

## Step 1: By Hand

15th of October, 1999 to 25th of October, 1999

October						
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

$$25 - 15 = 10$$

23rd of October, 1999 to 1st of December, 1999

October						
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

$$31 - 23 = 8$$

November						
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

$$30$$

December						
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

$$1$$

$$8 + 30 + 1 = 40$$

21st of October, 1999 to 4th of March, 2004

1999																						
October								November								December						
1	2	3	4	5	6	7		1	2	3	4	5	6	7		1	2	3	4	5	6	7
8	9	10	11	12	13	14		8	9	10	11	12	13	14		8	9	10	11	12	13	14
15	16	17	18	19	20	21		15	16	17	18	19	20	21		15	16	17	18	19	20	21
22	23	24	25	26	27	28		22	23	24	25	26	27	28		22	23	24	25	26	27	28
29	30	31						29	30							29	30	31				
31 - 21 = 10								30								31						

2000

366

2001

365

2002

365

2003

365

2004																						
January								February								March						
1	2	3	4	5	6	7		1	2	3	4	5	6	7		1	2	3	4	5	6	7
8	9	10	11	12	13	14		8	9	10	11	12	13	14		8	9	10	11	12	13	14

15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

31

15	16	17	18	19	20	21
22	23	24	25	26	27	28
29						

29

15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

4

$$10 + 30 + 31 + 366 + 365 + 365 + 365 + 31 + 29 + 4 = 1596$$

## Step 2: Approach

Prompt the user for the start year, start month, start day, end year, end month, and end day. To get the days between the two dates, not including the start day, I'll use a series of if statements to determine which of the three parts of the date is different then for each number of months/years between them add the number of days in each month and year. Then also if at least the months of the dates are different then add the days between the start day to the end of it's month and the end day as well as the full months/years between them.

## Step 3: Pseudocode

```

1  get_days_in_month( month, year )
2      SWITCH month
3          CASE 1 OR 3 OR 5 OR 7 OR 8 OR 10 OR 12
4              days ← 31
5          CASE 4 OR 6 OR 9 OR 11
6              days ← 30
7          CASE 2
8              IF is_leap_year( year )
9                  days ← 29
10             ELSE
11                 days ← 28
12      RETURN days
13
14
15  PROMPT for startYear
16  PROMPT for startMonth
17  PROMPT for startDay
18
19  PROMPT for endYear
20  PROMPT for endMonth
21  PROMPT for endDay

```

```

22
23 IF startYear = endYear
24     IF startMonth = endMonth
25         IF startDay = endDay
26             daysBetween ← 0      // Executes when there is no difference in the dates
27
28         ELSE      // Executes when the days are different
29             // Gets days between the start day and end day
30             daysBetween ← endDay - startDay
31
32     ELSE      // Executes when months are different
33         // Gets days in the start month after the start day
34         daysBetween ← get_days_in_month( startMonth, startYear) - startDay
35
36         // Gets days from all the months after the start up to an NOT including the end month
37         FOR i_month ← startMonth + 1 ... endMonth - 1
38             daysBetween ← daysBetween + get_days_in_month( i_month )
39
40         // Gets days in the end month
41         daysBetween ← daysBetween + endDay
42
43 ELSE      // Executes when years are different
44     // Gets days in the start month after the start day
45     daysBetween ← get_days_in_month( startMonth, startYear) - startDay
46
47     // Gets days from all the months after the start month through end of the year
48     FOR i_month ← startMonth + 1 ... 12
49         daysBetween ← daysBetween + get_days_in_month( i_month, startMonth )
50
51     // Gets days from each year between the start year and end year
52     FOR i_year ← startYear + 1 ... endYear - 1
53         IF is_leap_year( i_year )
54             daysBetween ← daysBetween + 366
55         ELSE
56             daysBetween ← daysBetween + 365
57
58     // Gets days from each month preceding the end month in the end year
59     FOR i_month ← 1 ... endMonth - 1
60         daysBetween ← daysBetween + get_days_in_month( i_month, endYear )
61
62     // Gets days in the end month
63     daysBetween ← daysBetween + endDay
64
65
66 PUT daysBetween

```

## Step 4: Copilot

```
function daysBetweenDates():
    // Prompt user for start and end dates
    startYear = input("Enter start year: ")
    startMonth = input("Enter start month: ")
    startDay = input("Enter start day: ")
    endYear = input("Enter end year: ")
    endMonth = input("Enter end month: ")
    endDay = input("Enter end day: ")

    // Initialize variables
    totalDays = 0
    daysInMonth = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]

    // Check for leap year
    def isLeapYear(year):
        if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
            return True
        return False

    // Adjust days in February for leap years
    if isLeapYear(startYear):
        daysInMonth[1] = 29

    // Calculate days from the start date to the end of the start month
    if startYear == endYear and startMonth == endMonth:
        totalDays += endDay - startDay
    else:
        totalDays += daysInMonth[startMonth - 1] - startDay + 1

    // Calculate days for the full months in between
    for year in range(startYear, endYear + 1):
        for month in range(1, 13):
            if year == startYear and month <= startMonth:
                continue
            if year == endYear and month > endMonth:
                break
            totalDays += daysInMonth[month - 1]

    // Calculate days for the end month
    if startYear != endYear or startMonth != endMonth:
        totalDays += endDay

    return totalDays - 1 // Exclude the start day
```

## Step 5: Compare and Contrast

Provide an analysis as to the pros and cons of the two solutions?

- Mine accounts for if the years between the dates are leap years while copilots does not, only the start year.
- I like the idea of using a list for the months, but it doesn't do as well at accounting for leap years so I'm just going to stick with what I came up with.
- After getting rid of the innermost IF, mine is easier to read

How can your solution be improved based on what Copilot provided?

- I can get rid of my IF statement comparing the start and end days, it's unnecessary.
- I also made a mistake on line 49 putting the wrong variable into the function parameter

How can Copilot's solution be improved based on what you know?

- Use better formatting for pseudocode, it looks more like just python code.
- Account for leap years better.

Does the pseudocode in Step 3 and Step 4 match the algorithm you performed in Step 1?

- They do, but just a little differently and the copilot pseudocode is flawed with the leap years

## Step 6: Update

```
1  get_days_in_month( month, year )
2      SWITCH month
3          CASE 1 OR 3 OR 5 OR 7 OR 8 OR 10 OR 12
4              days ← 31
5          CASE 4 OR 6 OR 9 OR 11
6              days ← 30
7          CASE 2
8              IF is_leap_year( year )
9                  days ← 29
10             ELSE
11                 days ← 28
12      RETURN days
13
14
15  PROMPT for startYear
16  PROMPT for startMonth
17  PROMPT for startDay
18
19  PROMPT for endYear
20  PROMPT for endMonth
```

```

21  PROMPT for endDay
22
23  IF startYear = endYear
24      IF startMonth = endMonth
25          daysBetween ← endDay - startDay
26
27      ELSE    // Executes when months are different
28          // Gets days in the start month after the start day
29          daysBetween ← get_days_in_month( startMonth, startYear) - startDay
30
31          // Gets days from all the months after the start up to an NOT including the end month
32          FOR i_month ← startMonth + 1 ... endMonth - 1
33              daysBetween ← daysBetween + get_days_in_month( i_month )
34
35          // Gets days in the end month
36          daysBetween ← daysBetween + endDay
37
38  ELSE    // Executes when years are different
39      // Gets days in the start month after the start day
40      daysBetween ← get_days_in_month( startMonth, startYear) - startDay
41
42      // Gets days from all the months after the start month through end of the year
43      FOR i_month ← startMonth + 1 ... 12
44          daysBetween ← daysBetween + get_days_in_month( i_month, startYear )
45
46      // Gets days from each year between the start year and end year
47      FOR i_year ← startYear + 1 ... endYear - 1
48          IF is_leap_year( i_year )
49              daysBetween ← daysBetween + 366
50          ELSE
51              daysBetween ← daysBetween + 365
52
53      // Gets days from each month preceding the end month in the end year
54      FOR i_month ← 1 ... endMonth - 1
55          daysBetween ← daysBetween + get_days_in_month( i_month, endYear )
56
57      // Gets days in the end month
58      daysBetween ← daysBetween + endDay
59
60
61  PUT daysBetween

```

## Step 7: Trace

Days between November 17th, 2002 and April 6th, 2004.

line number	daysBetween	i_month	i_year	days	startYear	startMonth	startDay	endYear	endMonth	endDay	
21	/	/	/	/	2002	11	17	2004	4	6	
6	/	/	/	30	2002	11	17	2004	4	6	
40	13	/	/	30	2002	11	17	2004	4	6	
43	13	12	/	30	2002	11	17	2004	4	6	
4	13	12	/	31	2002	11	17	2004	4	6	
44	44	12	/	31	2002	11	17	2004	4	6	
47	44	12	2003	31	2002	11	17	2004	4	6	
51	409	12	2003	31	2002	11	17	2004	4	6	
54	409	1	2003	31	2002	11	17	2004	4	6	
4	409	1	2003	31	2002	11	17	2004	4	6	
55	440	1	2003	31	2002	11	17	2004	4	6	
54	440	2	2003	31	2002	11	17	2004	4	6	
9	440	2	2003	29	2002	11	17	2004	4	6	
55	469	2	2003	29	2002	11	17	2004	4	6	
54	469	3	2003	29	2002	11	17	2004	4	6	
4	469	3	2003	31	2002	11	17	2004	4	6	
55	500	3	2003	31	2002	11	17	2004	4	6	
58	506	3	2003	31	2002	11	17	2004	4	6	



## Step 8: Efficiency

```
1  get_days_in_month( month, year )      // 0(1)
2      SWITCH month
3          CASE 1 OR 3 OR 5 OR 7 OR 8 OR 10 OR 12
4              days ← 31      // 0(1)
5          CASE 4 OR 6 OR 9 OR 11
6              days ← 30      // 0(1)
7          CASE 2
8              IF is_leap_year( year )
9                  days ← 29 // 0(1)
10             ELSE
11                 days ← 28 // 0(1)
12      RETURN days
13
14
15  PROMPT for startYear      // 0(1)
16  PROMPT for startMonth    .
17  PROMPT for startDay      .
18
19  PROMPT for endYear        .
20  PROMPT for endMonth       .
21  PROMPT for endDay         // 0(1)
22
23  IF startYear = endYear
24      IF startMonth = endMonth
25          daysBetween ← endDay - startDay    // 0(1)
26
27      ELSE    // Executes when months are different
28          // Gets days in the start month after the start day
29          daysBetween ← get_days_in_month( startMonth, startYear ) - startDay    // 0(1)
30
31          // Gets days from all the months after the start up to an NOT including the end month
32          FOR i_month ← startMonth + 1 ... endMonth - 1    // 0(n)
33              daysBetween ← daysBetween + get_days_in_month( i_month )    // 0(1)
34
35          // Gets days in the end month
36          daysBetween ← daysBetween + endDay    // 0(1)
37
38  ELSE    // Executes when years are different
39      // Gets days in the start month after the start day
40      daysBetween ← get_days_in_month( startMonth, startYear ) - startDay    // 0(1)
41
42      // Gets days from all the months after the start month through end of the year
43      FOR i_month ← startMonth + 1 ... 12 // 0(n)
44          daysBetween ← daysBetween + get_days_in_month( i_month, startYear )    // 0(1)
```

```

45
46 // Gets days from each year between the start year and end year
47 FOR i_year ← startYear + 1 ... endYear - 1 // 0(n)
48     IF is_leap_year( i_year )
49         daysBetween ← daysBetween + 366 // 0(1)
50     ELSE
51         daysBetween ← daysBetween + 365 // 0(1)
52
53 // Gets days from each month preceding the end month in the end year
54 FOR i_month ← 1 ... endMonth - 1 // 0(n)
55     daysBetween ← daysBetween + get_days_in_month( i_month, endYear ) // 0(1)
56
57 // Gets days in the end month
58 daysBetween ← daysBetween + endDay // 0(1)
59
60
61 PUT daysBetween // 0(1)

```

**Total efficiency =  $O(n)$**