

Независимая  
научно-практическая конференция  
«Разработка ПО 2011»

31 октября - 3 ноября, Москва



Программирование олимпиадными командами как  
разновидность экстремального программирования  
для решения задач дискретной математики (на  
примере сборки генома)

**Царев Ф. Н., Шалыто А. А.**

Санкт-Петербургский национальный исследовательский  
университет информационных технологий, механики и оптики

# Решаемые в работе задачи

- «Зачем нужны чемпионы мира по программированию?» - какие задачи надо ставить в университете, чтобы участникам и призерам чемпионата мира по программированию было интересно работать
- Как решать научные задачи в области разработки программного обеспечения на основе алгоритмов дискретной математики

# Чемпионат мира по программированию среди студентов (1)

- Организуется ACM (Association for Computing Machinery), спонсируется IBM
- Наиболее престижное соревнование для молодых программистов
- Проводится с 1979 года
- Чемпионат мира – с начала 90-х годов

# Чемпионат мира по программированию среди студентов (2)

- Более 7000 команд из 2000 университетов 88 стран мира
- Медалями награждаются 12 команд – 4 золотых, 4 серебряных, 4 бронзовых медали
- Многоуровневое соревнование
  - Четвертьфиналы – 260 в мире
  - Полуфиналы – 30 в мире
  - Финал (100 команд)

# Чемпионат мира по программированию среди студентов (3)

- Каждый этап длится 5 часов
- Команда состоит из трех студентов
- Дается 10-12 задач
- Команде предоставляется один компьютер
- Чтобы решать задачи, необходимо:
  - Знать алгоритмы и структуры данных
  - Быстро и безошибочно их реализовывать



## Problem A A Careful Approach Input: approach.in

If you think participating in a programming contest is stressful, imagine being an air traffic controller. With human lives at stake, an air traffic controller has to focus on tasks while working under constantly changing conditions as well as dealing with unforeseen events.

Consider the task of scheduling the airplanes that are landing at an airport. Incoming airplanes report their positions, directions, and speeds, and then the controller has to devise a landing schedule that brings all airplanes safely to the ground. Generally, the more time there is between successive landings, the "safer" a landing schedule is. This extra time gives pilots the opportunity to react to changing weather and other surprises.

Luckily, part of this scheduling task can be automated – this is where you come in. You will be given scenarios of airplane landings. Each airplane has a time window during which it can safely land. You must compute an order for landing all airplanes that respects these time windows. Furthermore, the airplane landings should be stretched out as much as possible so that the minimum time gap between successive landings is as large as possible. For example, if three airplanes land at 10:00am, 10:05am, and 10:15am, then the smallest gap is five minutes, which occurs between the first two airplanes. Not all gaps have to be the same, but the smallest gap should be as large as possible.

### Input

The input file contains several test cases consisting of descriptions of landing scenarios. Each test case starts with a line containing a single integer  $n$  ( $2 \leq n \leq 8$ ), which is the number of airplanes in the scenario. This is followed by  $n$  lines, each containing two integers  $a_i, b_i$ , which give the beginning and end of the closed interval  $[a_i, b_i]$  during which the  $i^{\text{th}}$  plane can land safely. The numbers  $a_i$  and  $b_i$  are specified in minutes and satisfy  $0 \leq a_i \leq b_i \leq 1440$ .

The input is terminated with a line containing the single integer zero.

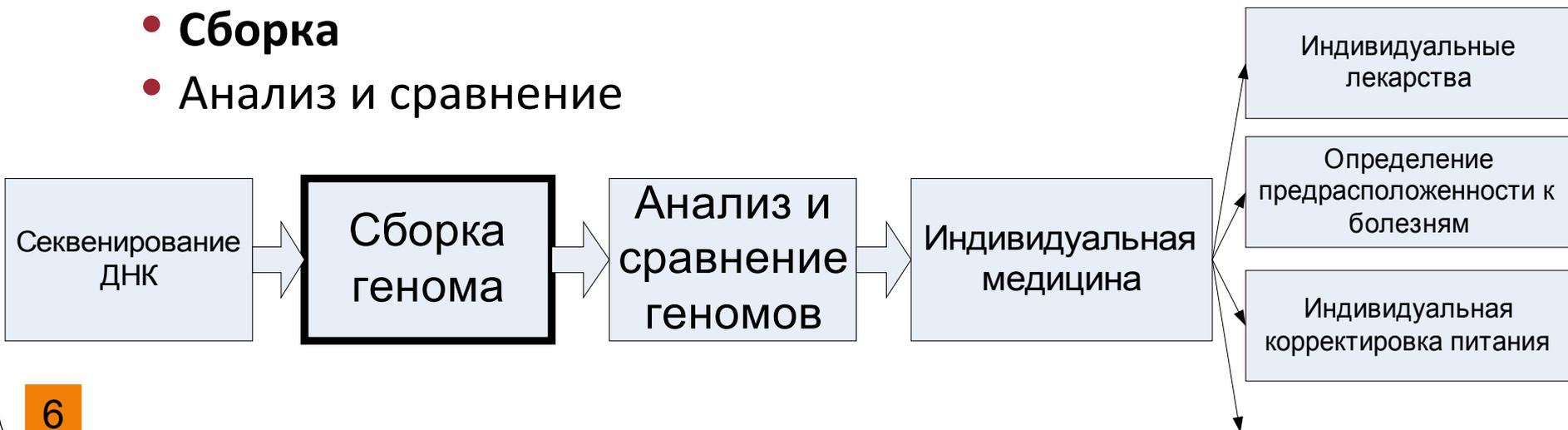
### Output

For each test case in the input, print its case number (starting with 1) followed by the minimum achievable time gap between successive landings. Print the time split into minutes and seconds, rounded to the closest second. Follow the format of the sample output.

Sample Input	Output for the Sample Input
3 0 10 5 15 10 15 2 0 10 10 20 0	Case 1: 7:30 Case 2: 20:00

# Секвенирование, сборка и анализ генома

- **Наша цель** – разработка технологии сборки генома, превосходящей по качеству или производительности мировой уровень
  - Алгоритмы
  - Программное обеспечение
- Задачи в области генома
  - Секвенирование
  - **Сборка**
  - Анализ и сравнение



# Актуальность и сложность задачи



# Идея (1)

- В мире широко применяются «гибкие» методологии разработки программного обеспечения
- **Предлагается:** для решения задачи сборки генома (и других научно-технических задач на основе алгоритмов дискретной математики) использовать новую «гибкую» методологию – на основе опыта **побед** на олимпиадах по программированию

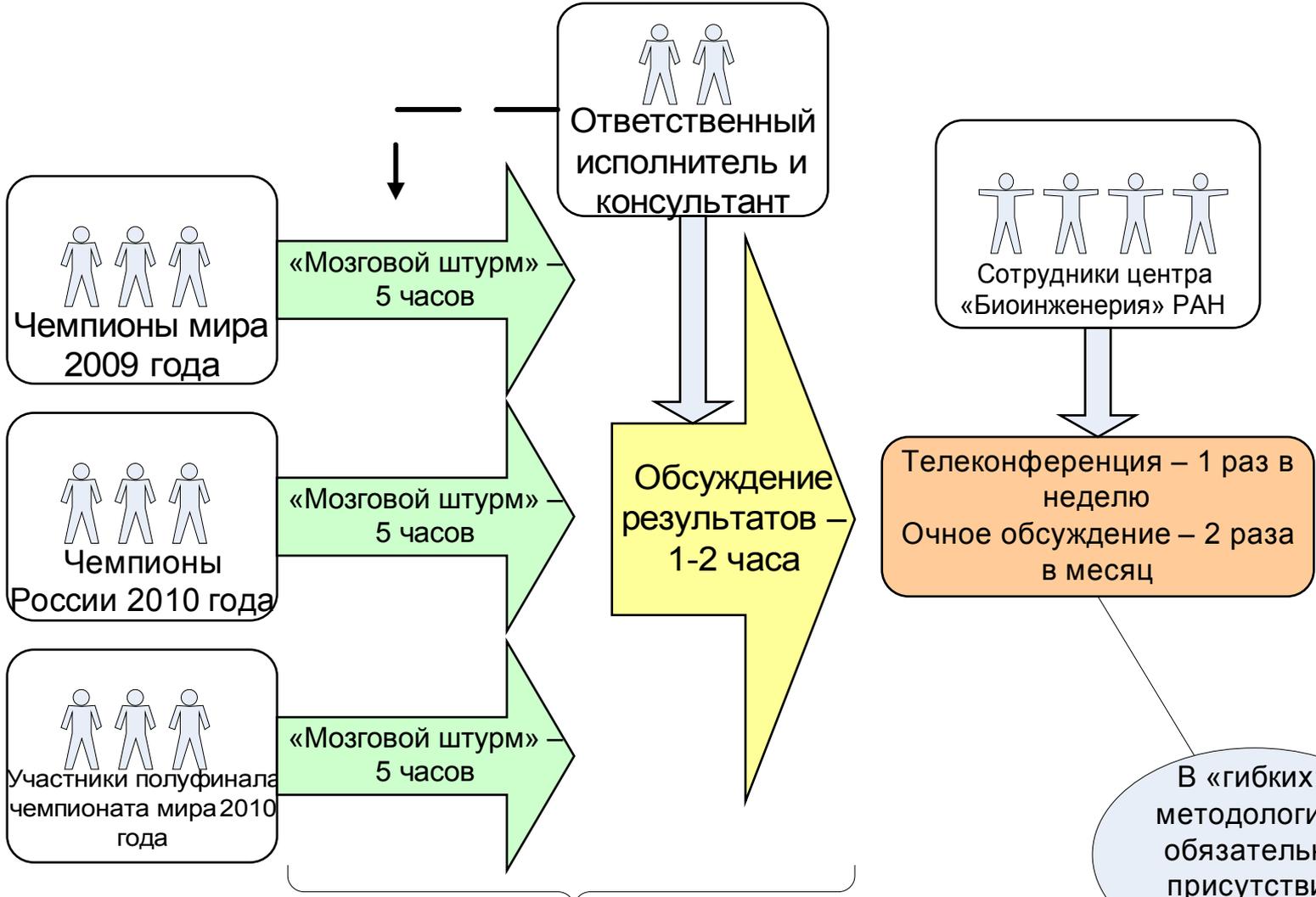
## Идея (2)

- Попытаться сделать процесс решения научных задач таким же увлекательным, как процесс участия в соревнованиях по программированию

# Команды для олимпиад по программированию

- Для участия в олимпиадах по программированию создаются команды из трех человек, участники которых совместно тренируются по 5-6 лет
- Тренировки проходят два раза по пять часов
- На тренировке обычно дается 10-11 задач для решения
- **Предлагается:** вместо этого решать **одну задачу**, например, сборку генома

# Как мы работаем? (1)



Два раза в неделю

## Как мы работаем? (2)

- 14 часов в неделю – «мозговой штурм» по командам и обсуждение его результатов
- 26 часов в неделю – индивидуальная работа: изучение статей, книг, учебников, программная реализация алгоритмов, ...

# Работа на начальном этапе

- Участие в проекте-соревновании de novo Genome Assembly Assessment Project (dnGASP, организован Национальным центром геномного анализа, Барселона, Испания)
- Искусственный геном размером 1.8 Гб
- Можно было использовать любые программные средства и любое оборудование

# Участие в dnGASP

- Был выбран путь разработки собственных алгоритмов
- Мозговые штурмы проводились не только в виде очных встреч, но и по скайпу
  - Новогодние каникулы
  - Сессия
  - Сборы по подготовке к чемпионату мира по программированию
  - ...
- Число участвующих команд сократилось сначала до двух, а потом – до одной

# Результаты dnGASP (1)

- За два месяца (конец декабря 2010 года – начало февраля 2011 года) удалось:
  - Разработать алгоритмы для сборки генома
  - Реализовать их
  - Собрать геномную последовательность из предоставленных организаторами проекта данных
- По результатам оценки сборок генома заняли не последнее место

# Результаты dnGASP (2)

- Приняли участие в итоговом семинаре в апреле 2011 года
  - European Bioinformatics Institute, UK
  - Wellcome Trust Sanger Institute, UK
  - Beijing Genomics Institute, China
  - ...
- НИУ ИТМО стал одним из примерно 20 университетов и исследовательских центров мира, обладающих технологией сборки генома

# Дальнейшая работа (1)

- Работа по разработке методов сборки геномных последовательностей была продолжена **одной командой из четырех человек**
- Эта команда стала «ядром» исполнителей проекта
- В дальнейшем, в команду вошел еще один участник
- Все, кто сейчас работает над этим проектом, имеют опыт участия в чемпионате мира по программированию

# Дальнейшая работа (2)

- Выигран государственный контракт в рамках Федеральной целевой программы «Научные и научно-педагогические кадры инновационной России на 2009-2013 годы»
- Выигран государственный контракт в рамках Федеральной целевой программы «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007—2013 годы»
- Подана статья в журнал Genome Research
- Получено свидетельство о регистрации программы для ЭВМ «Программное средство для удаления ошибок из набора чтений нуклеотидной последовательности »

# Дальнейшая работа (3)

- Июнь – сентябрь 2011 года: соревнование Assemblathon 2
- Сборка геномов трех живых существ
- Единственная команда из России, которая принимает участие
- 5 ноября – подведение промежуточных итогов

# Выводы

- Предложен подход к решению научных задач, основанных на алгоритмах дискретной математики
- Выяснилось, зачем нужны чемпионы мира по программированию
- Этот подход успешно применен при разработке алгоритмов и программного обеспечения для сборки генома
- Может быть полезным не только в университетах, но и в компаниях при разработке ядра наукоемких программных продуктов

Спасибо за внимание!

Спасибо за внимание!  
[fedor.tsarev@gmail.com](mailto:fedor.tsarev@gmail.com)  
[shalyto@mail.ifmo.ru](mailto:shalyto@mail.ifmo.ru)