

# Алгоритмы и структуры данных

специальность: 1-31 03 04 «Информатика»



## **Соболевская Елена Павловна**

доцент кафедры дискретной математики и алгоритмики,  
кандидат физико-математических наук, доцент

Лауреат премии имени А.Н. Севченко в номинации «Образование»  
за цикл пособий по дискретной математике, проектированию и  
анализу алгоритмов

<http://fpmi.bsu.by/main.aspx?guid=30051>



### **Буславский Александр Андреевич**

старший преподаватель кафедры дискретной математики и алгоритмики,  
Лауреат специального фонда Президента Республики Беларусь по социальной поддержке одарённых учащихся и студентов.

<http://fpmi.bsu.by/main.aspx?guid=39321>



### **Лукьянов Иван Денисович**

старший преподаватель кафедры дискретной математики и алгоритмики,  
Лауреат специального фонда Президента Республики Беларусь по социальной поддержке одарённых учащихся и студентов, принимал участие в финале ICPC.

<http://fpmi.bsu.by/main.aspx?guid=47071>

## Информационно-коммуникационные технологии:

1. Образовательный портал БГУ <https://edufpmi.bsu.by>
2. Образовательная платформа Insight Runner <https://acm.bsu.by/>
3. Группы в мессенджере Telegram, сервисы Google.



## Соболь Сергей Александрович

инженер-программист ООО ЯндексБел,  
старший преподаватель кафедры дискретной математики и  
алгоритмики (2014-2020 год),  
магистр математики и информационных технологий,  
серебряная медаль ACM ICPC 2013.



<https://acm.bsu.by/>

ЛОГИН номер вашего студенческого (семь цифр) или S22230716  
пароль (по умолчанию): 11111

год группа № в списке на момент регистрации

↓ ↓ ↙

После первого входа в iRunner необходимо сменить пароль и установить двухфакторную аутентификацию (для исключения противоправных действий в системе при несанкционированном доступе.)

## Двухфакторная аутентификация

✓ 2ФА включена

Токены будут генерироваться мобильным приложением.

### Резервные токены

Если у вас нет с собой мобильного устройства, вы можете получить доступ к учётной записи посредством резервных токенов. У вас осталось 0 токенов.

[Показать токены](#)

### Отключить двухфакторную аутентификацию

Если вы переходите на новое мобильное устройство или выполняете общий сброс, временно отключите двухфакторную аутентификацию, затем включите её снова.

[Отключить 2ФА](#)

# Insight Runner Wiki

## руководство по работе с системой



БЕЛОРУССКИЙ  
ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ

iRUNNER WIKI

Искать в iRunner Wiki

iR2

### Навигация

Система iRunner 2  
Система iRunner  
(старая)  
Заглавная страница  
Свежие правки  
Случайная статья  
Справка

### Вики-инструменты

Загрузить файл  
Служебные страницы

## Заглавная страница

[Заглавная](#) [Обсуждение](#) [★](#)

### Содержание

[скрыть](#)

- 1 Система iRunner
- 2 Алгоритмы и структуры данных / Теория алгоритмов
  - 2.1 Образовательный портал БГУ
  - 2.2 Учебные программы
  - 2.3 Задачи iRunner
  - 2.4 Теоретический материал (учебные пособия, статьи)
- 3 Разное
- 4 Операционные системы семейства UNIX (магистратура)
- 5 Эксплуатация и администрирование UNIX-систем (магистратура)
- 6 Внутреннее устройство ОС семейства UNIX (магистратура)
- 7 Системное программирование на языке C (магистратура)
- 8 Дискретная математика и математическая логика (1–2-й курс)

## Система iRunner [править](#)

- Руководство по работе с системой iRunner.

## Алгоритмы и структуры данных / Теория алгоритмов

### Образовательный портал БГУ [править](#)

- [Ссылка на курс АиСД \(ПИ, ПМ\) на портале БГУ](#)
- [Ссылка на курс ТА \(ИНФ\) на портале БГУ](#)

### Учебные программы [править](#)

- Теория алгоритмов. Учебная программа (Информатика)

Работа в системе — iRunner Wiki

Искать в iRunner Wiki

## Работа в системе

[Статья](#) [Обсуждение](#) [★](#)

### Содержание

[скрыть](#)

- 1 Общий принцип
- 2 Ввод и вывод данных
- 3 Тестирование решений
  - 3.1 CE — Ошибка компиляции (Compilation Error)
  - 3.2 TLE — Нарушен предел времени (Time Limit Exceeded)
  - 3.3 ILE — Нарушен предел ожидания (Idleness Limit Exceeded)
  - 3.4 MLE — Нарушен предел памяти (Memory Limit Exceeded)
  - 3.5 RTE — Ошибка во время выполнения (Run-time Error)
  - 3.6 PE — Ошибка представления (Presentation Error)
  - 3.7 WA — Неправильный ответ (Wrong Answer)
  - 3.8 OK — Принято (Accepted)
  - 3.9 CF — Ошибка тестирования (Check Failed)
  - 3.10 SV — Нарушение безопасности (Security Violation)
- 4 Особенности языков программирования
  - 4.1 Выбор языка программирования
- 5 Конфигурация тестирующего сервера
- 6 Языки программирования

### Общий принцип [править](#)

В систему посылаются только файлы с исходным кодом, а сама посылаемая программа должна состоять только из одного файла: `*.dpr`, `*.cpp`, `*.c`.  
Нельзя отправить в систему скомпилированный exe-файл, файл проекта Visual Studio и т. п.

В решениях запрещается:

- осуществлять доступ к сети;
- выполнять любые операции ввода/вывода, кроме открывания, закрывания, чтения и записи стандартных потоков `stdin`, `stdout`, `stderr` и файлов, прописанными в условии задачи;
- сознательно «ломать» тестирующую систему;
- выполнять другие программы и порождать новые процессы;
- изменять права доступа к файловой системе;
- работать с поддиректориями;
- создавать и манипулировать ресурсами GUI (окна, диалоговые сообщения и т. д.);
- работать со внешними устройствами (звук, принтер и т. д.);
- выполнять прочие действия, призванные нарушить ход учебного процесса.

Решения выполняются в специальном окружении, обеспечивающем безопасный запуск, и попытка выполнить какие-либо из указанных действий: всего, получением вердикта «Ошибка во время выполнения».

### Ввод и вывод данных [править](#)



acm.bsu.by

iRunner Wiki

Искать в iRunner Wiki

## Система iRunner [✎ править](#)

- Руководство по работе с системой iRunner.

## Алгоритмы и структуры данных / Теория алгоритмов [✎ править](#)

### Образовательный портал БГУ [✎ править](#)

- [Ссылка на курс АиСД \(ПИ, ПМ\) на портале БГУ](#)
- [Ссылка на курс ТА \(ИНФ\) на портале БГУ](#)

### Учебные программы [✎ править](#)

- Теория алгоритмов. Учебная программа ([Информатика](#))
- Алгоритмы и структуры данных. Типовая учебная программа ([ПИ, ПМ, ЭК, АМ, КБ](#))
- Алгоритмы и структуры данных. Учебная программа ([Прикладная информатика](#))
- Алгоритмы и структуры данных. Учебная программа ([Прикладная математика](#))
- Алгоритмы и структуры данных. Учебная программа ([Экономическая кибернетика, Актуарная математика, Компьютерная безопасность](#))

### Задачи iRunner [✎ править](#)

- Описание тем задач.
- Система оценок по ТА (подгруппа Иржавского).

### Теоретический материал (учебные пособия, статьи) [✎ править](#)

- Учебное пособие «Алгоритмы и структуры данных» (В. М. Котов, Е. П. Соболевская, А. А. Толстикова) [↗](#).
- Учебное пособие «Теория алгоритмов» (П. А. Иржавский, В. М. Котов, А. Ю. Лобанов, Ю. Л. Орлович, Е. П. Соболевская) [↗](#).
- [Материалы по структурам данных \(И. С. Метельский\)](#).
- Сборник задач по теории алгоритмов. Бинарные поисковые деревья. Алгоритмы на графах. 2017 год.» (В. М. Котов, Ю. Л. Орлович, Е. П. Соболевская, С. А. Соболев).
- Сборник задач по теории алгоритмов. Структуры данных. 2020 год.» (С. А. Соболев, К. Ю. Вильчевский, В. М. Котов, Е. П. Соболевская).
- Бинарные поисковые деревья.
- Программная реализация бинарных поисковых деревьев.
- Терминология теории графов.
- Максимальное и наибольшее.





# Insight Runner

## тесты для самоконтроля

Для самоконтроля усвоения теоретического материала в [Insight Runner](#) разработана **система тестов**.

Тесты разработаны для большинства разделов учебной дисциплины. Ответы к тестам (на усмотрение преподавателя) могут быть открыты после прохождения теста всеми учащимися.

Так как тесты в [iRunner](#) генерируются автоматически, то это позволяет бороться со списыванием.

**Итоговый тест** включает в себя **20** тестовых вопросов (**30** минут) и его результат учитывается в рейтинговой оценке за работу в семестре.

# Insight Runner

## тесты для самоконтроля

### АиСД. Самоконтроль. Сбалансированные деревья.

Удалить

Верно  
Неверно, но выбрано  
Верно, но не выбрано

Студент: Илья Айана  
Оценка: 5  
Баллы: 5,0 из 10,0  
Время начала: 14 ноября 2022 г. 10:38  
Длительность: 0:20:00

#### Вопрос 1

0,0 из 5,0

Задан массив  $H = (2, 4, 4, 0, 1)$ . Будем считать, что индексы начинаются с единицы.

Строится помеченное корневое дерево  $T$  из пяти вершин (вершины дерева занумерованы целыми числами от 1 до 5, ключ вершины совпадает с её номером). Дерево строится таким образом, что массив  $H$  является каноническим представлением дерева  $T$  ( $H[i]$  — отец вершины  $i$ ; если  $H[i] = 0$ , то вершина  $i$  является корнем дерева).

Определить, какие из следующих утверждений являются верными.

Можно восстановить дерево  $T$  так, чтобы оно было АВЛ-деревом.

Можно восстановить дерево  $T$  так, чтобы оно было 3-сбалансированным (по высоте).

Можно восстановить дерево  $T$  так, чтобы оно было идеально сбалансированным (по вершинам).

Можно восстановить дерево  $T$  так, чтобы оно было полным бинарным.

Можно восстановить дерево  $T$  так, чтобы оно было бинарным поисковым.

Нет верных утверждений.

#### Вопрос 2

5,0 из 5,0

Если для последовательности элементов  $(2, 5, 4, 3, 1)$  построить АВЛ-дерево последовательным добавлением нового элемента с поддержкой инварианта сбалансированности после каждого добавления, то какая вершина будет являться корнем дерева и какие повороты (RR, LL, LR, RL) будут выполнены при этом? Если  $x$  — некоторая вершина дерева, то в её левом поддереве ключи вершин меньше, чем ключ вершины  $x$ , а в правом — больше (ключ вершины совпадает с её номером).

1

2

3

4

5

повороты не выполнялись

RR

LL

LR

Для закрепления на практике теоретических знаний в [Insight Runner](#) разработаны **общие задачи** (их должны выполнить все студенты).

Общие задачи открываются в iRunner, как правило, после каждой лекции и нацелены на проработку базовых знаний по пройденному на лекции материалу.

Эти задачи достаточно простые, не предполагают разработки сложных алгоритмов решения, а готовят студентов к решению индивидуальных задач.

Преподаватель может установить крайний срок выполнения задания. Задания, выполненные с нарушением этого срока, в системе iRunner имеют особые пометки.

# Insight Runner

## общие задачи

### **Тема 1. Рекуррентные соотношения (динамическое программирование)**

Кувшинки

Единицы 1

Единицы 2 (модульная арифметика)

Оптимальное перемножение группы матриц

Наибольшая общая подпоследовательность двух строк (LCS)

Наибольшая подпоследовательность-палиндром

Наибольшая возрастающая подпоследовательность (LIS)

Преобразование строк (взвешенное расстояние Левенштейна)

### **Тема 2. Структуры данных для организации поиска:**

Бинарный поиск (BinarySearch, LowerBound, UpperBound)

Построение бинарного поискового дерева

Удаление вершин из бинарного поискового дерева

Проверка является ли бинарное дерево поисковым

### **Тема 3. Специальные структуры данных**

Бинарная куча

Кодирование Хаффмана

Биномиальная куча

Задача о сумме (реализация структур данных для выполнения интервальных запросов)

Дерево отрезков

Хеш-таблица (разрешение коллизий метом открытой адресации)

# Insight Runner

## (общие задачи)

### Тема 4. Строковые алгоритмы и структуры данных:

Циклический сдвиг

Суффиксный массив

### Тема 5. Алгоритмы на графах

Матрица смежности

Канонический вид (по списку дуг)

Список смежности

Канонический вид (по матрице смежности)

Строительство дорог

Разрушение дорог

Разрушение дорог (большие ограничения)

BFS (поиск в ширину)

DFS (поиск в глубину)

Кратчайший путь. Алгоритм Дейкстры

Максимальный поток в сети (простая версия)

Максимальный поток в сети (большие ограничения, только для желающих)

# Insight Runner

## ИНДИВИДУАЛЬНЫЕ ЗАДАЧИ

В рамках учебной дисциплины студентами также выполняются **индивидуальные задачи**.

Число индивидуальных задач - по каждой из 5 тем курса не менее одной.

Индивидуальная задача предполагает разработку эффективного алгоритма решения задачи с последующей реализацией его в [Insight Runner](#) на любом языке программирования.

С 2016 г. в учебных курсах **ограничивается до пяти** число решений, которые можно отправить по каждой из назначенных задач в течение суток.

На протяжении любого 24-часового отрезка времени разрешается отправить не более чем 5 решений по каждой задаче. Ограничение не привязано к наступлению новых суток в 00:00. Решения с вердиктом «*Ошибка компиляции*», а также решения, в которых не пройдены тесты из условия задачи (остальные тесты не проверяются), при подсчёте оставшихся попыток игнорируются.

Отметим, что в течение многих лет для всех студентов, которые работали в старой версии системы [Insight Runner](#), действовало ограничение в 20 попыток по задаче в семестр. Это ограничение нельзя было увеличить индивидуально. Политика ограничения числа посылок в день вместо ограничения общего числа посылок является более гибкой и применяется во многих системах, например на платформе Kaggle.

**Призываем вас более качественно тестировать свои решения перед отправкой!!!**

# Insight Runner

## (индивидуальные задачи)

2-й курс 4-я группа ТА 2019–2020

🔒 В архиве

📅 Задачи по курсу

✉ Сообщения

📌 Назначение задач

**📖 Журнал**

📄 Ведомость

📋 Все решения

📋 Мои решения

🚀 Компиляторы

❓ Тесты

🔄 Электронная очередь

⚙ Настройки

↔ Широкоэкранный режим

### Индивидуальные задачи

№	Студент	Деревья поиска	Рекуррентные соотношения		Структуры данных		Алгоритмы на графах		Перебор	Приближенные алгоритмы	Задачи	Итог
1		10 A	10 A	10 A	10 A	10 A	9 A	10 A	10 A	10 A	9 (3/3)	10
2		9 A	9 A	9 A	10 A	10 A	10 A	10 A	8 A	8 A	8 (1/3)	8
3		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
4		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
5		9 A	9 A	9 A	10 A	10 A	10 A	10 A	10 A	10 A	8 (3/3)	8
6		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
7		9 A	9 A	9 A	10 A	10 A	8 A	10 A	10 A	10 A	9 (3/3)	10
8		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
9		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (2/4)	10
10		10 A	10 A	10 A	10 A	10 A	10 A	10 A			7 (0/3)	7
11		10 A	10 A	10 A	10 A	10 A	10 A				4 (1/3)	5
12		10 A	10 A	10 A	10 A	10 A	10 A	9 A	10 A	10 A	9 (3/3)	10
13		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
14		10 A	10 A	10 A	10 A	10 A	9 A	10 A	10 A 27/30	10 A	8 (3/3)	8
15		9 A	10 A	10 A	9 A	9 A	10 A	10 A	10 A	10 A	9 (3/3)	10
16		9 A	10 A	10 A	10 A	10 A 2/43	10 A 14/29	10 A 53/100			3 (2/3)	3
17		10 A	9 A	10 A	10 A	10 A	9 A	9 A	10 A	10 A	9 (3/3)	10
18		7 A	9 A	9 A	7 A	8 A	9 A	9 A	8 A 13/14	8 A 22/25	7 (3/3)	6
19		10 A	10 A	10 A	10 A	10 A	10 A	10 A 31/33	10 A	10 A	7 (3/4)	8
20		9 A	9 A	9 A	9 A	9 A	9 A	10 A	10 A	10 A	9 (3/3)	10
21		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
22		10 A	9 A	9 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
23		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
24		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
25		10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A 31/33	8 (3/3)	8
26		10 A	10 A	9 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (3/3)	10
27		10 A	10 A	10 A	10 A 22/23	10 A	10 A	10 A	10 A	10 A	8 (3/3)	10
28		9 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	10 A	9 (2/4)	10

# Insight Runner

## проверка на плагиат

acm.bsu.by

Все решения — 2-й курс 4-я группа ТА 2019–2020 — iRunner 2

Елена Соболевская Выход 12:05:10

2-й курс 4-я группа ТА 2019–2020

В архиве

Задачи по курсу

Сообщения

Назначение задач

Журнал

Ведомость

Все решения

Мои решения

Компиляторы

Тесты

Электронная очередь

Настройки

Страница 1: 1–25 из 997

7 12 25 50 100 Все

Автор	Задача	Время	Файл	Статус	Плагат
	10 21. Детали — станки	29 мая 2020 г. 23:20	C++	OK	25 0,00
	10 23. Частичный порядок обработки деталей	28 мая 2020 г. 0:06	C++	OK	11 0,00
	10 25. Станки — детали	26 мая 2020 г. 23:06	C++	OK	28 0,00
	7 15. Маршрут робота	25 мая 2020 г. 21:15	C++	OK	10 0,00
	8 32. Равенства и неравенства	25 мая 2020 г. 16:03	C++	OK	39 0,00
	10 23. Упаковка семи деталей	25 мая 2020 г. 15:38	C++	OK	10 0,00
	10 26. Нераспределённая задача с сервером	25 мая 2020 г. 10:26	C++	OK	14 0,32
	10 15. Детали с суммарным временем обработки	25 мая 2020 г. 10:17	C++	OK	36 0,00
	10 31. n работ, 1 исполнитель	25 мая 2020 г. 1:44	C++	OK	12 0,00
	10 31. n работ, 1 исполнитель	25 мая 2020 г. 1:40	C++	OK	12 0,00
	0.10. Алгоритм Дейкстры	25 мая 2020 г. 0:11	C++	OK	47 0,56
	8 10. Минимальная необходимая загруженность машины	24 мая 2020 г. 21:50	C++	OK	11 0,33
	0.7. Канонический вид (по матрице смежности)	24 мая 2020 г. 21:25	C++	OK	14 0,98
	0.5. Канонический вид (по списку дуг)	24 мая 2020 г. 21:15	C++	OK	10 1,00
	10 40. Искусство	24 мая 2020 г. 21:15	C++	OK	10 0,85
	0.6. Построить список смежности	24 мая 2020 г. 21:11	C++	OK	15 1,00
	0.4. Построить матрицу смежности	24 мая 2020 г. 21:07	C++	OK	20 1,00
	10 43. Машина времени	24 мая 2020 г. 20:33	C++	OK	33 0,50
	10 30. Пятнашки $n \times m$	24 мая 2020 г. 19:51	C++	OK	12 0,00
	0.9. Поиск в глубину	23 мая 2020 г. 3:20	C++	OK	27 0,94
	0.8. Поиск в ширину	23 мая 2020 г. 3:08	C++	OK	27 0,96
	0.3. Является ли бинарное дерево поисковым?	23 мая 2020 г. 1:32	C++	OK	55 0,13
	10 19. Обработка в две стадии	22 мая 2020 г. 16:32	C++	OK	16 0,00
	10 19. Обработка в две стадии	22 мая 2020 г. 15:52	C++	OK	16 0,00
	10 29. Мультиразрез	22 мая 2020 г. 14:25	C++	OK	22 0,00



# Insight Runner

## проверка на плагиат

```
19
N 20 Scanner in = new Scanner(new File("input.txt"));
21 PrintWriter out = new PrintWriter("output.txt");
22 int n = in.nextInt();
23
24 if(n==3){
N 25     out.println(0);
26     out.flush();
27
28 }
N 29
30 else {
N 31     while (in.hasNext()) {
32         points.add(new Point(in.nextInt(), in.nextInt()));
33     }
34     if(n==4){
N 35         out.format("%.2f%n",
Math.min(points.get(0).distance(points.get(2)),points.get(1).distance(points.get(3))));
36         out.flush();
37     }else {
N 38
39         double[][] m = new double[n + 1][n];
40
41         n--;
42         for (int i = 1; i <= n; i++)
N 43             m[i][i] = 0;
44
45         for (int p = 2; p <= n; p++) {
46             for (int i = 1; i <= n - p + 1; i++) {
47                 int j = i + p - 1;
N 48                 m[i][j] = Integer.MAX_VALUE;
49
50                 for (int k = i; k <= j - 1; k++) {
N 51                     double d = calcDiagonals(i - 1, k, j, points);
52                     double dMax = Math.max(Math.max(m[i][k], m[k + 1][j]), d);
53
54                     if (dMax < m[i][j]) {
```

```
19
N 20
21
22     int n = fin.nextInt();
23
24     if(n==3){
N 25         fout.println(0);
26         fout.flush();
27
28     }
N 29
30     else {
N 31         while (fin.hasNext()) {
32             points.add(new Point(fin.nextInt(), fin.nextInt()));
33         }
34         if(n==4){
N 35             fout.format("%.2f%n",
Math.min(points.get(0).distance(points.get(2)),points.get(1).distance(points.get(3))));
36             fout.flush();
37         }else {
N 38
39             double[][] memor = new double[n + 1][n];
40
41             n--;
42             for (int i = 1; i <= n; i++)
N 43                 memor[i][i] = 0;
44
45             for (int p = 2; p <= n; p++) {
46                 for (int i = 1; i <= n - p + 1; i++) {
47                     int j = i + p - 1;
N 48                     memor[i][j] = Integer.MAX_VALUE;
49
50                     for (int k = i; k <= j - 1; k++) {
N 51                         double d = countDiagonals(i - 1, k, j, points);
52                         double dMax = Math.max(Math.max(memor[i][k], memor[k +
1][j]), d);
48
53                         if (dMax <= memor[i][j]) {
```

# ДОВЕРЯЙ, НО ПРОВЕРЯЙ ...

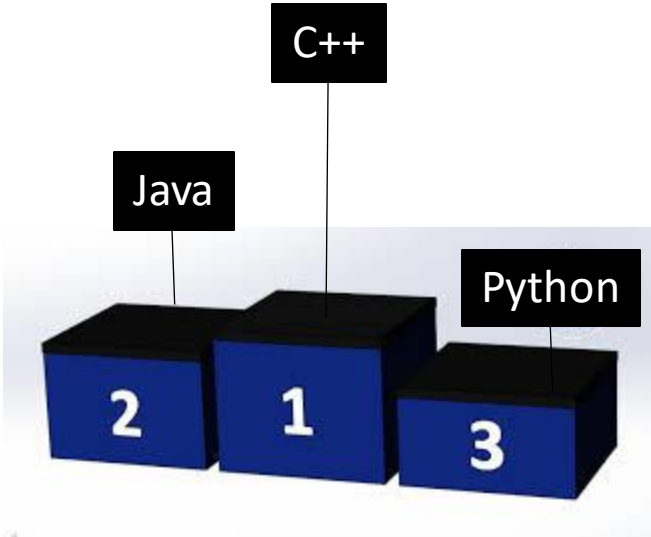
10	68. Ожерелье	4 декабря 2022 г. 14:25	C++	OK	58	1,00
8	29.2. Слова	4 декабря 2022 г. 14:16	Java	OK	35	1,00
8	29.2. Слова	4 декабря 2022 г. 14:14	Java	OK	35	1,00
8	29.2. Слова	4 декабря 2022 г. 14:08	Java	OK	35	1,00
7	51. Достроить дороги	2 декабря 2022 г. 19:18	Java	OK	33	1,00
	0.2. Бинарная куча?	2 декабря 2022 г. 19:11	C++	OK	24	1,00
	0.4. Задача о сумме	2 декабря 2022 г. 19:10	C++	OK	24	1,00
	69.2. Путь лягушки	1 декабря 2022 г. 19:20	C++	OK	22	1,00
	0.2. Бинарная куча?	30 ноября 2022 г. 20:31	C++	OK	24	1,00
	20. Палиндром	30 ноября 2022 г. 13:38	C++	OK	25	1,00
6	8. Разбиение на группы незнакомых	30 ноября 2022 г. 13:37	C++	OK	15	1,00
	0.2. Единицы (часть 1).	30 ноября 2022 г. 13:36	C++	OK	64	1,00
8	36. Звёздочки	30 ноября 2022 г. 11:59	C++	OK	72	1,00
	20. Палиндром	28 ноября 2022 г. 18:29	C++	OK	25	1,00
	0.5. Дерево отрезков	28 ноября 2022 г. 18:23	C++	OK	7	1,00
	0.2. Единицы (часть 1).	27 ноября 2022 г. 15:30	C++	OK	64	1,00
	0.3. Биномиальная куча	27 ноября 2022 г. 15:29	C++	OK	20	1,00
	0.2. Бинарная куча?	27 ноября 2022 г. 15:28	C++	OK	24	1,00
	0.1. Бинарный поиск	27 ноября 2022 г. 15:26	C++	OK	17	1,00
10	57. Снести	26 ноября 2022 г. 18:17	C++	OK	29	1,00
	25. Преобразование строк	25 ноября 2022 г. 14:54	Java	OK	13	1,00
	20. Палиндром	25 ноября 2022 г. 13:46	Java	OK	25	1,00
5	24. Блок из единиц	25 ноября 2022 г. 13:38	C++	OK	19	1,00
	0.2. Бинарная куча?	25 ноября 2022 г. 8:42	C++	OK	24	1,00
4	69.1. Кувшинки	25 ноября 2022 г. 8:39	C++	OK	18	1,00
4	69.1. Кувшинки	25 ноября 2022 г. 8:37	C++	OK	18	1,00
	6. Строго возрастающая без разрывов подпоследовательность	25 ноября 2022 г. 8:35	C++	OK	29	1,00
	6. Строго возрастающая без разрывов подпоследовательность	25 ноября 2022 г. 8:33	C++	OK	29	1,00
	25. Преобразование строк	25 ноября 2022 г. 8:31	C++	OK	13	1,00
	20. Палиндром	25 ноября 2022 г. 8:30	C++	OK	25	1,00
	0.2. Единицы (часть 1).	24 ноября 2022 г. 14:53	C++	OK	64	1,00
	0.3. Единицы (часть 2).	24 ноября 2022 г. 14:52	C++	OK	104	1,00

Отправлено с 2003г. в iRunner:

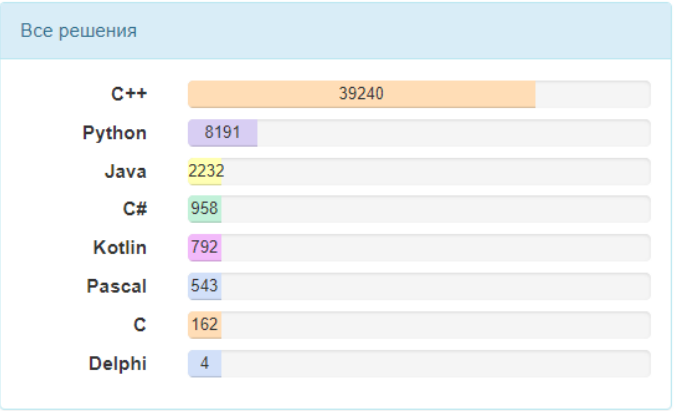
786 246 решений,

из которых **принято**  
174 090 (22%).

Число принятых решений на 12.01.2024	
C++	135 718
Java	13 848
Pascal	2 676
Delphi	2 621
Python	2 621
C#	2 154
C	1 997
Kotlin	354



Данные с 1.09.2023-12.01.2024 г.



Вердикты		
WA	Неправильный ответ	21486
OK	Принято	9165
RTE	Ошибка во время выполнения	6588
TLE	Нарушен предел времени	5820
CE	Ошибка компиляции	5594
PE	Ошибка представления	2497
MLE	Нарушен предел памяти	963
ILE	Нарушен предел ожидания	4
SV	Нарушение безопасности	4
CF	Ошибка тестирования	1

# Литература



1. Котов, В. М. Алгоритмы и структуры данных : учеб. пособие / В. М. Котов, Е. П. Соболевская, А. А. Толстиков. – Минск : БГУ, 2011. – 267 с. – (Классическое университетское издание). [Электронный ресурс]. – Режим доступа: <https://elib.bsu.by/handle/123456789/8522>
2. Теория алгоритмов : учеб. пособие / П. А. Иржавский [и др.] – Минск : БГУ, 2013. – 159 с. [Электронный ресурс]. – Режим доступа: <https://elib.bsu.by/handle/123456789/91612>
3. Сборник задач по теории алгоритмов : учеб.-метод. пособие / В. М. Котов [и др.] – Минск : БГУ, 2017. – 183 с. [Электронный ресурс]. – Режим доступа: <https://elib.bsu.by/handle/123456789/181529>
4. Сборник задач по теории алгоритмов. Структуры данных : учеб.-метод. пособие / С. А. Соболев [и др.] – Минск : БГУ, 2020. – 159 с. [Электронный ресурс]. – Режим доступа: <https://elib.bsu.by/handle/123456789/255033>
5. Котов, В. М. Сборник задач по теории алгоритмов. Организация перебора и приближенные алгоритмы : электронный учебно-методический комплекс для специальности 1-31 03 04 «Информатика» / В.М.Котов, Е.П. Соболевская, Г.П. Волчкова; БГУ, Фак. прикладной математики и информатики, Каф. дискретной математики и алгоритмики. – Минск : БГУ, 2021. – 144 с. : ил. – Библиогр.: с. 143–144. [Электронный ресурс]. – Режим доступа: <https://elib.bsu.by/handle/123456789/272717>

6. [Алгоритмы: построение и анализ / Т. Кормен \[и др.\] – М.: Вильямс, 2005. – 1296 с.](#)



БЕЛОРУССКИЙ  
ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ

Спасибо за внимание!