Тренировочный контест — бэкенд

A. Хитрый шифр

✓ Полное решение </> OK Go 1.23.0

Исходный код

```go
package main

import (
    "bufio"
    "fmt"
    "os"
    "strconv"
    "strings"
)

type Candidate struct {
    surname    string
    name       string
    patronymic string
    birthDay   string
    birthMonth string
    birthYear  string
}

func NewCandidate(data []string) *Candidate {
    return &Candidate{
        surname:    data[0],
        name:       data[1],
        patronymic: data[2],
        birthDay:   data[3],
        birthMonth: data[4],
        birthYear:  data[5],
    }
}

func (c *Candidate) CreateCode() string {
    unique := make(map[rune]bool)
    for _, r := range c.surname + c.name + c.patronymic {
        unique[r] = true
    }
    numberSum := len(unique)

    for _, let := range c.birthDay + c.birthMonth {
        if d, err := strconv.Atoi(string(let)); err == nil {
            numberSum += d * 64
        }
    }

    numberSum += int(strings.ToLower(c.surname)[0]-'a'+1) * 256

    code := fmt.Sprintf("%X", numberSum)
    code = code[len(code)-3:]

    if len(code) < 3 {
        code = strings.Repeat("0", 3-len(code)) + code
    }
```

```
52      return code
53  }
54
55  func main() {
56      file, err := os.Open("input.txt")
57      if err != nil {
58          panic(err)
59      }
60      defer file.Close()
61
62      scanner := bufio.NewScanner(file)
63      scanner.Scan()
64
65      for scanner.Scan() {
66          line := scanner.Text()
67          candidate := NewCandidate(strings.Split(line, ","))
68          fmt.Print(candidate.CreateCode(), " ")
69      }
70  }
71
```

Отличия от предыдущей посылки

Лог компиляции

№	Вердикт	Ресурсы	Баллы
1	ok	3ms / 1.99Mb	—
2	ok	3ms / 2.00Mb	—
3	ok	3ms / 2.00Mb	—
4	ok	3ms / 2.00Mb	—
5	ok	3ms / 2.03Mb	—
6	ok	3ms / 2.05Mb	—
7	ok	3ms / 1.99Mb	—
8	ok	3ms / 2.00Mb	—
9	ok	3ms / 2.00Mb	—
10	ok	3ms / 2.06Mb	—
11	ok	3ms / 1.98Mb	—
12	ok	3ms / 1.96Mb	—
13	ok	55ms / 4.64Mb	—