

## Раздел «Язык Си» . GenBrackets :

## Рекурсивная функция генерации скобочных выражений

Одна из основных идей, используемых при решении задачи, – это погружение задачи в двухпараметрическое множество задач.

Вместо однопараметрической задачи задачи

**Генерация скобочных выражений длины n.**

мы переходим к двухпараметрической задаче

**Генерация скобочных выражений длины n, которые имеют k лишних закрывающихся скобок (то есть являются правильными, если к ним в начало приписать k открывающихся скобок).**

Назовём такие выражения выражениями типа (n,k).

Выражения типа (2n, 0) – это правильные скобочные выражения из n пар скобок.

```
#include<stdio.h>

/* В этот массив мы будем помещать символы скобочного выражения, которое
 */
char expr[100];

/* индекс ячейки в массиве expr, куда нужно поместить следующий символ
 */
int pt = 0;

/* Вывести значение полученного скобочного выражения в стандартный поток вывода
 */
void print_expr() {
    expr[pt] = 0;
    printf("%s\n", expr);
}

/* Сгенерировать в ячейки expr[pt], expr[pt+1], .. expr[pt+n-1] все возможные варианты
 * скобочных выражений типа (n,k).
 */
void gen (int n, int k) {
    if (n == 0) {
        print_expr();
    } else {
        if (n-1 >= k+1) {
            expr [pt] = '('; pt++;
            gen(n-1, k+1); pt--;
        }
        if ( k > 0 ) {
            expr[pt] = ')'; pt++;
            gen(n-1, k-1); pt--;
        }
    }
}

int main () {
    int n;
    scanf("%d", &n); // считываем
    gen(2*n, 0);      // выводим все правильные скобочные выражения из n пар скобок
    return 0;
}
```

-- ArtemVoroztsov - 17 Sep 2007

(с) Материалы раздела "Язык Си" публикуются под лицензией GNU Free Documentation License.

Поиск

Поиск

Раздел «Язык Си»

Главная  
Зачем учить C?  
Определения

Инструменты:

Поиск  
Изменения  
Index  
Статистика

Разделы

Информация  
Алгоритмы  
Язык Си  
Язык Ruby  
Язык  
Ассемблера  
EJ Judge  
Парадигмы  
Образование  
Сети  
Objective C

Login&gt;&gt;