val ==6 :

print(‘si ton cheval est dans l’écurie tu as le droit de le sortir’)

else :

print(‘si ton cheval est dans l’écurie, désolé mon coco mais il va y rester un tour de plus ! ‘)

**Exercice 6**

from random import randint

carte=randint(1,32)

if carte==1 or carte==9 or carte ==17 or carte == 25 : # condition synthétique : carte%8==1

 print (‘gagné’)

else :

 print (‘perdu’)

version difficile

on peut rajouter

if carte%8==1 :

 phrase=“c’est un as de “

if carte%8==6 :

 phrase=“c’est un valet de “

if carte%8==7 :

 phrase=“c’est une reine de “

if carte%8==0 :

 phrase=“c’est un roi de “

if carte%8>1 and  carte%8<6:

 val = carte+5

 phrase=“c’est un “+val+“ de “

if carte<9 :

 phrase+=“ piques“

if carte>8 and carte<17 :

 phrase+=“coeur“

if carte>16 and carte<25 :

 phrase+=“trèfle“

if carte>24 :

 phrase+=“carreau“

print(phrase)

**Exercice 7**

from random import randint

for i in rang(10) :

 val=randint(1,6)

 print (“le lancer vaut ” +val)

**Exercice 8**

from random import randint

for i in rang(10) :

 val=randint(1,6)

if val == 6 :

 print(“gagné“)

else :

 print(“perdu“)

**Exercice 9**

Ce programme simule 4 tirages d’un dé classique à six faces et range le résultat obtenu dans une liste appelée L, qui est affiché à la dernière étape

**Exercice 10**

from random import randint

score =0

for i in rang(5000) :

 val=randint(1,6)

if val == 6 :

 score+=1 (autrement dit score = score + 1)

print(“sur 5000 lancers on a obtenu “ +val+ “ six“ )

**Exercice 11**

from random import randint

resultats =[]

for i in rang(5) :

 resultat.append(randint(1,49))

print(resultat)

**Exercice 12**

from random import randint

for i in rang(1,11) :

 val=randint(1,6)

 print (“le lancer n°”+i+” vaut ” +val)

**Exercice 13**

for i in rang(500,566,3) :

 print (i)

**Exercice 14**

from math import sqrt

min = int(input(“ valeur minimale de x“))

max = int(input(“ valeur maximale de x “))

pas = int(input(“ quel est le pas “))

for i in rang(min,max+1,pas) :

 print (“f(“+i+“)=“+(sqrt(i)+4)\*\*2)

Exercice 15

Liste=[“Alice“, “Nawel“, “Lou“, “Alexandra“, “Candice“,“ Shaïna“]

For i in Liste :

 Print(““+i+“ a “+len(i)+ “ lettres dans son prénom“)

**Palindromes**

1) la première lettre a pour rang 0 et la dernière lettre a pour rang -1 ou encore « nombre total de lettre – 1 » (penser au fait que les indices/rang sont décalés par rapport aux numéros quand on compte à la main.

La seconde lettre sera : 1 et l’avant dernière sera -2 ou « nombre de lettres total -2 »

Donc la i ème lettre aura pour lettre associée –i-1 ou nombre de lettres total moins –i -1

2) Traduire l’algorithme sous la forme d’un programme en Python

mot = input(“ quel est votre mot ?“))

longueur = len(mot)

palin=True

for i in range(longueur) :

 if mot[i]!=mot[-i-1] :

 palin = False

print(palin)

**Exercice 16**

from math import sqrt

min = int(input(“ valeur minimale de x“))

max = int(input(“ valeur maximale de x “))

pas = int(input(“ quel est le pas “))

i = min

while i<=max :

 i+=pas

 print (“f(“+i+“)=“+(sqrt(i)+4)\*\*2)

**Exercice 17**

from random import randint

inconnue = randint(1,100)

prop = 0 #fausse première proposition

compteur = 0

while prop != inconnue :

 compteur+=1

prop = int(input(“ Votre proposition ? “))

if prop>inconnue :

 print(“trop grand“)

else :

 print(“trop petit“)

print(“ bien joué ! vous avez réussi en “+ compteur + “ coups “)

**Exercice 18**

from random import randint

potentilel=list(range(1,50))

resultats =[]

for i in rang(5) :

 rangGagnant = randint(0,len(resultat)-1)

 resultat.append(potentiel[rangGagnant])

 del potentiel[rangGagnant]

print(resultat)

**Fonctions**

**Exercice 1**

from random import randint

def lotoGen(x) :

potentilel=list(range(1,50))

resultats =[]

for i in rang(5) :

 rangGagnant = randint(0,len(resultat)-1)

 resultat.append(potentiel[rangGagnant])

 del potentiel[rangGagnant]

return resultat

**Exercice 2**

def palindrome(mot) :

longueur = len(mot)

palin=True

for i in range(longueur) :

 if mot[i]!=mot[-i-1] :

 palin = False

return(palin)

**Exercice 3**

def summ(min, max) :

 nombre = min

s=nombre

 while nombre <max :

 nombre=1+nombre

 s=s+nombre

 return s

**Exercice 4**

def facto(nombre) :

 facto = 1

for i in range (2,nombre+1)

 facto = facto\*i

return facto