

2015 年度 大問 5

hari64boli64 (hari64boli64@gmail.com)

2023 年 5 月 6 日

1 問題

$$D_i = \{j \in \mathbb{N} | 1 \leq j < i, S[j] = S[i]\}$$

$$d(i) = \begin{cases} i - \max D_i & (D_i \neq \emptyset) \\ i & (D_i = \emptyset) \end{cases}$$

2 解答

5.py が解答。

ソースコード 1 answer

```
1 import math
2 import random
3 from collections import defaultdict
4
5 random.seed(0)
6
7
8 def makeProblem(size):
9     sigma = random.randint(1, size)
10    n = random.randint(1, size)
11    A = list(range(1, sigma + 1))
12    S = [random.choice(A) for _ in range(n)]
13    k = len(set(S))
14    return sigma, n, S, k
15
16
17 def problem1(sigma, n, S):
```

```

18 # 問題(1)の解答
19
20 # 1. max(D_i)を求めるのに必要な、
21 #     各 A の要素に関して最後の出現位置を記録する配列を用意する
22 #     空間 O(sigma) 時間 O(sigma)
23 last = [0 for _ in range(sigma)]
24
25 # 2. 各 i について、d(i)を計算する
26 #     空間 O(1) 時間 O(n)
27 for i in range(n):
28     yield (i + 1) - last[S[i] - 1]
29     last[S[i] - 1] = i + 1
30
31
32 def problem2(sigma, n, S):
33     # 問題(2)の解答
34
35     # 1. max(D_i)を求めるのに必要な、
36     #     各
37     #     A の要素に関して最後の出現位置を記録する配列を辞書で持つ
38     #     空間 O(k) 時間 O(1)
39     last = defaultdict(int)
40
41     # 2. 各 i について、d(i)を計算する
42     #     空間 O(1) 時間 O(n log k)
43     for i in range(n):
44         # この計算量が O(log k) になる
45         yield (i + 1) - last[S[i] - 1]
46         last[S[i] - 1] = i + 1
47
48 def problem3(sigma, n, S):
49     # 問題(3)の解答
50
51     # ならし計算量の解析に近い
52     # S に現れる要素それぞれに注目して見ると、
53     # 総計で見る回数は必ず n で抑えられる
54     # よって、全体で O(nk)
55
56     for i in range(n):
57         for j in range(i - 1, -1, -1):
58             if S[j] == S[i]:
59                 yield i - j
60                 break
61         else:
62             yield i + 1
63

```

```

64
65 def trial():
66     sigma, n, S, k = makeProblem(size=10)
67     print(f"{sigma=}, {n=}, {k=}")
68     print(f"{S=}")
69
70     Ds = [[j + 1 for j in range(i) if S[j] == S[i]] for i in
71           range(n)]
72     ds = [i - max(D) if D else i for i, D in enumerate(Ds, start
73               =1)]
74
75     print(f"{Ds=}")
76     print(f"{ds=}")
77
78     ans1 = list(problem1(sigma, n, S))
79     ans2 = list(problem2(sigma, n, S))
80     ans3 = list(problem3(sigma, n, S))
81
82     assert ds == ans1, ans1
83     assert ds == ans2, ans2
84     assert ds == ans3, ans3
85
86 def computeSumOfd():
87     # 問題(4)の解答
88
89     #  $\sum d \leq kn$ 
90     #  $\Rightarrow \frac{1}{n} \sum d \leq k$ 
91     # logの凸性より、
92     #  $\frac{1}{n} \sum \log d \leq \log(\frac{1}{n} \sum d) \leq \log k$ 
93     #  $\Rightarrow \sum \log d \leq n \log k$ 
94
95     sigma, n, S, k = makeProblem(size=10000)
96     print(f"{sigma=}, {n=}, {k=}")
97
98     ds = list(problem1(sigma, n, S))
99
100     print(f"{sum(math.log(d) for d in ds)=}")
101     print(f"{n*math.log(k)=}")
102
103
104 def main():
105     for i in range(5):
106         trial()
107         if i != 4:
108             print("-" * 10)

```

```

109
110     print("=" * 10)
111
112     for i in range(5):
113         computeSumOfd()
114         if i != 4:
115             print("-" * 10)
116
117
118 if __name__ == "__main__":
119     main()

```

ソースコード 2 output

```

1  sigma=7, n=7, k=5
2  S=[1, 3, 5, 4, 4, 7, 7]
3  Ds=[[], [], [], [], [4], [], [6]]
4  ds=[1, 2, 3, 4, 1, 6, 1]
5  -----
6  sigma=5, n=8, k=4
7  S=[3, 5, 2, 5, 2, 3, 2, 1]
8  Ds=[[], [], [], [2], [3], [1], [3, 5], []]
9  ds=[1, 2, 3, 2, 2, 5, 2, 8]
10 -----
11 sigma=10, n=5, k=5
12 S=[9, 10, 3, 5, 2]
13 Ds=[[], [], [], [], []]
14 ds=[1, 2, 3, 4, 5]
15 -----
16 sigma=2, n=6, k=2
17 S=[2, 1, 2, 2, 2, 1]
18 Ds=[[], [], [1], [1, 3], [1, 3, 4], [2]]
19 ds=[1, 2, 2, 1, 1, 4]
20 -----
21 sigma=9, n=8, k=6
22 S=[8, 9, 5, 1, 9, 1, 2, 7]
23 Ds=[[], [], [], [], [2], [4], [], []]
24 ds=[1, 2, 3, 4, 3, 2, 7, 8]
25 =====
26 sigma=19, n=8087, k=19
27 sum(math.log(d) for d in ds)=19935.714435979888
28 n*math.log(k)=23811.678024519002
29 -----
30 sigma=1397, n=38, k=35
31 sum(math.log(d) for d in ds)=96.64773012539216
32 n*math.log(k)=135.10322633659771
33 -----
34 sigma=8410, n=3796, k=3077

```

```
35 sum(math.log(d) for d in ds)=26698.671661842043
36 n*math.log(k)=30488.372584722474
37 -----
38 sigma=7194, n=6237, k=4169
39 sum(math.log(d) for d in ds)=45840.86686343042
40 n*math.log(k)=51988.08612754253
41 -----
42 sigma=8530, n=4156, k=3267
43 sum(math.log(d) for d in ds)=29505.23672883638
44 n*math.log(k)=33628.80352261405
```