Тренировочный контест — бэкенд

Задачи   Посылки   Сообщения

E. Занимательная алхимия

✓ Полное решение   </> OK   Go 1.23.0

∧ Исходный код   ⤓ ⧉

```go
1    package main
2
3    import (
4        "bufio"
5        "fmt"
6        "os"
7        "strconv"
8        "strings"
9    )
10
11   const CAP_LIMIT = int(1e9)
12
13   func mustAtoi(s string) int {
14       i, err := strconv.Atoi(s)
15       if err != nil {
16           panic(err)
17       }
18       return i
19   }
20
21   func buildRecipes(lines [][]int) (map[int][]int, map[int]bool) {
22       unreachable := make(map[int]bool)
23       saved := map[int][]int{
24           1: {1, 0},
25           2: {0, 1},
26       }
27
28       for ind, line := range lines {
29           potion := ind + 3
30           if _, exists := unreachable[potion]; exists {
31               continue
32           }
33           if _, exists := saved[potion]; exists {
34               continue
35           }
36
37           stack := []int{potion}
38           temp := map[int][]int{
39               potion: {len(line) - 1, 0, 0},
40           }
41
42           for len(stack) != 0 {
43               curr_potion := stack[len(stack)-1]
44               curr_index := temp[curr_potion][0]
45
46               if curr_index < 0 {
47                   reqA := temp[curr_potion][1]
48                   reqB := temp[curr_potion][2]
49                   saved[curr_potion] = []int{reqA, reqB}
50                   stack = stack[0 : len(stack)-1]
51                   delete(temp, curr_potion)
```

```go
52
53                    if len(stack) != 0 {
54                        parent_potion := stack[len(stack)-1]
55
56                        temp[parent_potion][1] += reqA
57                        temp[parent_potion][2] += reqB
58
59                        if temp[parent_potion][1] > CAP_LIMIT || temp[parent_potion][2] > CAP_LIMIT {
60                            for _, p := range stack {
61                                unreachable[p] = true
62                            }
63                            break
64                        }
65                    }
66                    continue
67                }
68
69                recipe_line := lines[curr_potion-3]
70                x := recipe_line[curr_index]
71                temp[curr_potion][0] -= 1
72
73                if _, exists := saved[x]; exists {
74                    temp[curr_potion][1] += saved[x][0]
75                    temp[curr_potion][2] += saved[x][1]
76
77                    if temp[curr_potion][1] > CAP_LIMIT || temp[curr_potion][2] > CAP_LIMIT {
78                        for _, p := range stack {
79                            unreachable[p] = true
80                        }
81                        break
82                    }
83                } else if _, exists := unreachable[x]; exists {
84                    for _, p := range stack {
85                        unreachable[p] = true
86                    }
87                    break
88                } else if _, exists := temp[x]; exists {
89                    for _, p := range stack {
90                        unreachable[p] = true
91                    }
92                    break
93                } else {
94                    stack = append(stack, x)
95                    new_line := lines[x-3]
96                    temp[x] = []int{len(new_line) - 1, 0, 0}
97                }
98            }
99        }
100
101    return saved, unreachable
102 }
103
104 func turnToIntLine(line []string) []int {
105    intLine := make([]int, 0, len(line))
106
107    for _, x := range line {
108        intLine = append(intLine, mustAtoi(x))
109    }
110
111    return intLine
112 }
113
114 func main() {
115    file, err := os.Open("input.txt")
116    if err != nil {
117        panic(err)
118    }
119    defer file.Close()
120
121    scan := bufio.NewScanner(file)
122    buf := make([]byte, 0, bufio.MaxScanTokenSize)
123    scan.Buffer(buf, 2*1024*1024)
```

```go
124        scan.Scan()
125        N, err := strconv.Atoi(scan.Text())
126        if err != nil {
127            panic(err)
128        }
129
130        lines := make([][]int, 0, N-2)
131
132        for range N - 2 {
133            scan.Scan()
134            line := strings.Split(scan.Text(), " ")[1:]
135            intLine := turnToIntLine(line)
136
137            lines = append(lines, intLine)
138        }
139
140        recipes, unreachable := buildRecipes(lines)
141
142        // Пропускаем кол-во вопросов
143        scan.Scan()
144
145        for scan.Scan() {
146            line := strings.Split(scan.Text(), " ")
147            intLine := turnToIntLine(line)
148            a := intLine[0]
149            b := intLine[1]
150            potion := intLine[2]
151
152            if _, exists := unreachable[potion]; exists {
153                fmt.Print("0")
154            } else if a >= recipes[potion][0] && b >= recipes[potion][1] {
155                fmt.Print("1")
156            } else {
157                fmt.Print("0")
158            }
159        }
160    }
161
```

∨ Отличия от предыдущей посылки

∨ Лог компиляции

№	Вердикт	Ресурсы	Баллы	
1	ok	3ms / 2.13Mb	—	∨
2	ok	3ms / 1.98Mb	—	
3	ok	3ms / 2.00Mb	—	
4	ok	3ms / 2.00Mb	—	
5	ok	3ms / 2.04Mb	—	
6	ok	3ms / 1.99Mb	—	
7	ok	3ms / 2.12Mb	—	
8	ok	3ms / 1.98Mb	—	
9	ok	3ms / 2.00Mb	—	
10	ok	3ms / 2.08Mb	—	
11	ok	3ms / 2.12Mb	—	
12	ok	3ms / 1.87Mb	—	

№	Вердикт	Ресурсы	Баллы
13	ok	3ms / 2.11Mb	—
14	ok	3ms / 1.96Mb	—
15	ok	3ms / 2.01Mb	—
16	ok	3ms / 2.00Mb	—
17	ok	3ms / 2.05Mb	—
18	ok	3ms / 2.09Mb	—
19	ok	3ms / 2.03Mb	—
20	ok	3ms / 1.98Mb	—
21	ok	3ms / 2.08Mb	—
22	ok	3ms / 2.09Mb	—
23	ok	3ms / 2.10Mb	—
24	ok	3ms / 2.03Mb	—
25	ok	3ms / 2.13Mb	—
26	ok	3ms / 2.00Mb	—
27	ok	3ms / 2.00Mb	—
28	ok	3ms / 2.00Mb	—
29	ok	3ms / 2.06Mb	—
30	ok	3ms / 2.16Mb	—
31	ok	3ms / 1.95Mb	—
32	ok	3ms / 2.13Mb	—
33	ok	3ms / 2.11Mb	—
34	ok	0.636s / 29.39Mb	—
35	ok	0.633s / 28.36Mb	—
36	ok	0.648s / 34.55Mb	—
37	ok	0.63s / 28.75Mb	—
38	ok	0.679s / 52.60Mb	—
39	ok	0.688s / 50.02Mb	—
40	ok	0.694s / 50.15Mb	—
41	ok	0.669s / 40.50Mb	—
42	ok	0.687s / 48.86Mb	—
43	ok	480ms / 35.71Mb	—
44	ok	0.593s / 36.74Mb	—
45	ok	0.566s / 30.68Mb	—
46	ok	0.636s / 32.36Mb	—

№	Вердикт	Ресурсы	Баллы
47	ok	0.502s / 31.97Mb	—
48	ok	364ms / 27.59Mb	—
49	ok	0.507s / 29.52Mb	—
50	ok	481ms / 32.36Mb	—
51	ok	0.614s / 29.65Mb	—
52	ok	372ms / 31.07Mb	—
53	ok	303ms / 27.07Mb	—
54	ok	459ms / 29.78Mb	—
55	ok	423ms / 27.07Mb	—
56	ok	0.59s / 33.00Mb	—
57	ok	299ms / 27.46Mb	—
58	ok	0.633s / 32.36Mb	—
59	ok	0.611s / 27.20Mb	—
60	ok	0.639s / 31.46Mb	—
61	ok	0.66s / 31.07Mb	—
62	ok	0.655s / 34.42Mb	—
63	ok	0.659s / 31.46Mb	—
64	ok	0.627s / 27.72Mb	—
65	ok	0.624s / 28.36Mb	—
66	ok	0.64s / 30.30Mb	—
67	ok	0.619s / 27.71Mb	—
68	ok	0.619s / 28.36Mb	—
69	ok	0.604s / 24.63Mb	—
70	ok	0.648s / 32.10Mb	—
71	ok	0.634s / 33.39Mb	—
72	ok	0.624s / 28.75Mb	—
73	ok	0.649s / 32.10Mb	—
74	ok	0.631s / 30.04Mb	—
75	ok	0.64s / 31.33Mb	—
76	ok	0.661s / 36.74Mb	—
77	ok	0.626s / 32.49Mb	—
78	ok	0.646s / 35.20Mb	—
79	ok	0.686s / 33.91Mb	—
80	ok	0.697s / 43.32Mb	—

№	Вердикт	Ресурсы	Баллы
81	ok	0.643s / 27.33Mb	—
82	ok	0.756s / 47.31Mb	—