

Множества

Занятие кружка по информатике ФИМЛИ 5

г. Долгопрудный, МО
Грицуляк РТ,
28 сентября 2016

Представление множеств

- Задача с мешком слов:
 - Есть множество слов $\{ w[0], w[1], w[2], \dots, w[n-1] \}$
 - $n < 64$
 - Есть $f(\text{набор}) \rightarrow$ функция, которая работает непонятно как, но выдает нам на каждый набор.
 $\{w[i_1], w[i_2], w[i_3], \dots, w[i_k]\} \rightarrow$ число. Например, это может быть хэш функция.
 - Известно, что количество документов не зависит от порядка слов и количества вхождений слов в список
 - Найти набор на котором функция будет максимальна
 - Решение не должно зависеть от вида функции f .

Мощность(1)

- множества подмножеств из $\{w[1], w[2], w[3], \dots, w[n]\}$ — булеан
- Мощность булеана (количество входящих в него элементов) 2^n
- Действительно, закодируем двоичное число с помощью элемента булеана следующим образом:
- Для подмножества $\{w[i_1], w[i_2], w[i_3], \dots, w[i_k]\}$
число $= 2^{i_1} + 2^{i_2} + 2^{i_3} + \dots + 2^{i_k}$
 - То есть для элемента множества i — если он есть в подмножестве — он есть и на i -й с конца позиции начиная с 0й.

```
int n;

vector<string> words;

cin >> n;

bitset<64> bits(0);

for(int i=0;i<n;i++) {

    string word;

    cin >> word;

    words.push_back(word);

}

int last_max=-1;
```

```
for(int j=0;bits.count()<n;j++)

{

    bitset<64> bitelem(j);

    bits=bitelem;

    vector<string> element;

    for(int k=0;k<n;k++){

        if(bitelem.test(k))

            element.push_back(words[k]);

    }

    int cur_max = f(element);

    if(cur_max>last_max)

        last_max=cur_max;

}
```

- Как будет выглядеть пересечение 2х множеств?
- Объединение?

Bitset<100> a(300); bitset<100> b(900);

- Как будет выглядеть пересечение 2х множеств?

$a \& b$

- Объединение?

$a | b$

Задание

- Паскалистам — на паскале написать свой вариант для 30 слов (для 200 на ***)
.
- C++ - реализовать на `uint64_t` (допустим у нас меньше 64 слов), и для 200 слов на `uint64_t wrds[4]`
- Сравнить результаты на {«one», «two», «three», «four», «five», «six», «seven», «eight», «nine», «ten»}
- Попробовать «взломать» решения друг-друга (первые 2 решивших в пару итд)
- Решение = 700; взлом +100 ; решение «200» - 1500, +100