

Bonjour,

Jean MEEUS, dans un livre édité par la SAF paru en 1986, sous le titre « *Calculs astronomiques à l'usage des amateurs* » propose un algorithme qui permet de calculer le jour julien pour une date contenue dans l'un des deux calendriers : julien et grégorien.

(Cet ouvrage a été réédité récemment. Actuellement épuisé, il est en cours de réédition. Voir <http://boutique-saf-astronomie.com/acatalog/Livre-neuf-SAF-Jean-MEEUS-Calculs-astronomiques-lusage-des-amateurs-Calastro001.html>)

L'algorithme est également présent dans « *Astronomical algorithms* », du même auteur, publié par Willmann-Bell, Inc. Voir : <https://www.willbell.com/math/mc1.htm>)

1. Algorithme pour les deux calendriers

(remarque : la fonction int() est l'équivalent de la fonction TRONQ() présentée précédemment : elle retourne la valeur entière d'un nombre décimal en respectant son signe).

Soit MM = mois, YYYY = année, jour = numéro du jour.

Si MM > 2

alors an = YYYY et mois = MM

sinon an = YYYY - 1 et mois = MM + 12

Calcul du correctif entre les deux calendriers :

determinant = YYYY + MM / 100 + jour / 10000

Si determinant >= 1582.1015:

alors A = int(an / 100) et B = 2 - A + int(A / 4)

sinon B = 0

Calcul du jour julien

jj = int(365.25 * an) + int(30.6001*(mois + 1)) + jour + 1720994.5 + B

2. Exemple de programme informatique en langage Python utilisant cet algorithme

Le calcul est effectué par une fonction nommée **jdMeeus()**

Les paramètres jour et heure sont passés sous la forme de chaînes de caractères (entre guillemets) :

Par exemple :

jour = "1957-10-04"

heure = "12:00:00.0"

La fonction float() convertit en nombre flottant.

La fonction sexaToDecimal() convertit l'heure en nombre décimal.

Code source de la fonction

```
def jdMeeus(date = "1957-10-04", time = "00:00:00.0") :
    """
    Calcul du jour julien selon la formule de Jean Meeus
    page 18 de "Calculs astronomiques à l'usage des amateurs"
    Édité par la SAF - 1986
    Retourne un float
    """

    joArr = date.split("-")
```

```

for i in range(len(joArr)):
    joArr[i] = float(joArr[i])

YYYY = float(joArr[0]) # année
MM    = float(joArr[1]) # mois
jour  = float(joArr[2]) # jour

if MM > 2:
    an    = YYYY
    mois  = MM
else:
    an    = YYYY - 1
    mois  = MM + 12

determinant = YYYY + MM / 100 + jour / 10000

if determinant >= 1582.1015:
    A = int(an / 100)
    B = 2 - A + int(A / 4)
else:
    B = 0

jj = int(365.25 * an) + int(30.6001*(mois + 1)) + jour + 1720994.5 + B

UT = sexaToDecimal(time, False)

return jj + (UT / 24)

```

3. Le programme complet avec l'ensemble du code.

Le code proposé n'est pas optimisé, mais conçu pour être facile à lire.

```

#!/usr/bin/python3
# -*- coding: utf-8 -*-

# ~ Calcul du jour julien en utilisant un formulaire
# ~ de Jean MEEUS publié dans :
# ~ Calculs astronomiques à l'usage des amateurs
# ~ SAF - 1986

# ~ Par Alain Leraut - septembre 2019
# code pour Python 3

# ~ ##### Fonctions #####

def sexaToDecimal(ang = "04:34:00.0", horaire = False):
    """
    Sexagésimal vers décimal
    Retourne un float
    """
    anghost = ang.split(":")

```

```

for i in range(len(anglist)):
    anglist[i] = float(anglist[i])
decimal = (anglist[1] / 60) + (anglist[2] / 3600)
if anglist[0] < 0:
    ang = anglist[0] - decimal
else:
    ang = anglist[0] + decimal

if horaire == True: # cas des ascensions droites
    ang *= 15

return ang

```

```

def jdMeeus(date = "1957-10-04", time = "00:00:00.0" ):
    """

```

```

    Calcul du jour julien selon la formule de Jean Meeus
    page 18 de "Calculs astronomiques à l'usage des amateurs"
    Édité par la SAF - 1986
    Retourne un float
    """

```

```

    joArr = date.split("-")
    for i in range(len(joArr)):
        joArr[i] = float(joArr[i])

```

```

    YYYY = float(joArr[0]) # année
    MM = float(joArr[1]) # mois
    jour = float(joArr[2]) # jour

```

```

    if MM > 2:
        an = YYYY
        mois = MM
    else:
        an = YYYY - 1
        mois = MM + 12

```

```

    determinant = YYYY + MM / 100 + jour / 10000

```

```

    if determinant >= 1582.1015:

```

```

        A = int(an / 100)
        B = 2 - A + int(A / 4)

```

```

    else:
        B = 0

```

```

    jj = int(365.25 * an) + int(30.6001*(mois + 1)) + jour + 1720994.5 + B

```

```

    UT = sexaToDecimal(time, False)

```

```

    return jj + (UT / 24)

```

```
# ~ ##### Partie principale #####
```

```
jour, heure = "1957-10-04", "12:00:00.0"
```

```
JJ = jdMeeus(jour, heure)
```

```
print("Jour julien pour date : ", jour, " et ", heure, " heures : ", JJ)
```

```
jour, heure = "333-01-27", "12:00:00.0"
```

```
JJ = jdMeeus(jour, heure)
```

```
print("Jour julien pour date : ", jour, " et ", heure, " heures : ", JJ)
```