天测天力大作业

2020 年 11 月 7 日

姓名：李雪劼
学号：201800401058
班级：2018 级空间班

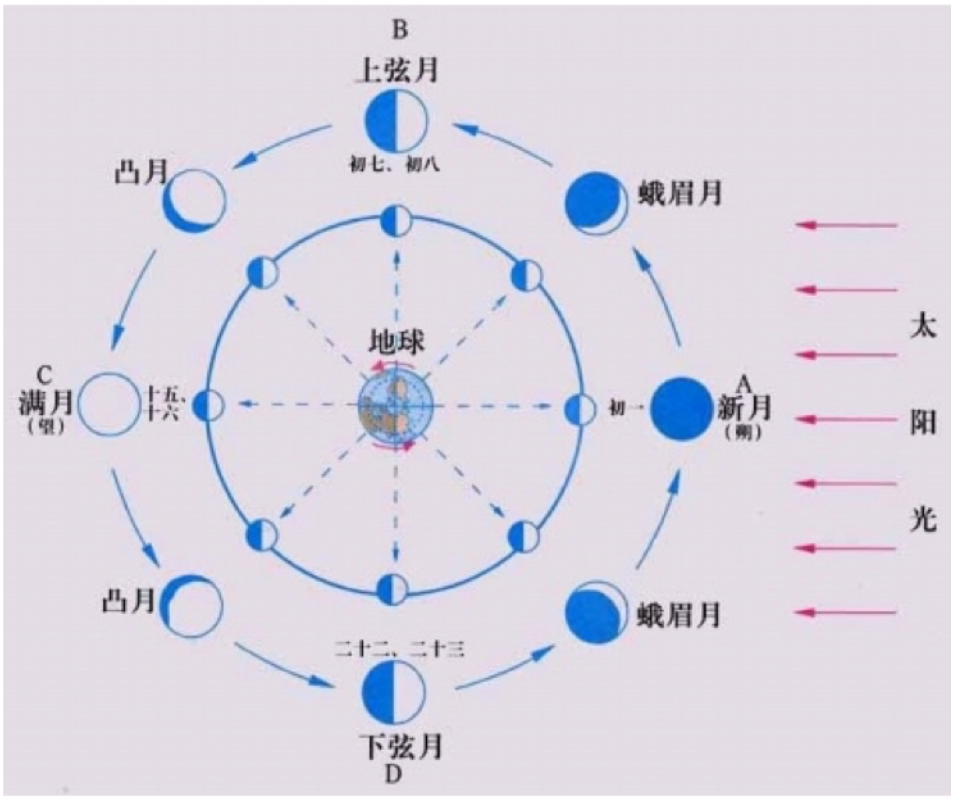
1 题目介绍

计算 2020 年内阴历的

(1)朔时刻(精确到分钟)
(2)望时刻(精确到分钟)
(3)并据此给出年内阴历与阳历的对应日期(给出朔、望日的对应即可)

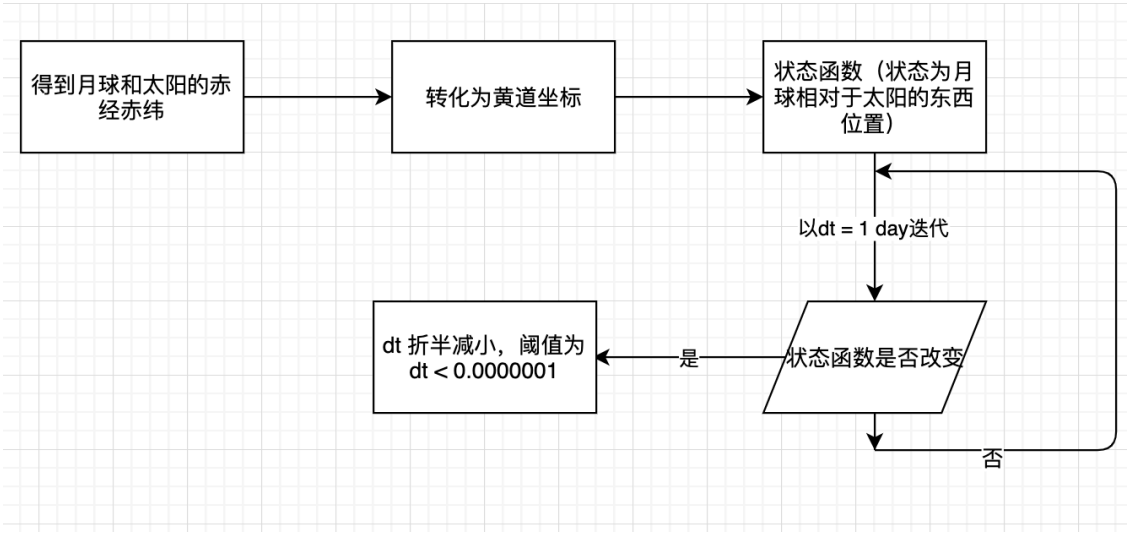
2 原理分析

朔日：太阳黄经与月球黄经一致
望日：太阳黄经与月球黄经相差 180°



3 模型建立

对于本问题，我选择构建：状态函数变化 + 步长折半减小算法求解



4 程序代码

4.1 准备条件

1. Python 安装可以使用 Anaconda(具体安装步骤可参考Anaconda 安装步骤)

2. ehpem package 安装可以参考 https://pypi.org/project/ephem/
3. 为了完成不同的计算任务，可能需要不同的库，常见如 scipy, numpy, matplotlib... 可以视情况使用命令 `conda install xxx(库名)` 安装配置。

4.2 具体代码实现

[1]:
```
from ephem import *
import math
import datetime
import ephem
```

4.2.1 My Way to 计算下一个朔望日

[2]:
```
event=[] # 不同天体不同状态对应的天象名称
event.append({1:"新　月", 2:"满　月"}) # 月亮
# 设置数字对应的天体名称
body={0:"月亮"}
```

[3]:
```
# 迭代求天象时间
def iteration(jd, n, sta):  #jd: 要求的开始时间，n: 天体的代号，sta: 不同的状态函数
    s1 = sta(jd, n)  # 初始状态
    s0 = s1
    dt = 1.0  # 初始时间改变量设为 1 天
    while True:
        jd += dt
        s = sta(jd, n)
        if s0 != s:
            s0 = s
            dt = -dt/2  # 使时间改变量折半减小
        if abs(dt) < 0.0000001 and s == s1:  #s==s1 是为了让求得的时间在天象发生之前
            break
    return jd
```

```
# 计算黄经
def ecliptic_coor(jd_utc, n):
    if n==0:
        p=Moon(jd_utc)# 月球
    else:
        p=Sun(jd_utc)# 太阳
    eq = Equatorial(p.ra, p.dec, epoch=jd_utc)  # 求天体的视赤经视赤纬（epoch 设为所求时间就是视赤经视赤纬）
    ec = Ecliptic(eq)  # 赤经赤纬转到黄经黄纬
    return ec.lon, ec.lat  # 返回黄经


# 给定两个黄经 a1 和 a2，判断 a2 在 a1 的哪一侧，东（设为 0），西（设为 1）
def east_west_angle(a1, a2):
    if abs(a1-a2) < math.pi: # 意味着两个天体之间没有跨越黄经 0 点
        if a2 < a1:
            return 1
        else:
            return 0
    else:
        if a2 < a1:
            return 0
        else:
            return 1


# 设 m 为太阳，求相对于太阳的参数
# 求黄经差
def ecliptic_angle(jd, m, n):
    a1 = ecliptic_coor(jd, m)
    a2 = ecliptic_coor(jd, n)
    if abs(a1[0]-a2[0]) < math.pi: # 意味着两个天体之间没有跨越黄经 0 点
        return abs(a1[0] - a2[0])
    else:
        return 2*math.pi - abs(a1[0]-a2[0])
```

```
# 根据黄经判断天体 n 在太阳的哪一侧，东（设为 0）或者西（设为 1）
def east_west(jd_utc, n):
    a1 = ecliptic_coor(jd_utc, -1)
    a2 = ecliptic_coor(jd_utc, n)
    return east_west_angle(a1[0], a2[0])


# 判断并输出计算的天象名称
def print_time(jd, n):
    i = 0
    s = 0
    e = ecliptic_angle(jd, -1, n)*180 / math.pi # 判断黄经差值
    if abs(180-e) < 1:
        s = 2
    elif e < 1:
        s = 1
    print("{0}: {1}".format(event[i][s], Date(jd+1/3)))


# 计算天体 n 在 jd 时间时的下个满月/新月时间
def next_conjunction(jd, n):
    return iteration(jd, n, east_west)


# 求未来 1 次天体处于特殊几何位置的时间
def event_time(jd, n, num): # 分别输入时间（ephem 儒略日），天体，要计算的接下来要发生的天象个数
    jd0 = jd
    k = 0
    while True:
        jd1 = next_conjunction(jd0, n)
        print_time(jd1, n)
        k += 1
        if k == num:
            break
        jd0 = jd1 + 2 # 加上 2 天以改变天体状态
    return jd1
```

[4]:
```
ini_jd=ephem.date('2020/1/1')
end_date = ephem.date('2020/12/31')
while (1):
    jd = ini_jd
    if jd < end_date:
        jd = event_time(ini_jd, 0, 2)
        ini_jd = jd + 2
    else:
        break
```

```
满  月: 2020/1/11 03:21:17
新  月: 2020/1/25 05:41:59
满  月: 2020/2/9 15:33:16
新  月: 2020/2/23 23:32:00
满  月: 2020/3/10 01:47:43
新  月: 2020/3/24 17:28:11
满  月: 2020/4/8 10:35:03
新  月: 2020/4/23 10:25:49
满  月: 2020/5/7 18:45:11
新  月: 2020/5/23 01:38:49
满  月: 2020/6/6 03:12:21
新  月: 2020/6/21 14:41:25
满  月: 2020/7/5 12:44:22
新  月: 2020/7/21 01:32:55
满  月: 2020/8/3 23:58:43
新  月: 2020/8/19 10:41:37
满  月: 2020/9/2 13:22:02
新  月: 2020/9/17 19:00:10
满  月: 2020/10/2 05:05:13
新  月: 2020/10/17 03:31:01
满  月: 2020/10/31 22:49:07
新  月: 2020/11/15 13:07:09
满  月: 2020/11/30 17:29:40
新  月: 2020/12/15 00:16:33
满  月: 2020/12/30 11:28:11
新  月: 2021/1/13 13:00:09
```

4.2.2 Use Ephem Function to 计算下一个朔望日

[107]:
```
full_moon_date = []
ini_date = '2020/1/1'
end_date = '2020/12/31'
while (1):
    d = next_full_moon(ini_date)
    if julian_date(d) <  julian_date(end_date):
        d = Date(ephem.date(d+1/3))
        print(d)
        full_moon_date.append(str(d))
        ini_date = d
    else:
        break
```

```
2020/1/11 03:21:17
2020/2/9 15:33:16
2020/3/10 01:47:43
2020/4/8 10:35:03
2020/5/7 18:45:11
2020/6/6 03:12:21
2020/7/5 12:44:22
2020/8/3 23:58:43
2020/9/2 13:22:02
2020/10/2 05:05:13
2020/10/31 22:49:07
2020/11/30 17:29:40
2020/12/30 11:28:11
```

[106]:
```
new_moon_date = []
ini_date = '2020/1/1'
end_date = '2020/12/31'
while (1):
    d = next_new_moon(ini_date)
    if julian_date(d) <  julian_date(end_date):
        d = Date(ephem.date(d+1/3))
        print(d)
        new_moon_date.append(str(d))
        ini_date = d
    else:
```

```
        break
```

```
2020/1/25 05:41:59
2020/2/23 23:32:00
2020/3/24 17:28:11
2020/4/23 10:25:49
2020/5/23 01:38:49
2020/6/21 14:41:25
2020/7/21 01:32:55
2020/8/19 10:41:37
2020/9/17 19:00:10
2020/10/17 03:31:01
2020/11/15 13:07:09
2020/12/15 00:16:33
```

4.2.3 Transfrom to Lunar Day

[110]:
```
import datetime

def days(str1,str2):
    date1=datetime.datetime.strptime(str1[0:-9],"%Y/%m/%d")
    date2=datetime.datetime.strptime(str2[0:-9],"%Y/%m/%d")
    num=(date1-date2).days
    return num
```

[108]:
```
# full_moon_date = full_moon_date[1:]
full_moon_date
```

[108]:
```
['2020/2/9 15:33:16',
 '2020/3/10 01:47:43',
 '2020/4/8 10:35:03',
 '2020/5/7 18:45:11',
 '2020/6/6 03:12:21',
 '2020/7/5 12:44:22',
 '2020/8/3 23:58:43',
 '2020/9/2 13:22:02',
 '2020/10/2 05:05:13',
 '2020/10/31 22:49:07',
 '2020/11/30 17:29:40',
```

```
 '2020/12/30 11:28:11']
```

[109]:
```
new_moon_date
```

[109]:
```
['2020/1/25 05:41:59',
 '2020/2/23 23:32:00',
 '2020/3/24 17:28:11',
 '2020/4/23 10:25:49',
 '2020/5/23 01:38:49',
 '2020/6/21 14:41:25',
 '2020/7/21 01:32:55',
 '2020/8/19 10:41:37',
 '2020/9/17 19:00:10',
 '2020/10/17 03:31:01',
 '2020/11/15 13:07:09',
 '2020/12/15 00:16:33']
```

[111]:
```
delta_day = []
for d1, d2 in zip(full_moon_date, new_moon_date):
    delta_day.append(days(d1, d2))
delta_day
```

[111]:
```
[15, 16, 15, 14, 14, 14, 13, 14, 15, 14, 15, 15]
```

5 结果展示

朔			
Python			PMO
阴历	阳历	hh:mm:ss	hh:mm
正月初一	2020/1/25	5:41:59	5:42
二月初一	2020/2/23	23:32:00	23:32
三月初一	2020/3/24	17:28:11	17:28
四月初一	2020/4/23	10:25:49	10:26
闰四月初一	2020/5/23	1:38:49	1:39
五月初一	2020/6/21	14:41:25	14:41
六月初一	2020/7/21	1:32:55	1:33
七月初一	2020/8/19	10:41:37	10:42
八月初一	2020/9/17	19:00:10	19:00
九月初一	2020/10/17	3:31:01	3:31
十月初一	2020/11/15	13:07:09	13:07
十一月初一	2020/12/15	0:16:33	0:17

望			
Python			PMO
阴历	阳历	hh:mm:ss	hh:mm
正月十六	2020/2/9	15:33:16	15:33
二月十七	2020/3/10	1:47:43	1:48
三月十六	2020/4/8	10:35:03	10:35
四月十五	2020/5/7	18:45:11	18:45
闰四月十五	2020/6/6	3:12:21	3:12
五月十五	2020/7/5	12:44:22	12:44
六月十四	2020/8/3	23:58:43	23:59
七月十五	2020/9/2	13:22:02	13:22
八月十六	2020/10/2	5:05:13	5:05
九月十五	2020/10/31	22:49:07	22:49
十月十六	2020/11/30	17:29:40	17:30
十一月十六	2020/12/30	11:28:11	11:28

与 PMO（紫金山天文台）给出的天文年历在分钟精度量级上保持一致无差异