

tp3-algo

TP 3

Exercice 1 :

Programmer la calculatrice, vue en TD.
Fiabiliser ce programme en interdisant la saisie de valeurs incorrectes.

Exercice 2 :

Programmer le calcul du PGCD basé sur l'algorithme d'Euclide, vu en TD.
Fiabiliser ce programme.

Exercice 3 :

Ecrire un programme permettant de saisir une date (3 entiers, représentant le jour, le mois et l'année) et de vérifier si cette date est correcte. En effet, le 32 janvier 2000 n'existe pas, ni le 30 février 2001.

Informations utiles :

- Les mois de janvier, mars, mai, juillet, août, octobre et décembre ont 31 jours.
- Les mois d'avril, juin, septembre et novembre ont 30 jours
- Le mois de février a 28 jours si l'année n'est pas bissextile, sinon il a 29 jours.

Depuis l'ajustement du calendrier grégorien, l'année n'est bissextile (comportant 366 jours) que dans l'un des deux cas suivants :

- *Si l'année est divisible par 4 et non divisible par 100 ;*
- *Si l'année est divisible par 400.*

Sinon, l'année n'est pas bissextile (elle a 365 jours).

(...)

L'an 2008 était bissextile suivant la première règle (divisible par 4). L'an 1900 n'était pas bissextile, car divisible par 100 mais non divisible par 400. L'an 2000 était bissextile car divisible par 400.

exo1

Ce programme doit pouvoir faire les opérations de base telles la soustraction, l'addition, la multiplication et la division.

Il possède un menu vérifiant si l'utilisateur utilise bien les valeurs proposées et qui lui redemande s'il rentre un entier non dispo jusqu'à que son entrée soit bonne.

pour la division il regarde si l'utilisateur divise par 0 et lui demande s'il souhaite remplacer sa valeur par une autre pour effectuer la division.

programme

```
if __name__ == "__main__":
    a: int
    b: int
    resultat: int
    choix: int
    operateur: int

    print("bienvenue\nVeuillez choisir un opérateur:")
    print("1-addition\n2-soustraction\n3-multiplication\n4-division")

    operateur= int(input("votre choix: "))
    while (operateur<1 or operateur>4):
        print("bienvenue\nVeuillez choisir un opérateur:")
        print("1-addition\n2-soustraction\n3-multiplication\n4-division")

        operateur= int(input("votre choix: "))
    a = int(input("saisir a: "))
    b = int(input("saisir b: "))

    if (operateur==1):
        resultat=a+b
        print(a,"+",b,"=",resultat)
    elif (operateur==2):
        resultat=a-b
        print(a,"-",b,"=",resultat)
    elif (operateur==3):
        resultat=a*b
        print(a,"*",b,"=",resultat)
```

```

else:
    if (b==0):
        print("la division par 0 est impossible souhaitez vous changer de valeur?")
        print("1-oui\n2-non")
        choix= int(input("votre choix: "))
        while (choix<1 or choix>2):
            print("la division par 0 est impossible souhaitez vous changer de valeur?")
            print("1-oui\n2-non")
            choix= int(input("votre choix: "))
        if (choix==1):
            b = int(input("saisir b: "))
            while(b==0):
                b = int(input("saisir b: "))
            resultat =a//b
            print(a,"/",b,"=",resultat)
        else:
            print("au revoir")
    else:
        resultat =a//b
        print(a,"/",b,"=",resultat)

```

jeux d'essais

```

bienvenue
Veuillez choisir un opérateur:
1-addition
2-soustraction
3-multiplication
4-division
votre choix: 2
saisir a: 6
saisir b: 4
6 - 4 = 2

```

soustraction faite

```

bienvenue
Veuillez choisir un opérateur:
1-addition
2-soustraction
3-multiplication
4-division
votre choix: 0
bienvenue
Veuillez choisir un opérateur:
1-addition
2-soustraction
3-multiplication
4-division
votre choix: 5
bienvenue
Veuillez choisir un opérateur:
1-addition
2-soustraction
3-multiplication
4-division
votre choix: 1
saisir a: 6
saisir b: 8
6 + 8 = 14

```

menu fonctionnant parfaitement et addition faite

```
bienvenue
Veuillez choisir un opérateur:
1-addition
2-soustraction
3-multiplication
4-division
votre choix: 3
saisir a: 9
saisir b: -2
9 * -2 = -18
```

multiplication faite

```
bienvenue
Veuillez choisir un opérateur:
1-addition
2-soustraction
3-multiplication
4-division
votre choix: 4
saisir a: 4
saisir b: 0
la division par 0 est impossible souhaitez vous changer de valeur?
1-oui
2-non
votre choix: 3
la division par 0 est impossible souhaitez vous changer de valeur?
1-oui
2-non
votre choix: 1
saisir b: 0
saisir b: 2
4 / 2 = 2
```

si le diviseur est 0 le choix de changer de diviseur est proposé et si c'est accepté un nouveau diviseur remplacera le 0 (dû la la boucle while rendant cela sûr) et l'opération est effectuée

```
bienvenue
Veuillez choisir un opérateur:
1-addition
2-soustraction
3-multiplication
4-division
votre choix: 4
saisir a: 3
saisir b: 0
la division par 0 est impossible souhaitez vous changer de valeur?
1-oui
2-non
votre choix: 2
au revoir
```

si la réponse est non l'utilisateur quitte le programme

```
bienvvenue  
Veuillez choisir un opérateur:  
1-addition  
2-soustraction  
3-multiplication  
4-division  
votre choix: 4  
saisir a: 4  
saisir b: 2  
4 / 2 = 2
```

si l'opération est juste elle est validée

Exo2:

le programme regarde si les deux entiers sont supérieurs et différents de 0
sinon redemande l'entrée de ces derniers.
selon la valeur des entiers entrés le programme calculera leur pgcd
correspondant.

Programme:

```
if __name__ == "__main__":
    a: int
    b: int
    c: int

    print("Trouveur de PGCD:")
    a = int(input("saisir a: "))
    while (a<=0):
        a = int(input("saisir a superieur et different de 0: "))
    b = int(input("saisir b: "))
    while (b<=0):
        b = int(input("saisir b superieur et different de 0: "))

    while (a!=b):
        if a>b:
            a=a-b
        else:
            b=b-a
    print("le PGCD est ", a)
```

jeux d'essais

```
Trouveur de PGCD:
saisir a: -2
saisir a superieur et different de 0: 0
saisir a superieur et different de 0: 5
saisir b: -2
saisir b superieur et different de 0: 0
saisir b superieur et different de 0: 6
le PGCD est  1
PS C:\Users\Moi\Desktop\autres\cours\2024
2025/s1/r1.01/tp3/ex2.py
Trouveur de PGCD:
saisir a: 69
saisir b: 7
le PGCD est  1
```

des valeur égales et inférieures à 0 ne sont pas saisissables par ce fait le pgcd se fait facilement par la suite

exo3

Le vérificateur de date prendra en compte l'entrée d'un jour, d'un mois et d'une année qui lui permettra de vérifier si la date est juste.

selon le mois le nombre de jours sera vérifié, s'il est supérieur au nombre de jours du mois ou inférieur à 1 la date est incorrecte. Similairement la vérification du mois se fait de la sorte ou l'on doit se situer entre 1 et 12 compris. Seule exception pour le mois de février ou l'on vérifie si l'année est bissextile ou non avant de vérifier le jour.

Programme:

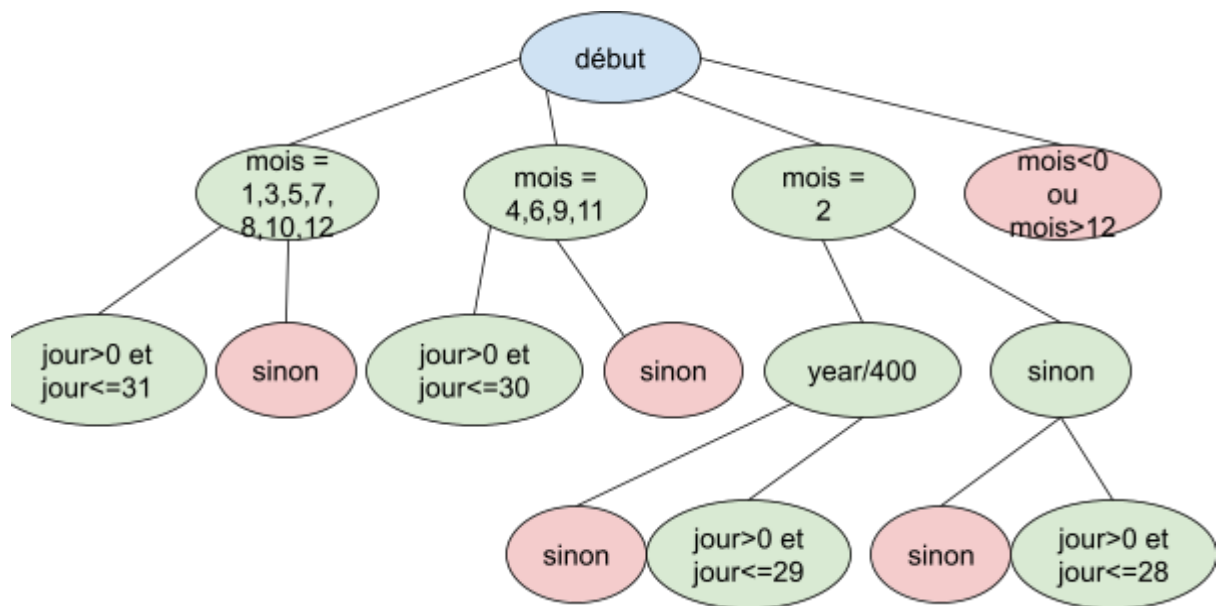
```

if __name__ == "__main__":
    day: int
    month: int
    year: int
    day=(int(input("saisir le jour: ")))
    month=(int(input("saisir le mois: ")))
    year=(int(input("saisir l'année: ")))

    if month in [1,3,5,7,8,10,12]:
        if(day<=0 or day>31):
            print("la date est incorrecte")
        else:
            print("la date est correcte")
    elif month in [4,6,9,11]:
        if (day<=0 or day>30):
            print("la date est incorrecte")
        else:
            print("la date est correcte")
    elif (month==2):
        if(year%400==0):
            if(day<=0 or day>29):
                print("la date est incorrecte")
            else:
                print("la date est correcte")
        else:
            if(day<=0 or day>28):
                print("la date est incorrecte")
            else:
                print("la date est correcte")
    else:
        print("la date est incorrecte")

```

jeux d'essais:



```

saisir le jour: -1
saisir le mois: 1
saisir l'année': 555
la date est incorrecte
PS C:\Users\Moi\Desktop> 2025/s1/r1.01/tp3/ex3.py
saisir le jour: 63
saisir le mois: 1
saisir l'année': 555
la date est incorrecte
PS C:\Users\Moi\Desktop> 2025/s1/r1.01/tp3/ex3.py
saisir le jour: 5
saisir le mois: 1
saisir l'année': 555
la date est correcte
  
```

```

saisir le jour: 31
saisir le mois: 4
saisir l'année': 555
la date est incorrecte
PS C:\Users\Moi\Desktop> 1/tp3/ex3.py
saisir le jour: 0
saisir le mois: 4
saisir l'année': 444
la date est incorrecte
PS C:\Users\Moi\Desktop> 1/tp3/ex3.py
saisir le jour: 6
saisir le mois: 4
saisir l'année': 444
la date est correcte
  
```

```
saisir le jour: 29
saisir le mois: 2
saisir l'année': 2024
la date est incorrecte
PS C:\Users\Moi\Desktop\
1/tp3/ex3.py
saisir le jour: 0
saisir le mois: 2
saisir l'année': 2024
la date est incorrecte
PS C:\Users\Moi\Desktop\
PS C:\Users\Moi\Desktop\
1/tp3/ex3.py
saisir le jour: 13
saisir le mois: 2
saisir l'année': 2024
la date est correcte
```

```
saisir le jour: 29
saisir le mois: 2
saisir l'année': 2000
la date est correcte
PS C:\Users\Moi\Desktop\
1/tp3/ex3.py
saisir le jour: 30
saisir le mois: 2
saisir l'année': 2000
la date est incorrecte
PS C:\Users\Moi\Desktop\
1/tp3/ex3.py
saisir le jour: 0
saisir le mois: 2
saisir l'année': 2000
la date est incorrecte
```

```
saisir le jour: 2
saisir le mois: 0
saisir l'année': 2000
la date est incorrecte
PS C:\Users\Moi\Desktop\
2025/s1/r1.01/tp3/ex3.p
saisir le jour: 2
saisir le mois: 18
saisir l'année': 2000
la date est incorrecte
```