


```

40         // 若权值总和大于50，计入答案
41         if (cnt > 50) ans++;
42
43         // 若到达目标日期2024年4月13日，输出结果并结束
44         if (year == 2024 && month == 4 && day == 13) {
45             printf("%d", ans);
46             return;
47         }
48     }
49 }
50 }
51 }
52 int main() { // 日期范围: 2000-01-01到2024-04-13
53     solve();
54 }

```

- Java

```

1  public class Main {
2      // 每月天数数组，平年默认2月28天
3      static int[] months = {0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30,
4      31};
5      // 对应数字的权值数组
6      static int[] hz = {13, 1, 2, 3, 5, 4, 4, 2, 2, 2};
7
8      // 判断闰年函数
9      static boolean isLeapYear(int year) {
10         // 闰年条件: 是400的倍数，或者是4的倍数但不是100的倍数
11         return (year % 400 == 0) || (year % 4 == 0 && year % 100 != 0);
12     }
13
14     // 主逻辑处理函数
15     public static void solve() {
16         int ans = 0; // 符合条件的日期计数
17         for (int year = 2000; year <= 2024; year++) {
18             // 设置当前年份的2月天数
19             months[2] = isLeapYear(year) ? 29 : 28;
20
21             // 遍历每月
22             for (int month = 1; month <= 12; month++) {
23                 // 遍历每天
24                 for (int day = 1; day <= months[month]; day++) {
25                     int cnt = 0; // 当前日期的权值总和
26                     int y1, y2, y3, y4, m1, m2, d1, d2;
27
28                     // 提取年份、月份、日期的各个位
29                     y1 = year / 1000;
30                     y2 = year / 100 % 10;
31                     y3 = year / 10 % 10;
32                     y4 = year % 10;
33                     m1 = month / 10;
34                     m2 = month % 10;
35                     d1 = day / 10;
36                     d2 = day % 10;
37
38                     // 计算权值总和
39                     cnt += hz[y1] + hz[y2] + hz[y3] + hz[y4]

```

```

39         + hz[m1] + hz[m2] + hz[d1] + hz[d2];
40
41         // 若权值总和大于50，计入答案
42         if (cnt > 50) ans++;
43
44         // 若到达目标日期2024年4月13日，输出结果并结束
45         if (year == 2024 && month == 4 && day == 13) {
46             System.out.println(ans);
47             return;
48         }
49     }
50 }
51 }
52 }
53
54 public static void main(String[] args) { // 日期范围：2000-01-01到2024-
04-13
55     solve();
56 }
57 }
58

```

- Python

```

1  # 每月天数数组，平年默认2月28天
2  months = [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]
3  # 对应数字的权值数组
4  hz = [13, 1, 2, 3, 5, 4, 4, 2, 2, 2]
5
6  # 判断闰年函数
7  def is_leap_year(year):
8      # 闰年条件：是400的倍数，或者是4的倍数但不是100的倍数
9      return year % 400 == 0 or (year % 4 == 0 and year % 100 != 0)
10
11 # 主逻辑处理函数
12 def solve():
13     ans = 0 # 符合条件的日期计数
14     for year in range(2000, 2025):
15         # 设置当前年份的2月天数
16         months[2] = 29 if is_leap_year(year) else 28
17
18         # 遍历每月
19         for month in range(1, 13):
20             # 遍历每天
21             for day in range(1, months[month] + 1):
22                 cnt = 0 # 当前日期的权值总和
23
24                 # 提取年份、月份、日期的各个位
25                 y1, y2, y3, y4 = year // 1000, year // 100 % 10, year // 10
% 10, year % 10
26                 m1, m2 = month // 10, month % 10
27                 d1, d2 = day // 10, day % 10
28
29                 # 计算权值总和
30                 cnt += hz[y1] + hz[y2] + hz[y3] + hz[y4] \
31                     + hz[m1] + hz[m2] + hz[d1] + hz[d2]
32

```

```
33         # 若权值总和大于50，计入答案
34         if cnt > 50:
35             ans += 1
36
37         # 若到达目标日期2024年4月13日，输出结果并结束
38         if year == 2024 and month == 4 and day == 13:
39             print(ans)
40             return
41
42     # 日期范围：2000-01-01到2024-04-13
43     solve()
44
```