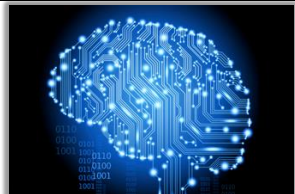


2017

PISL 10

PISL10. Итоги дин. программирования.



Кафедра экономической информатики. Бгуир., 2017

PISL10. Итоги дин. программирования.

- Задача о вечеринке с весами.
- Обзор итогов
- Решения (примеры)

- Ресурс для онлайн обучения stepik.org
- Онлайн сертификация от PISL Center (Россия, г. Санкт-Петербург)

- Материалы: <http://tinyurl.com/ei-pisl>
- Github: <https://github.com/Khmellov/PISL2017-01-26>

Кафедра экономической информатики. Бгуир. 2017

2

PISL10. Итоги дин. программирования.

Максимальное по весу независимое множество в дереве

Вход: дерево в весах на вершинах.

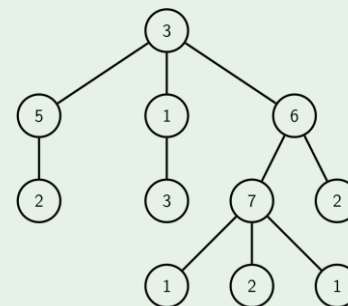
Выход: независимое множество (множество не соединённых друг с другом вершин) максимального суммарного веса.

Кафедра экономической информатики. Бгуир. 2017

3

PISL10. Итоги дин. программирования.

Пример

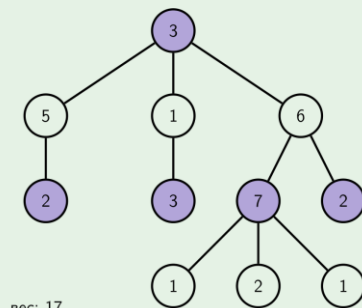


Кафедра экономической информатики. Бгуир. 2017

4

PISL10. Итоги дин. программирования.

Пример

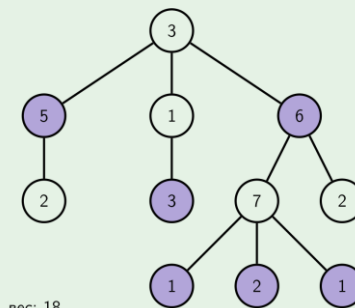


Кафедра экономической информатики. Бгуир. 2017

5

PISL10. Итоги дин. программирования.

Пример



Кафедра экономической информатики. Бгуир. 2017

6

PISL10. Итоги дин. программирования.

Подзадачи и рекуррентное соотношение

- $D[v]$ — максимальный вес независимого множества в поддереве с корнем в v .

Кафедра экономической информатики. Бгуир. 2017

7

PISL10. Итоги дин. программирования.

Подзадачи и рекуррентное соотношение

- $D[v]$ — максимальный вес независимого множества в поддереве с корнем в v .
- Рекуррентное соотношение:

$$D[v] = \max \left\{ w(v) + \sum_{\substack{\text{внуки } w \\ \text{вершины } v}} D[w], \sum_{\substack{\text{дети } w \\ \text{вершины } v}} D[w] \right\}.$$

Кафедра экономической информатики. Бгуир. 2017

8

PISL10. Итоги дин. программирования.

Функция $\text{MAXINDSETTD}(v)$

```
если  $D[v] = \infty$ :
  если  $u \in v$  нет детей:
     $D[v] \leftarrow w(v)$ 
```

PISL10. Итоги дин. программирования.

Функция $\text{MAXINDSETTD}(v)$

```
если  $D[v] = \infty$ :
  если  $u \in v$  нет детей:
     $D[v] \leftarrow w(v)$ 
  иначе:
     $m_1 \leftarrow w(v)$ 
    для всех детей  $u$  вершины  $v$ :
      для всех детей  $w$  вершины  $u$ :
         $m_1 \leftarrow m_1 + \text{MAXINDSETTD}(w)$ 
```

PISL10. Итоги дин. программирования.

Функция $\text{MAXINDSETTD}(v)$

```
если  $D[v] = \infty$ :
  если  $u \in v$  нет детей:
     $D[v] \leftarrow w(v)$ 
  иначе:
     $m_1 \leftarrow w(v)$ 
    для всех детей  $u$  вершины  $v$ :
      для всех детей  $w$  вершины  $u$ :
         $m_1 \leftarrow m_1 + \text{MAXINDSETTD}(w)$ 
     $m_0 \leftarrow 0$ 
    для всех детей  $u$  вершины  $v$ :
       $m_0 \leftarrow m_0 + \text{MAXINDSETTD}(u)$ 
```

PISL10. Итоги дин. программирования.

Функция $\text{MAXINDSETTD}(v)$

```
если  $D[v] = \infty$ :
  если  $u \in v$  нет детей:
     $D[v] \leftarrow w(v)$ 
  иначе:
     $m_1 \leftarrow w(v)$ 
    для всех детей  $u$  вершины  $v$ :
      для всех детей  $w$  вершины  $u$ :
         $m_1 \leftarrow m_1 + \text{MAXINDSETTD}(w)$ 
     $m_0 \leftarrow 0$ 
    для всех детей  $u$  вершины  $v$ :
       $m_0 \leftarrow m_0 + \text{MAXINDSETTD}(u)$ 
     $D[v] \leftarrow \max(m_1, m_0)$ 
  вернуть  $D[v]$ 
```

PISL10. Итоги дин. программирования.

Функция $\text{MAXINDSETD}(v)$

```

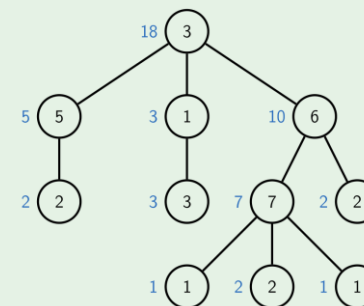
если  $D[v] = \infty$ :
    если у  $v$  нет детей:
         $D[v] \leftarrow w(v)$ 
    иначе:
         $m_1 \leftarrow w(v)$ 
        для всех детей  $u$  вершины  $v$ :
            для всех детей  $w$  вершины  $u$ :
                 $m_1 \leftarrow m_1 + \text{MAXINDSETD}(w)$ 
         $m_0 \leftarrow 0$ 
        для всех детей  $u$  вершины  $v$ :
             $m_0 \leftarrow m_0 + \text{MAXINDSETD}(u)$ 
         $D[v] \leftarrow \max(m_1, m_0)$ 
    вернуть  $D[v]$ 

```

Время работы: $O(|T|)$.

PISL10. Итоги дин. программирования.

Пример



PISL10. Итоги дин. программирования.

Общие принципы

- Проанализировать структуру оптимального решения, чтобы определить подзадачи и рекуррентное соотношение на них.
- Преобразовать рекуррентное соотношение в рекурсивный алгоритм, обернув его мемоизацией и решая подзадачи сверху вниз, от больших к меньшим.
- Преобразовать рекурсивный алгоритм в итеративный, решая подзадачи снизу вверх, от меньших к большему.
- Доказать оценку на время работы и используемую память.
- Восстановить оптимальное решение.
- Проанализировать структуры таблицы и рекурсивных вызовов, чтобы понять, нельзя ли сэкономить память.

PISL10. Итоги дин. программирования.

Подзадачи: обзор

- 1 Числа Фибоначчи: $D[i] = F_i$.
- 2 Наибольшая возрастающая подпоследовательность: $D[i]$ — длина НВП, заканчивающейся в $A[i]$.
- 3 Расстояние редактирования: $D[i, j]$ — расстояние между $A[1 \dots i]$ и $B[1 \dots j]$.
- 4 Рюкзак: $D[w]$ — оптимальная стоимость заполнения рюкзака вместимости w .
- 5 Перемножение последовательности матриц: $D[i, j]$ — стоимость вычисления $A_i \times \dots \times A_j$.
- 6 Независимые множества в деревьях: $D[v]$ — максимальный вес независимого множества в поддереве с корнем в v .

PISL10. Итоги дин. программирования.

Часто используемые подзадачи

- Вход — $A[1 \dots n]$, подзадача — $A[1 \dots i]$.

- Вход — $A[1 \dots n]$, подзадача — $A[i \dots j]$.

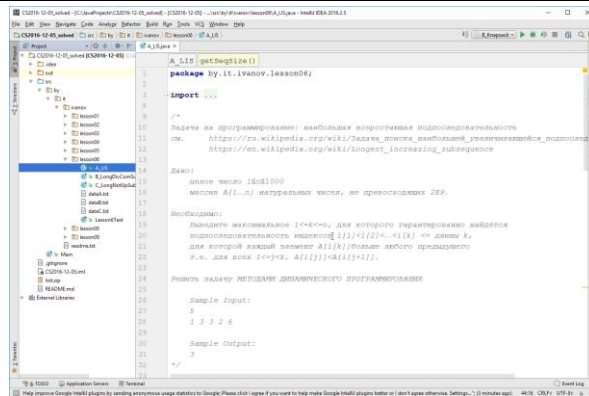
- Вход — $A[1 \dots n]$ и $B[1 \dots m]$, подзадача — $A[1 \dots i]$ и $B[1 \dots j]$.

- Вход — дерево, подзадача — поддереву.


Кафедра экономической информатики. Бгуйр. 2017

17

PISL10. Итоги дин. программирования.



```

package by.it.ivanov.ivanov04;

import java.util.*;

// Задача на программирование: максимальная подпоследовательность
// см. https://ru.wikipedia.org/wiki/Задача_Максима_Максимовича_Умножения_Максимовича_Подпоследовательности
// https://en.wikipedia.org/wiki/Longest_increasing_subsequence

// Дано:
// - массив чисел 1 ≤ a[i] ≤ 1000
// - массив A[1..n] натуральных чисел, не превосходящих 200.
// Необходимо:
// - вывести максимальное k ≤ n, для которого характерно:
//   подпоследовательность индексов i[1] < i[2] < ... < i[k] <= n длины k,
//   для которой каждый элемент A[i[j]] больше любого предыдущего
//   т.е. для всех 1 ≤ j < k, A[i[j]] < A[i[j+1]].
// Решить задачу МЕТОДАМИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

// Sample Input:
// 9
// 1 3 3 2 6
//
// Sample Output:
// 3
  
```

Кафедра экономической информатики. Бгуйр. 2017

19

PISL10. Итоги дин. программирования.

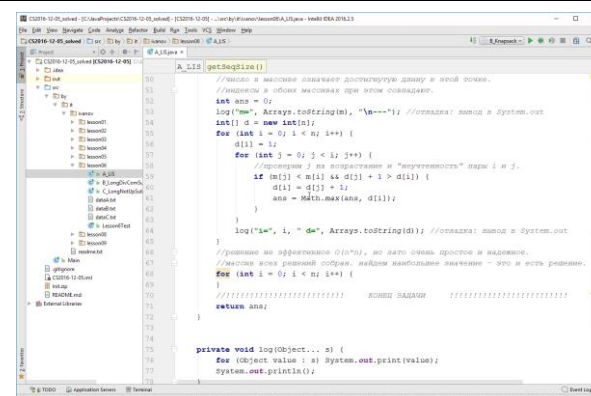
Снизу вверх или сверху вниз

- Преимущества решения подзадач снизу вверх:
 - нет накладных расходов на рекурсию
 - потенциальная возможность сэкономить память
- Преимущества решения подзадач сверху вниз:
 - может оказаться быстрее, если не нужно решать все подзадачи
 - неявный порядок на подзадачах

Кафедра экономической информатики. Бгуйр. 2017

18

PISL10. Итоги дин. программирования.



```

package by.it.ivanov.ivanov04;

import java.util.*;

// Задача на программирование: максимальная подпоследовательность
// см. https://ru.wikipedia.org/wiki/Задача_Максима_Максимовича_Умножения_Максимовича_Подпоследовательности
// https://en.wikipedia.org/wiki/Longest_increasing_subsequence

// Дано:
// - массив чисел 1 ≤ a[i] ≤ 1000
// - массив A[1..n] натуральных чисел, не превосходящих 200.
// Необходимо:
// - вывести максимальное k ≤ n, для которого характерно:
//   подпоследовательность индексов i[1] < i[2] < ... < i[k] <= n длины k,
//   для которой каждый элемент A[i[j]] больше любого предыдущего
//   т.е. для всех 1 ≤ j < k, A[i[j]] < A[i[j+1]].
// Решить задачу МЕТОДАМИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

// Sample Input:
// 9
// 1 3 3 2 6
//
// Sample Output:
// 3
  
```

Кафедра экономической информатики. Бгуйр. 2017

20

PISL10. Итоги дин. программирования.

```

1 package by.it.ivanov.lesson04;
2
3 import java.util.*;
4
5 /**
6  * Задача на программирование: наибольшая кратная подпоследовательность
7  *
8  * Дано:
9  *   число n (1<=n<=1000)
10  *   массив A[1..n] натуральных чисел, не превосходящих 255.
11  *
12  * Необходимо:
13  *   вывести максимальное k<=n, для которого гарантированно найдется
14  *   подпоследовательность индексов i[1]<i[2]<...<i[k] с длины k,
15  *   для которой каждый элемент A[i[k]] делится на предыдущий
16  *   т.е. для всех i<=k, A[i[j]]%A[i[j-1]]==0.
17  *
18  * Решить задачу МЕТОДАМИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ
19  *
20  * Sample Input:
21  *   4
22  *   3 6 7 12
23  *
24  * Sample Output:
25  *   3
26  */
27
28 public class B_LongDivComSubSeq {
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Кафедра экономической информатики. Бгуйр. 2017

21

PISL10. Итоги дин. программирования.

```

1 package by.it.ivanov.lesson04;
2
3 import java.util.*;
4
5 /**
6  * Задача на программирование: наибольшая кратная подпоследовательность
7  *
8  * Дано:
9  *   число n (1<=n<=1000)
10  *   массив A[1..n] натуральных чисел, не превосходящих 255.
11  *
12  * Необходимо:
13  *   вывести максимальное k<=n, для которого гарантированно найдется
14  *   подпоследовательность индексов i[1]<i[2]<...<i[k] с длины k,
15  *   для которой каждый элемент A[i[k]] делится на предыдущий
16  *   т.е. для всех i<=k, A[i[j]]%A[i[j-1]]==0.
17  *
18  * Решить задачу МЕТОДАМИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ
19  *
20  * Sample Input:
21  *   4
22  *   3 6 7 12
23  *
24  * Sample Output:
25  *   3
26  */
27
28 public class B_LongDivComSubSeq {
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Кафедра экономической информатики. Бгуйр. 2017

22

PISL10. Итоги дин. программирования.

```

1 package by.it.ivanov.lesson04;
2
3 import java.util.*;
4
5 /**
6  * Задача на программирование: наибольшая кратная подпоследовательность
7  *
8  * Дано:
9  *   число n (1<=n<=1000)
10  *   массив A[1..n] натуральных чисел, не превосходящих 255.
11  *
12  * Необходимо:
13  *   вывести максимальное k<=n, для которого гарантированно найдется
14  *   подпоследовательность индексов i[1]<i[2]<...<i[k] с длины k,
15  *   для которой каждый элемент A[i[k]] делится на предыдущий
16  *   т.е. для всех i<=k, A[i[j]]%A[i[j-1]]==0.
17  *
18  * Решить задачу МЕТОДАМИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ
19  *
20  * Sample Input:
21  *   4
22  *   3 6 7 12
23  *
24  * Sample Output:
25  *   3
26  */
27
28 public class B_LongDivComSubSeq {
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Кафедра экономической информатики. Бгуйр. 2017

23

PISL10. Итоги дин. программирования.

```

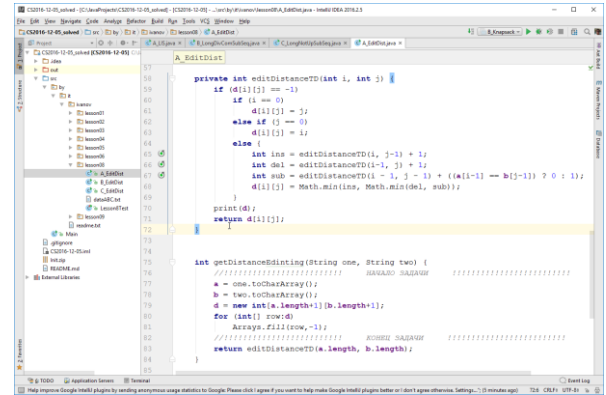
1 package by.it.ivanov.lesson04;
2
3 import java.util.*;
4
5 /**
6  * Задача на программирование: наибольшая кратная подпоследовательность
7  *
8  * Дано:
9  *   число n (1<=n<=1000)
10  *   массив A[1..n] натуральных чисел, не превосходящих 255.
11  *
12  * Необходимо:
13  *   вывести максимальное k<=n, для которого гарантированно найдется
14  *   подпоследовательность индексов i[1]<i[2]<...<i[k] с длины k,
15  *   для которой каждый элемент A[i[k]] делится на предыдущий
16  *   т.е. для всех i<=k, A[i[j]]%A[i[j-1]]==0.
17  *
18  * Решить задачу МЕТОДАМИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ
19  *
20  * Sample Input:
21  *   4
22  *   3 6 7 12
23  *
24  * Sample Output:
25  *   3
26  */
27
28 public class B_LongDivComSubSeq {
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Кафедра экономической информатики. Бгуйр. 2017

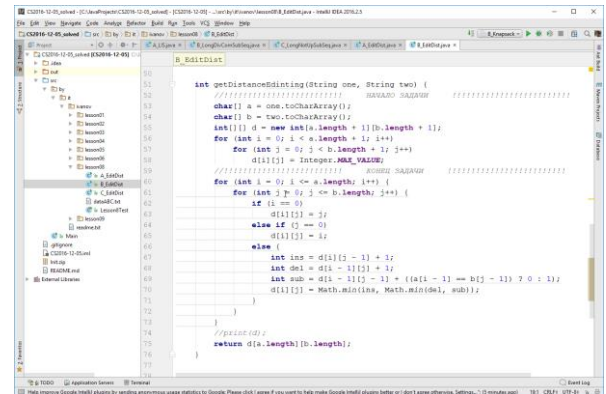
24

PISL10. Итоги дин. программирования.



Кафедра экономической информатики. Бгуир. 2017

PISL10. Итоги дин. программирования.



Кафедра экономической информатики. Бгуир. 2017

PISL10. Итоги дин. программирования.

```

17 public class A_Knapsack {
18     // ...
19     public int getKnapsackWeight(InputStream stream) {
20         // ...
21         Scanner scanner = new Scanner(stream);
22         int W = scanner.nextInt();
23         int n = scanner.nextInt();
24         int gold[] = new int[n];
25         for (int i = 0; i < n; i++) {
26             gold[i] = scanner.nextInt();
27         }
28         // ...
29         int[] d = new int[W+1];
30         for (int w = 1; w <= W; w++) {
31             for (int i = 0; i < n; i++) {
32                 if (gold[i] <= w) {
33                     int w1 = gold[i];
34                     int d1 = d[w-w1];
35                     int added = d[w-w1] + gold[i];
36                     d[w] = Math.max(d[w], added);
37                 }
38             }
39             System.out.println(Arrays.toString(d) + " " + Arrays.toString(gold));
40         }
41         return d[W];
42     }
43 }

```

Кафедра экономической информатики. Бугур. 2017

33

PISL10. Итоги дин. программирования.

```

17 public class A_Knapsack_TD {
18     // ...
19     private Map<Integer, Integer> hash = new HashMap<>();
20     private int[] gold;
21     private int knapsackTD(int w) {
22         if (hash.containsKey(w)) {
23             return hash.get(w);
24         }
25         for (int ci : gold) { // ...
26             if (ci <= w) {
27                 int added = knapsackTD(w-ci) + ci;
28                 int v = Math.max(v, added);
29             }
30         }
31         hash.put(w, v);
32         System.out.println(hash);
33         return hash.get(w);
34     }
35     private int getKnapsackWeight(InputStream stream) {
36         // ...
37         Scanner scanner = new Scanner(stream);
38         int W = scanner.nextInt();
39         int n = scanner.nextInt();
40         gold = new int[n];
41         for (int i = 0; i < n; i++) {
42             gold[i] = scanner.nextInt();
43         }
44     }
45 }

```

Кафедра экономической информатики. Бугур. 2017

34

PISL10. Итоги дин. программирования.

```

43     hash.put(w, v);
44     System.out.println(hash);
45     return hash.get(w);
46 }
47 private int getKnapsackWeight(InputStream stream) {
48     // ...
49     Scanner scanner = new Scanner(stream);
50     int W = scanner.nextInt();
51     int n = scanner.nextInt();
52     gold = new int[n];
53     for (int i = 0; i < n; i++) {
54         gold[i] = scanner.nextInt();
55     }
56     hash.clear();
57     return knapsackTD(W);
58 }
59 public static void main(String[] args) throws FileNotFoundException {
60     String root = System.getProperty("user.dir") + "/src/";
61     InputStream stream = new FileInputStream(root + "by/it/ivanov/lesson09/data1.txt");
62     A_Knapsack_TD instance = new A_Knapsack_TD();
63     int res = instance.getKnapsackWeight(stream);
64     System.out.println(res);
65 }

```

Кафедра экономической информатики. Бугур. 2017

35

PISL10. Итоги дин. программирования.

```

17 public class A_Knapsack {
18     // ...
19     public int getKnapsackWeight(InputStream stream) {
20         // ...
21         Scanner scanner = new Scanner(stream);
22         int W = scanner.nextInt();
23         int n = scanner.nextInt();
24         int gold[] = new int[n];
25         for (int i = 0; i < n; i++) {
26             gold[i] = scanner.nextInt();
27         }
28         // ...
29         int[] d = new int[W+1];
30         for (int w = 1; w <= W; w++) {
31             for (int i = 0; i < n; i++) {
32                 if (gold[i] <= w) {
33                     int w1 = gold[i];
34                     int d1 = d[w-w1];
35                     int added = d[w-w1] + gold[i];
36                     d[w] = Math.max(d[w], added);
37                 }
38             }
39             System.out.println(Arrays.toString(d) + " " + Arrays.toString(gold));
40         }
41         return d[W];
42     }
43 }

```

Кафедра экономической информатики. Бугур. 2017

36

PISL10. Итоги дин. программирования.

```

public class B_Knapsack {
    int getMaxWeight(InputStream stream) {
        Scanner scanner = new Scanner(stream);
        int n = scanner.nextInt();
        int w = scanner.nextInt();
        int gold[] = new int[n];
        for (int i = 0; i < n; i++) {
            gold[i] = scanner.nextInt();
        }
        int[] d = new int[w+1];
        for (int i = 1; i <= n; i++) {
            int wGold[i-1];
            for (int v = 0; v <= w; v++) {
                d[i][v] = d[i-1][v];
                if (v >= gold[i-1]) {
                    int added = d[i-1][v-gold[i-1]] + gold[i-1];
                    d[i][v] = Math.max(d[i][v], added);
                }
            }
        }
        return d[n][w];
    }
}

```

Кафедра экономической информатики. Бугур. 2017

37

PISL10. Итоги дин. программирования.

```

public class C_Stairs {
    int getMaxSum(InputStream stream) {
        Scanner scanner = new Scanner(stream);
        int n = scanner.nextInt();
        int stairs[] = new int[n];
        for (int i = 0; i < n; i++) {
            stairs[i] = scanner.nextInt();
        }
        int[] d = new int[n+1];
        d[0] = 0;
        for (int i = 1; i <= n; i++) {
            d[i] = Math.max(d[i-1], d[i-2] + stairs[i-1]);
        }
        return d[n];
    }
}

```

Кафедра экономической информатики. Бугур. 2017

38

PISL10. Итоги дин. программирования.

```

public class C_Stairs {
    int getMaxSum(InputStream stream) {
        Scanner scanner = new Scanner(stream);
        int n = scanner.nextInt();
        int stairs[] = new int[n];
        for (int i = 0; i < n; i++) {
            stairs[i] = scanner.nextInt();
        }
        int[] d = new int[n+1];
        d[0] = 0;
        for (int i = 1; i <= n; i++) {
            if (i == 1) {
                d[i] = stairs[0];
            } else {
                d[i] = Math.max(d[i-1], d[i-2] + stairs[i-1]);
            }
        }
        return d[n];
    }
}

```

Кафедра экономической информатики. Бугур. 2017

39

PISL10. Сертификация

Алгоритмы: теория и практика. Методы

Поступить на курс

Computer Science Center

О курсе

Алгоритмы – важный раздел теории систем и программирования. Благодаря развитию информационных технологий и алгоритмы мы сегодня можем использовать быстро находить информацию в интернете (в частности, искать по картинкам), находить кратчайший путь, определять, какие из двух функций являются эквивалентными на заданном множестве.

<https://goo.gl/kZAtdE>

Кафедра экономической информатики. Бугур. 2017

40

PISL10. Сертификация

2.3 Наибольший общий делитель

Александр Куликов, из курса Алгоритмы: теория и практика. Методы — 2 Введение: теория и задачи

~27 мин. 4027 1993

Содержание урока

Наибольший общий делитель (слайды)

- Наибольший общий делитель: определение, наивный алгоритм вычисления, важная лемма
- Алгоритм Евклида, анализ времени работы
- Заключение
- Задача на программирование: наибольший общий делитель
- Теоретическая задача для самостоятельной проверки: наибольший общий делитель соседних чисел Фибоначчи

Публикация: Алгоритмы Последнее обновление: 20.12.2016

Комментарии (1)

Новые обсуждения

Кафедра экономической информатики. Бгуир. 2017

41

PISL10. Сертификация

Александр Куликов, из курса Алгоритмы: теория и практика. Методы — 2 Введение: теория и задачи

~27 мин. 4027 1993

Онлайн-лекция: Наибольший общий делитель

Публикация: Алгоритмы Последнее обновление: 20.12.2016

Кафедра экономической информатики. Бгуир. 2017

42

PISL10. Сертификация

Задача на программирование: наибольший общий делитель

По данным двум числам $1 \leq a, b \leq 2 \cdot 10^9$ найдите их наибольший общий делитель.

Sample Input 1:
18 35

Sample Output 1:
1

Sample Input 2:
14159572 63967872

Sample Output 2:
4

Нажмите, чтобы начать решать

Решено: 3076 Успешных решений: 41% Вы получите: 1 балл +3 знаний

Кафедра экономической информатики. Бгуир. 2017

43

PISL10. Сертификация

Задача на программирование

Sample Input 2:
4

Time Limit: 2 seconds
Memory Limit: 256 MB

Выбор языка программирования:

- Java 8
- ASM32
- ASM64
- C#
- C++
- C++11
- Clojure
- Haskell (ghc 7.10)
- Haskell (ghc 7.8)
- Java 7
- Java 8
- JavaScript
- Octave
- Python 2
- R
- Rust
- Scala
- Shell

Upload file

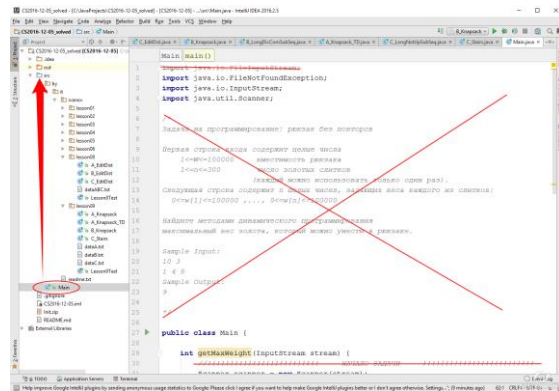
Очистить

Решено: 3076 Успешных решений: 41% Вы получите: 1 балл +3 знаний

Кафедра экономической информатики. Бгуир. 2017

44

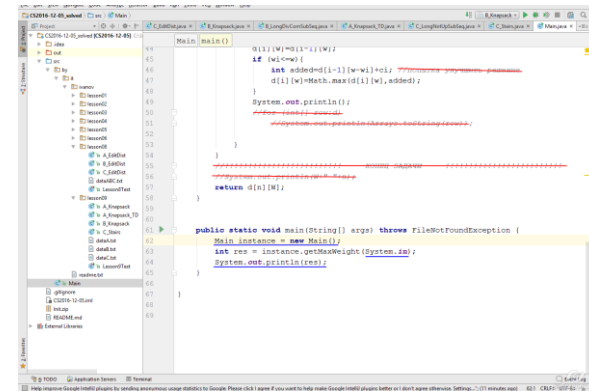
PISL10. Сертификация



Кафедра экономической информатики. Бгуйр. 2017

45

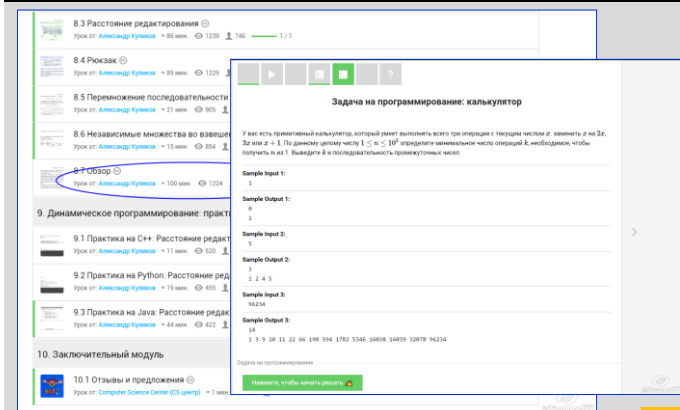
PISL10. Сертификация



Кафедра экономической информатики. Бгуйр. 2017

46

Задание А В С.



Кафедра экономической информатики. Бгуйр. 2017

47