

# **Олімпіада з інформатики Солом'янського району**

**2009/10 н. р.**

**Автори:**

**Данило Мисак (задачі 1, 2),**

**Олександр Рудик (задача 3)**

## 1. Задача «Числа» (назва програми: `numbers.pas/numbers.cpp`)

Дано 5 чисел. Вказати таке з них, що має найбільшу суму цифр.

### Вхідні дані

У єдиному рядку вхідного файлу `numbers.in` через пробіл записано п'ять натуральних чисел, що не перевищують 100 000.

### Вихідні дані

Єдиний рядок вихідного файлу `numbers.out` має містити число, сума цифр якого найбільша. Відомо, що число, яке має найбільшу суму цифр, лише одне.

### Приклад

| Вхідний файл <code>numbers.in</code> | Вихідний файл <code>numbers.out</code> |
|--------------------------------------|--|
| 712 8 1000 5903 11111                | 5903                                   |

### Пояснення до прикладу

Суми цифр заданих чисел:  $7+1+2=10$ ,  $8=8$ ,  $1+0+0+0=1$ ,  $5+9+0+3=17$ ,  $1+1+1+1+1=5$ . Найбільшу суму, 17, має число 5903. Його й слід вивести.

## 2. Задача «Магазин» (назва програми: `shop.pas/shop.cpp`)

У магазині працюють  $n$  продавців. Відомо, коли кожен продавець приходить на роботу й коли йде з неї. Визначте, скільки годин протягом дня у магазині працюють водночас усі  $n$  продавців.

### Вхідні дані

У першому рядку вхідного файлу `shop.in` міститься натуральне число  $n$  ( $n \leq 1000$ ) — кількість продавців. У наступних  $n$  рядках міститься інформація про всіх працівників магазину: в кожному рядку записано по два натуральних числа — година, о котрій деякий продавець приходить на роботу, й час, коли він іде з магазину (саме в такому порядку). Жодне з цих чисел не перевищує 23. Якщо друге число не перевищує перше, це означає, що продавець залишається у магазині на ніч і йде додому наступного дня.

### Вихідні дані

Єдиний рядок вихідного файлу `shop.out` має містити шукане число — кількість годин (упродовж одного дня), коли всі продавці перебувають у магазині.

### Приклад

| Вхідний файл <code>shop.in</code> | Вихідний файл <code>shop.out</code> |
|-----------------------------------|-------------------------------------|
| 3<br>8 20<br>14 19<br>12 16       | 2                                   |

### Пояснення до прикладу

Всі три продавці перебувають у магазині протягом двох годин — із 14-ї до 16-ї.

### 3. Задача «Судоку» (назва програми: `sudoku.pas/sudoku.cpp`)

Прообраз сучасного судоку, так звані магічні квадрати, знали ще у Стародавньому Китаї. В Європі щось схоже згадують з XVIII століття. Тоді швейцарський математик Леонард Ейлер з'ясував, що в матриці розміром  $9 \times 9$  кожен рядок і кожен стовпчик можна заповнити цифрами від 1 до 9 без повторення. Судоку в сучасному вигляді з'явилася в одному з американських журналів пазлів в 1979 році. Автором ломиголовки був громадянин США, 74-річний архітектор на пенсії Говард Гарнс (Howard Garns). Видавець, журнал Math Puzzles and Logic Problems, дав пазлу ім'я Number Place, яке досі використовують у США. Справжню популярність головоломка здобула у 2005 році, коли японський журнал Nikoli став регулярно друкувати її на своїх сторінках.

Правила судоку такі. Ігрове поле складається з квадрата розміру  $9 \times 9$  клітин, розділеного на менші квадрати-регіони розміру  $3 \times 3$  клітини. Таким чином, все поле налічує 81 клітинку. У деяких з них вже на початку гри розташовані числа від 1 до 9 включно. Мета головоломки — потрібно заповнити вільні клітинки цифрами від 1 до 9 так, щоб в кожному рядку, в кожному стовпчику й у кожному малому квадраті-регіоні розміру  $3 \times 3$  кожна цифра, відмінна від нуля, траплялася лише один раз.

Створіть програму, яка розв'яже ломиголовку судоку.

#### Вхідні дані

Кожний з 9 рядків вхідного файлу `sudoku.in` містить 9 символів, кожен з яких є пробілом або цифрою від 1 до 9 включно. Пробіл відповідає незаповненому полю. Вхідні дані гарантують єдиність розв'язку та можливість його отримання без перебору гіпотез щодо вмісту незаповнених полів.

#### Вихідні дані

Вихідний файл `sudoku.out` повинен містити результат цілковитого заповнення поля судоку, поданого вхідним файлом. Пробілів у вихідному файлі бути не має.

#### Приклад

| Вхідний файл <code>sudoku.in</code> | Вихідний файл <code>sudoku.out</code> |
|-------------------------------------|---------------------------------------|
| 6 5 4                               | 675139824                             |
| 83 4 7                              | 831624975                             |
| 92 761                              | 924857613                             |
| 1 85 6                              | 719385246                             |
| 82 61                               | 358246197                             |
| 2 79 3                              | 246791538                             |
| 629 81                              | 562973481                             |
| 8 4 59                              | 187462359                             |
| 4 7 2                               | 493518762                             |

## Ідеї розв'язання

### Числа

Для кожного з п'яти чисел слід підрахувати суму цифр (наприклад, шляхом послідовного ділення на 10) та визначити, яка зі знайдених сум найбільша — у циклі або шляхом попарного порівняння.

### Магазин

Можна розв'язувати завдання як задачу про знаходження перетину відрізків: додаючи працівників по одному, послідовно знаходити спільний час їхньої роботи. Більш простий спосіб такий. Слід перебрати всі 24 години доби і для кожної перевірити, чи всі працівники є в магазині у відповідний час. Кількість годин, які пройшли цю перевірку, і є шуканою.

### Судоку

Слід у циклі проходити по всіх клітинках поля. Якщо клітинка порожня і є лише один варіант цифри, яку можна вписати у цю клітинку (тобто решта цифр уже записані або в тому ж рядку, або в тому ж стовпчику, або в тому ж квадраті  $3 \times 3$ , в якому розташована клітинка), записуємо у цю клітинку відповідну цифру. Якщо під час чергового проходу поля у жодну з 81 клітинок не було вписано цифру, виходимо з циклу. За умовою задачі («вхідні дані гарантують єдиність розв'язку та можливість його отримання без перебору гіпотез щодо вмісту незаповнених полів»), коли алгоритм завершить свою роботу, вміст усіх порожніх клітинок буде відновлено.