

# Некоторые вопросы методики подготовки учащихся к решению олимпиадных задач по программированию

Спиридонова Д.М.  
БГПУ, г. Барнаул  
sdm@uni-altai.ru

В 1985-86 учебном году в учебных планах всех общеобразовательных школ страны появился новый предмет «Основы информатики и вычислительной техники». Сильная поддержка нового учебного предмета научным сообществом, во главе с академиками АН СССР А.П. Ершовым, Е.П. Велиховым и другими, способствовало достаточно быстрому появлению олимпиад школьников по информатике. Таким образом, уже в 1987-88 учебном году была проведена первая Всесоюзная олимпиада по информатике, а в 1988-89 учебном году при поддержке Министерства просвещения РСФСР была проведена I Всероссийская олимпиада. В дальнейшем олимпиады стали проводиться ежегодно.

Термин «олимпиада» подразумевает организованное соревнование, состязание в области каких-либо знаний или умений. На сегодняшний день олимпиады по программированию фактически сведены к отбору чемпионов в предметной области «Информатика»: программированию на языке Паскаль/Си/Си++ усложнённых алгоритмов допускающих автоматическую проверку с помощью тестов на современной технике. Организаторы олимпиад ставят перед собой следующие цели и задачи:

- пропаганда научных знаний и развитие у учащихся образовательных учреждений интереса к научной деятельности;
- создание необходимых условий для выявления одаренных детей;
- активизация работы факультативов, кружков, научных обществ учащихся и других форм внеклассной и внешкольной работы с учащимися;
- оказание помощи старшеклассникам в профессиональном самоопределении.

Рабочие органы олимпиады формируют состав жюри из числа преподавателей вузов и педагогических работников учреждений образования. Жюри в свою очередь:

- готовит задания;
- определяет критерии оценки выполнения заданий;
- проверяет и оценивает работы участников олимпиады;
- знакомит участников с результатами проверки олимпиадных работ, проводит апелляцию (при необходимости) всех туров олимпиады;
- определяет победителей;
- отчитывается перед оргкомитетом по итогам олимпиады.

Задачи, предлагаемые участникам олимпиад по информатике, выглядят как задание на сложную алгоритмическую задачу, доступную пониманию участника.

При подготовке к олимпиаде по программированию Гордня Л.В. [2] рекомендует придерживаться следующих правил подбора и комплектации типов задач:

- *Лёгкая среднешкольная задача*, посильная почти любому грамотному ученику. На районной олимпиаде 80% участников должны решить хотя бы одну задачу.
- *Трудная задача*, показательная для отбора на более высокий уровень. Нужна одна задача посильная только потенциальным победителям следующего уровня, что бы не было сомнений в объективности выбора победителей.
- *Задача на сообразительность*, не требующая техники владения методами программирования. Необходимость заинтересовать
- *Демонстрация техники программирования*. Для того, чтобы участники могли продемонстрировать свои знания и умения.
- *Задачи на понимание затуманенных, объёмных формулировок задач* (почти страница). Эта задача предназначена для выявления ценных членов команд российских и международных олимпиад, где чаще всего встречаются такие формулировки.
- *Стабильная по постановке задача*, опирающаяся на готовый текст отлаженной программы. Выполняет одну из функций конкурса - учёба в стрессовой обстановке.
- *Информатика без программирования*. Описать информацию, которую можно извлечь из заданного представления в зависимости от разных контекстов.
- *Решение в суженной системе команд*. Тренировка ума. Например, при заданной системе операций построить решение общеизвестной задачи.

Общее число задач должно быть не менее трёх. Это позволит исключить зависимость «успеха» от случайного знакомства с отдельными задачами.

Несмотря на то, что в последние годы в школьном курсе информатики, заметно сократилось количество часов, отводимых на алгоритмизацию и программирование, интерес школьников к различного рода конкурсам и олимпиадам по программированию только растёт. Но повышенная сложность олимпиадных задач по информатике ставит школьных педагогов в крайне неудобное положение. Дело в том, что методы решения большинства задач не опираются на школьную программу и требуют знание классических методов программирования и высшей математике. В связи с этим ряд педагогов выделяют так называемые классы задач, которым нужно научить решать учеников для успешного участия в олимпиадах. Данные «планы» включают в себя основные разделы алгоритмистики и понятия, которые они в себя включают.

Окулов С. М. [4] выделяют следующие классы:

- арифметика целых чисел;
- комбинаторика (подсчет комбинаторных конфигураций, комбинаторика конечных множеств, перечислительные задачи комбинаторного анализа);
- поиск и сортировка; алгоритмы на графах (связность, кратчайшие пути, циклы, потоки в сетях и т.д.);
- перебор и методы его сокращения (динамическое программирование, метод ветвей и границ, метод «решета» и т.д.);
- геометрия (формулы геометрических преобразований на плоскости, скалярное и векторное произведения, уравнение прямой, нормальный вектор, нахождение прямой, проходящей через две точки, уравнения параллельной и перпендикулярной прямых, уравнение окружности по трем точкам, пересечение прямой и окружности, принадлежность точки многоугольнику, выпуклая оболочка и т. д.);
- элементы теории формальных грамматик и абстрактных автоматов (алгоритмы синтаксического разбора выражения и построения соответствующего дерева, формулы Бэкуса-Наура, понятие лексемы, машины Тьюринга).

И даёт некоторые разъяснения:

«по каждой теме необходимо решить определенное количество задач, довести их до уровня работающих программ. Но этого мало. Мы задумываемся над тем, чему равно  $2$  умножить на  $2$ ? Нет, вероятно, только в первом классе. Без участия нашего сознания правильный ответ откуда-то извлекается. А в нашем случае? Если, например, задача, независимо от ее содержательной «упаковки», сводится после ряда преобразований к алгоритму нахождения кратчайшего пути в графе, то все - она решена. Участник олимпиады (по-другому - профессионал в определенной части информатики), установив этот факт на сознательном уровне, всю остальную работу выполняет почти как автомат, она не должна требовать значительных усилий на сознательном уровне - можно переключаться на следующую задачу. Сколько раз ребенку требуется  $2$  умножить на  $2$ , чтобы автоматически извлекать ответ? Столько же раз требуется использовать алгоритм Дейкстры для нахождения кратчайшего пути в графе при решении задач, чтобы ответ извлекался откуда-то с таким же количеством усилий, как и при умножении  $2$  на  $2$ ». [4]

Юрцовский [6] на 2-м [4] дает немного иную классификацию, более конкретную:

- 1 Числа
  - 1.1. Цифры числа
  - 1.2. Простые числа
    - 1.2.1. Простые делители
  - 1.3. Числа Фибоначчи
- 2 Сортировка
  - 2.1. Понятие сортировки
    - 2.1.1. Сортировка простым обменом

- 2.2. Поиск данных
  - 2.2.1. Линейный поиск
  - 2.2.2. Бинарный поиск
- 3 Геометрические фигуры на плоскости
  - 3.1. Прямая и отрезок прямой
  - 3.2. Треугольник
    - 3.2.1. Площадь произвольного треугольника
    - 3.2.2. Замечательные линии и точки треугольника
    - 3.2.3. Свойства треугольников
  - 3.3. Многоугольник
    - 3.3.1. Выпуклый многоугольник
    - 3.3.2. Площадь простого плоского многоугольника
- 4 Перебор и методы его сокращения
- 5 NP-полные задачи
  - 5.1. Решение NP-полных задач
    - 5.1.1. Типы рекурсивных алгоритмов
  - 5.2. Перебор вариантов
  - 5.3. Перебор с возвратом
  - 5.4. Перебор с отсечением и склеиванием ветвей
  - 5.5. Динамическое программирование
    - 5.5.1. Алгоритм
    - 5.5.2. Условия применения динамического программирования
    - 5.5.3. Реализация алгоритма ДП
      - 5.5.3.1. Сведение задачи к подзадачам
      - 5.5.3.2. Понятие рекуррентного соотношения
      - 5.5.3.3. Правильные рекуррентные соотношения
      - 5.5.3.4. Использование таблиц для запоминания решений подзадач
      - 5.5.3.5. Восстановление решения по матрице
  - 5.6. Волновые алгоритмы
  - 5.7. «Жадные» алгоритмы
    - 5.7.1. Условия применения «жадных» алгоритмов
- 6 Графы
  - 6.1. Основные понятия и определения
  - 6.2. Способы описания графов
    - 6.2.1. Матрица смежности
    - 6.2.2. Перечень ребер
    - 6.2.3. Список смежных вершин
  - 6.3. Понятие достижимости
  - 6.4. Поиск кратчайших путей в графе
    - 6.4.1. Поиск в глубину
    - 6.4.2. Поиск в ширину
    - 6.4.3. Волновой алгоритм

На одном из украинских сайтов [5], с материалами по подготовке и проведению олимпиад по информатике предлагается следующий план, по подготовке к олимпиаде:

## Раздел 1. Математические основы программирования

### Раздел 2. Техника программирования

#### 1. Основы языка программирования (Паскаль, Си).

Переменные и простейшие типы данных, размеры типов. Линейные программы. Условные операторы. Циклы. Процедуры и функции. Сложные типы данных (массивы, строки, записи, указатели, файлы).

#### 2. Массивы.

Одномерные массивы. Двумерные массивы (матрицы). Многомерные массивы.

#### 3. Строки. Элементы лексического и синтаксического разбора.

Операции над строками. Лексемы, подсчет лексем различных типов. Выделение чисел из строки.

#### 4. Работа с файлами.

Чтение и запись в текстовый файл. Преобразование полученных из файла данных в удобную структуру. Работа с типизированными файлами. Нетипизированные файлы. Буферизация ввода.

#### 5. Рекурсия.

Математические функции, задаваемые рекурсивно. Примеры рекурсивных подпрограмм. Проблема останова рекурсии. Замена рекурсии итерацией.

#### 6. "Длинная" арифметика.

Хранения в программе чисел, которые не вмещаются в стандартные типы. Арифметические операции над "длинными" числами. "Длинные" числа с десятичной частью. Извлечение корня с заданной точностью.

#### 7. Хранение информации в динамической памяти.

Хранение набора данных в линейных списках. Вставка в список, удаление из списка, поиск элемента в списке. Двусвязные списки. Понятия структур данных стека, кольца, очереди, дека; реализация их с помощью динамической памяти. Двоичные деревья. Деревья с неопределенным числом потомков. Хранение больших массивов.

### Раздел 3. Алгоритмы, методы и принципы решения задач

#### 1. Понятие сложности алгоритма.

Определение сложности. Классы задач P и NP. NP-полные задачи.

#### 2. Алгоритмы поиска и сортировки.

Поиск элемента в неупорядоченном массиве. Двоичный поиск по ключу в упорядоченном массиве (дихотомия). Поиск методом Фибоначчи. Поиск в упорядоченном n-мерном массиве. Поиск k-го по величине элемента массива. Простые методы сортировки ("пузырек", "выборка", "вставка", "подсчет"). Быстрые методы ("быстрая", "слиянием", "пирамидальная"), балансировка двоичных деревьев. Сортировка методом черпака.

#### 3. Решение задач методом перебора вариантов.

Применение рекурсии для перебора. Генерация сочетаний, размещений, перестановок и булеана множества. Полный перебор. Отсечение вариантов (эвристики). Метод ветвей и границ.

#### 4. Вычислительная геометрия и численные методы.

Длина отрезка. Уравнение прямой. Скалярное и векторное произведение. Точка пересечения отрезков. Принадлежность точки фигуре на плоскости (например: треугольнику). Площадь выпуклого многоугольника. Выпуклая оболочка множества точек: алгоритмы Грэхема, Джарвиса, "разделяй и властвуй". Ближайшая пара точек. Метод Гаусса для решения системы линейных уравнений. Нахождение решения уравнения.

5. Принцип динамического программирования.

Понятие, применимость. Сравнение с перебором.

6. Жадные алгоритмы.

Понятие, применимость. Сравнение с перебором и динамическим программированием.

7. Теория графов. Алгоритмы на графах.

Понятие графа. Определения теории графов. Структуры данных для представления графа в программе. Алгоритмы обхода графа (поиски в ширину и глубину). Лабиринт (метод волны). Эйлеров цикл. Кратчайший путь во взвешенном графе (алгоритмы Дейкстры и Минти). Транзитивное замыкание графа (алгоритм Флойда-Уоршелла). Минимальное остовное дерево (алгоритмы Прима и Краскала). Топологическая сортировка графа. Поток в сетях (алгоритм Форда-Фалкерсона). Паросочетания в двудольном графе (метод удлиняющей цепочки, потоковое решение). Задача о назначениях, назначения на узкое место (венгерский алгоритм). Игры на графах. Раскраска графа. Уложение графа на плоскости. Сильная связность и двусвязность графа. Изоморфизм графов. К-клика. Гамильтоновы циклы. Лексический и синтаксический анализ.

Задача "Калькулятор". Синтаксические диаграммы. Формы Бэкуса-Наура. Стековая и рекурсивная модель синтаксического разбора. Конечные автоматы. Грамматики.

9. Задачи с "изюминками"

На наш взгляд Математические основы программирования требуют более полного рассмотрения. Данные основы подробно рассматривают Т.Кормен, Ч.Лейзерсон, и Р.Ривест [3]:

- 1 Скорость роста функций
  - 1.1.Асимптотические обозначения
  - 1.2.Стандартные функции и обозначения
- 2 Суммирование
  - 2.1.Суммы и их свойства
  - 2.2.Способы оценки сумм
- 3 Рекуррентные соотношения
  - 3.1.Метод подстановки
  - 3.2.Метод итераций
  - 3.3.Общий рецепт
- 4 Множества
  - 4.1.Множества
  - 4.2.Отношения
  - 4.3.Функции

- 4.4. Графы
- 4.5. Деревья
- 5 Комбинаторика и вероятность
  - 5.1. Подсчёт количеств
  - 5.2. Вероятность
  - 5.3. Дискретные и случайные величины
  - 5.4. Геометрическое и биномиальное распределение
  - 5.5. Хвосты биномиального распределения
  - 5.6. Вероятностный анализ

Рассмотрев все приведенные выше классификации, следует отметить, что разбиение олимпиадных задач на классы можно вести, как минимум, по двум основаниям: условие и метод решения. На наш взгляд, эти основания тесно взаимосвязаны; более того, условие часто может указывать на метод решения. Но стоит отметить, что существуют и некоторые универсальные методы, которые годятся для многих, порой непохожих друг на друга задач. И именно эти методы очень часто применяются для решения олимпиадных задач.

#### Используемая литература

- 1 Брызгалов Е.В., Шестаков А.П. Олимпиадные задачи по программированию и их классификация // Сб. статей "Олимпиады по информатике". - Пермь: ПРИПИТ, 1998. - С. 108-138.
- 2 Городняя Л.В. О конкурсах по информатике // Компьютерные инструменты в образовании. - СПб.: Изд-во ЦПО "Информатизация образования", 2001, №1, С.35-40.
- 3 Кормен Т., Лейзерсон Ч. и Ривест Р. «Алгоритмы построение и анализ» М.: МЦНМО, 2001.
- 4 Окулов С. «Информатика в задачах»// Сайт «Разбор олимпиадных задач по информатике от Михаила Густокашина». <http://g6prog.narod.ru/okulov.rar>.
- 5 Сайт «Украинские олимпиады по программированию». <http://uoi.kiev.ua>.
- 6 Юрцева С. «Алгоритмы в школьной информатике», 2001// Сайт Барнаульского государственного педагогического университета. [http://bspu.secna.ru/Department/WMiP/Method\\_material/pos\\_zapis.pdf](http://bspu.secna.ru/Department/WMiP/Method_material/pos_zapis.pdf).