

Параллель А

- Дерево отрезков. Реализация сверху/снизу. Двумерное дерево отрезков. Сжатое двумерное дерево отрезков
- Дерево Фенвика. Многомерное дерево Фенвика. Модификация на отрезке. Дерево Фенвика для максимума.
- Декартово дерево. Декартово дерево по неявному ключу. Декартово дерево с деревом отрезков, групповые операции
- Система непересекающихся множеств. RMQ offline с использованием CHM без сведения к LCA. LCA offline с использованием CHM. Dynamic connectivity problem
- Задача о наименьшем общем предке (LCA). Связь с задачей RMQ. Двоичный подъём. Сведение LCA к $+1$ RMQ. Sparse table. Фарах-Колтон-Бендер. Сведение RMQ к LCA. Heavy-light decomposition
- Дерево ван Эмде-Боаса
- Простые алгоритмы на строках. КМП. Z-функция. Ахо Корасик. Использование хешей. Алгоритм Дюваля
- Суффиксный массив.
- Суффиксное дерево. Алгоритм Укконена
- Суффиксный автомат
- Задача о максимальном потоке. Алгоритмы Форда-Фалкерсона, Эдмонса-Карпа, масштабирования и декомпозиции потока. Алгоритм Диницы.
- Поток минимальной стоимости (mincost - maxflow)
- Задача о максимальном паросочетании в двудольном графе и задача о назначениях.
 - Алгоритм Куна
 - Венгерский алгоритм
- Игры. Функция Гранди. Ретроанализ. Алгоритм Смита
- Геометрия
 - Проверка принадлежности точки многоугольнику
 - Проверка пересечения прямой и многоугольника
 - Построение касательных к выпуклому многоугольнику
 - [Пересечение выпуклых многоугольников]
 - [Выпуклая оболочка объединения выпуклых многоугольников]
 - Алгоритм Мелькмана построения выпуклой оболочки простой ломаной
 - Рандомизированный алгоритм проверки непустоты пересечения полуплоскостей
 - Рандомизированный алгоритм нахождения окружности минимального радиуса, покрывающей заданные точки

Параллель А'

- Продвинутое применение поиска в глубину (топологическая сортировка, выделение компонент сильной связности, построение конденсации графа, задача 2-SAT, эйлеровы циклы и пути)
- Паросочетания в двудольном графе и связанные темы
- Поток. Алгоритмы Форда-Фалкерсона и Эдмондса-Карпа, масштабирования потока. Задача mincost-maxflow
- Задача о поиске наименьшего общего предка (LCA). Алгоритм двоичного подъема. Алгоритм Тарьяна для offline-поиска LCA.
- Поиск подстрок. Алгоритм Кнута-Морриса-Пратта. Z-функция. Бор. Алгоритм Ахо-Корасик. Хеши
- Суффиксный автомат
- Продвинутое динамическое программирование:
 - Рекурсивное восстановление ответа в задачах на ДП
 - Вывод объекта по номеру и определение номера по объекту
 - Способы написания ДП (с просмотром вперед, назад; рекурсия с запоминанием результата)
 - Продвинутое виды задач на ДП (ДП на ациклических графах, ДП на деревьях, ДП на подмножествах, ДП по профилю, ДП по изломанному профилю)
- Игры. Ретроанализ. Функция Гранди.
- Матрицы. Быстрое возведение в степень, быстрое решение линейных рекуррентных соотношений. Решение системы линейных уравнений методом Гаусса. Поиск обратной матрицы.
- Задачи на отрезках. Дерево интервалов (отрезков), sqrt-декомпозиция, дерево Фенвика
- Многомерные структуры данных. Многомерное дерево Фенвика. RMQ декартовых деревьев.

Параллель В

- Сортировки. Быстрая сортировка, сортировка слиянием.
- Списки. Односвязные и двусвязные, кольцевые. Хранение списков в массивах.
- Графы. Обход в глубину. Времена входа и выхода. Топологическая сортировка. Сильная связность. Конденсация. Мосты, точки сочленения. Эйлеровы циклы и пути
- Задачи на отрезках. Sqrt-декомпозиция. Дерево интервалов. Дерево Фенвика
- Двоичные деревья поиска. Декартово дерево.
- Групповые операции на дереве отрезков и на декартовом дереве.
- Геометрия. Скалярное, векторное произведения. Пересечение отрезков. Площадь многоугольника. Проверка многоугольника на выпуклость. Проверка точки на принадлежность многоугольнику. Выпуклая оболочка
- Динамическое программирование
 - Лесенки
 - Наибольшая возрастающая последовательность за $O(n \log n)$
 - ДП по подотрезкам
 - ДП по подмножествам.
 - ДП по поддереву.
 - ДП по профилю, ДП по изломанному профилю.
- Минимальное остовное дерево. Алгоритмы Прима и Краскала.
- Система непересекающихся множеств. Алгоритм Краскала с СНМ.
- Кратчайшие пути в графах. Поиск в ширину на 0-k графе. Алгоритм Дейкстры с деревом интервалов. Алгоритмы Форда-Беллмана и Флойда. Поиск отрицательных циклов.
- Парасочетания. Алгоритм Куна. Минимальное вершинное покрытие.
- Поиск подстрок. Алгоритм Кнута-Морриса-Пратта. Z-функция. Хеши. Хеш-таблицы. Бор. Динамика на боре. Алгоритм Ахо-Корасика.

Параллель В'

- Алгоритмы сортировки. Устойчивые и неустойчивые сортировки. Сложность алгоритмов сортировки.
- Бинарный поиск элемента в отсортированном массиве. Решение задач методом бинарного поиска по ответу.
- Нахождение корня функции методом деления отрезка пополам.
- Нахождение экстремума функции методом тернарного поиска.
- Задачи на обработку событий и множество отрезков на прямой.
- Метод сжатия координат.
- Метод сканирующей прямой. Нахождение площади объединения прямоугольников на плоскости.
- Скобочные последовательности и сводящиеся к ним задачи. Числа Каталана.
- Рекурсивные и нерекурсивные алгоритмы перебора всех правильных скобочных последовательностей.
- Восстановление правильной скобочной последовательности по шаблону.
- Быстрое определение последовательности по номеру и номера по последовательности.
- Алгоритмы поиска в глубину и в ширину. Применения алгоритма поиска в глубину.
- Алгоритмы поиска кратчайших путей в графе: Дейкстра, Флойд, Форд-Беллман.
- Алгоритмы построения гамильтонова пути и цикла.
- Игры на ациклических графах: определение выигрывающего игрока.
- Комбинаторные алгоритмы: перебор всех перестановок из n элементов, перебор всех k -элементных подмножеств множества из n элементов.
- Элементарная геометрия на плоскости.
- Многоугольники. Выпуклая оболочка.

Параллель С

- Структуры данных. Стек. Дек. Очередь. Куча
- Квадратичные сортировки. Сортировка выбором. Сортировка пузырьком. Сортировка вставками
- Вещественные числа (хранение в памяти, операции с погрешностью)
- Линейный поиск. Барьерные элементы. Двоичный поиск. Двоичный поиск по ответу
- Продвинутое сортировки. Быстрая сортировка Хоара. Сортировка подсчетом. Сортировка слиянием (merge)
- Динамическое программирование.
 - Количество 0-1 последовательностей заданной длины без двух подряд идущих единиц
 - Наибольшая возрастающая подпоследовательность (за $O(n^2)$)
 - Задача о рюкзаке
 - Динамика на подотрезках
- Комбинаторный перебор. Генерация всех комбинаторных объектов (двоичных последовательностей длины n , перестановок, сочетаний)
- Комбинаторика.
 - Двоичные последовательности длины n . Подсчет количества. Генерация следующей последовательности. Определение номера по объекту и объекта по номеру. k -ичные последовательности.
 - Перестановки. Подсчет количества. Генерация следующей перестановки. Определение номера по перестановке. Определение перестановки по номеру.
 - Сочетания. Подсчет количества. Генерация следующего сочетания.
 - Правильные скобочные последовательности. Генерация следующей ПСП. Подсчет количества (формула Каталана)
- Графы. Способы хранения графов. Обход в глубину. Выделение компонент связности. Поиск циклов. Топологическая сортировка.
- Кратчайшие пути в графе. Обход в ширину. Алгоритм Дейкстры. Алгоритм Флойда. Циклы отрицательного веса.
- Геометрия
 - Векторы. Точка. Вектор. Хранение в памяти. Операции над векторами. Расстояние между двумя точками. Теорема Пифагора. Длина вектора. Нормализация вектора.
 - Основы тригонометрии. Декартова и полярная система координат. Полярный угол. Радианы, градусы. Определение полярного угла точки.
 - Взаимное расположение векторов. Скалярное произведение векторов. Векторное произведение векторов. Определение взаимного положения векторов с использованием векторного и скалярного произведения. Угол между векторами. Использование $\text{atan2}([a, b], (a, b))$.
 - Взаимное расположение точек и фигур. Проверка принадлежности точки прямой, лучу, отрезку с использованием векторного и скалярного произведения. Расстояние от точки до прямой, луча, отрезка. Проверка двух отрезков на пересечение. Расстояние между

отрезками.

- Точность вычислений. Сравнение двух величин ($=$, \neq , $<$, $>$, \leq , \geq).
- Прямая. Уравнение прямой. Нормальное уравнение прямой $ax+by+c=0$. Расстояние от точки до прямой, заданной уравнением. Параллельная прямая, уравнение параллельной прямой. Пересечение прямых. Точка пересечения прямых. Параметрическое задание прямой. Уравнение биссектрисы угла.
- Луч, отрезок.
- Многоугольники. Ориентированная площадь треугольника. Ориентированная площадь многоугольника. Определение выпуклости многоугольника. Нахождение точки внутри многоугольника. Метод подсчета углов. Метод дополнительного луча.
- Окружность. Взаимное расположение точки и окружности. Взаимное расположение прямой и окружности. Уравнение касательной к окружности, проходящей через заданную точку. Нахождение точек пересечений прямой и окружности. Взаимное расположение двух окружностей. Нахождение пересечения двух окружностей.

Параллель С'

- Сложность алгоритмов.
- Логические и битовые операции: AND, OR, NOT, XOR - таблицы значений.
- Проверка на простоту, разложение на множители. Наибольший общий делитель и наименьшее общее кратное: алгоритм Евклида. Признак Паскаля. Расширенный алгоритм Евклида.
- Линейный поиск.
- Квадратичные сортировки (выбор, вставки, пузырек). Сортировка подсчетом.
- Функции. Виды параметров. Передача массивов.
- Рекурсия: НОД, Ханойские башни, перебор с возвратом.
- Стек. Стек функций.
- Теория графов. Определения графа, основных элементов графов. Типы графов. Деревья: число вершин и висячих вершин.
- Представления графов в памяти: с помощью матрицы смежности и списка ребер. Оценка сложности для простейших операций.
- DFS и BFS. Алгоритм. Реализация для простейших представлений в памяти. Оценка сложности. Восстановление пути.
- Динамика на одномерных и двумерных массивах: количество способов, есть ли способ, лучший способ.
- Динамическое программирование с 2 параметрами - длина и последний элемент.
- НОП.
- Бинарный поиск. Инвариант, время работы.
- Быстрая сортировка Хоара. Время работы в среднем/худшем случае.
- Длинная арифметика. Сложение, вычитание, умножение на короткое/длинное число.

Параллель D

- Стил ь ко да.
 - Выбор имен переменных (осмысленность, слишком длинные или слишком короткие имена, венгерская нотация).
 - Комментарии.
 - Форматирование (разные стили: пробелы и табуляции).
 - Константы.
- Целые числа.
 - Деление нацело и остаток. Получение k -ой цифры числа.
 - Обмен двух целочисленных переменных (через дополнительную переменную, сложением).
 - Обнуление переменных и целочисленной переполнение.
 - `Int64`.
- Условный оператор.
 - Конструкции `if-then` и `if-then-else`.
 - Вложенные условные операторы.
 - Объединение условий (`and`, `or`, `not`).
 - Оптимизация логических выражений компилятором (`false and X`, `true or Y`).
 - Оператор `Case`.
- Циклы и последовательности без массивов.
 - Замена всех отрицательных чисел положительными, равными им по модулям.
 - Поиск количества чётных и нечётных.
- Одномерные массивы.
 - Ввод и вывод.
 - Инициализация с помощью `fillchar`.
 - Перестановка элементов в обратном порядке.
 - Вывод массива задом-наперед без перестановки элементов в нём.
 - Циклический сдвиг.
 - Добавление и удаление элементов со сдвигом.
- Двумерные массивы.
 - Ввод и вывод.
 - Независимость индексации от визуального представления.
 - Квадратная матрица. Главная и побочная диагонали.
 - Нумерация клеток по вертикали и горизонтали.
 - Получение номера клетки по её координатам формулой.
 - Получение координат по номеру формулой.
- Линейный поиск.
 - Простой вывод для каждого элемента (“YES” или “NO”).
 - Флаг (вывод только тогда, когда элемент был найден).
 - Идём по массиву, пока не найден искомый элемент (`while`).
 - Барьерный элемент (добавление искомого элемента в конец массива, чтобы отсечь случай, когда он не был найден).

- Поиск по заданному свойству.
- Поиск ближайшего к заданному (например, по модулю разницы).
- Поиск второго по величине элемента за один проход.
- Массив, замкнутый в кольцо и его обход.
- Работа с файлами.
 - Основные операции: открытие/закрытие, считывание и запись.
 - Считывание до конца файла/строки (eof, eoln).
- Сортировка пузырьком.
- Сортировка выбором максимального элемента.
- Сортировка подсчётом.
- Понятие сложности алгоритма. Линейные и квадратичные алгоритмы.
- Деление нацело в математике. Отличия от деления нацело в большинстве ЯП.
- Некоторые свойства делимости.
- Поиск всех делителей числа перебором от 1 до N .
- Пары делителей, обратные делители. Поиск делителей перебором до \sqrt{N} .
- Почему нельзя искать делители от \sqrt{N} до N .
- Наибольший общий делитель и наименьшее общее кратное.
- Алгоритм Евклида.
 - С вычитанием.
 - С делением.
- Простые числа.
 - Проверка на простоту перебором делителей.
 - Решето Эратосфена.
 - Взаимно простые числа.
 - Разложение числа на простые множители.
- Признаки делимости.
 - Признаки делимости на 2, 5, 10.
 - Признаки делимости на 4, 8 и 25.
 - Признаки делимости на 3, 6 и 9.
 - Признак делимости на 11.
- Процедурное программирование.
 - Подпрограмма.
 - Локальные и глобальные переменные.
 - Формальные и фактические параметры.
 - Передача параметром по ссылке и по значению.
 - Отличия функций от процедур.
- Рекурсия.
 - Общие сведения, реализация точки выхода.
 - Факториал. Реализация подсчёта циклом и рекурсивная реализация.
 - Алгоритм Евклида.
 - Числа Фибоначчи. Почему не стоит реализовывать их вычисление рекурсивно.
- Типы данных: символы и строки.
- Типы данных: записи и множества.

- Дата и время. Число дней в месяце, високосные года.
- Позиционные системы счисления.
 - Перевод из десятичной системы счисления в p -ичную.
 - Перевод из p -ичной системы счисления в десятичную.
 - Перевод из p -ичной системы счисления в q -ичную через десятичную.
 - Перевод из p -ичной системы счисления в q -ичную, где p и q — степени 2.
 - Арифметические операции в различных системах счисления.
- Римская система счисления.
- Основные понятия булевой логики (высказывание и его истинность).
- Основные логические операции (отрицание, конъюнкция, дизъюнкция, импликация, эквиваленция, исключающее “или”).
- Булевы тождества и их проверка. Таблицы истинности.
- Законы булевой логики (свойства логических операций).
- Представление целых чисел в памяти компьютера. Битовые операции над целыми числами и их применения.
- Графы. Вершины и ребра, степень вершины, понятия смежности и инцидентности.
- Связность графа. Подграф. Компонента связности.
- Лемма о рукопожатиях и её доказательство (число людей, сделавших нечётное количество рукопожатий — чётно).
- Деревья, циклы, мосты, точки сочленения.
- Ориентированные графы.
- Петли, кратные ребра.
- Способы хранения графов в памяти компьютера.
 - Матрица смежности.
 - Список ребер.
 - Список смежных вершин.
- Перестановки, размещения, сочетания.
- Перестановки, размещения, сочетания с повторениями.
- Треугольник Паскаля.

Параллель К

(компьютерная безопасность)

- Ассемблер
- Реверс
- Сети. Порты и протоколы. Устройства и маршрутизация. NAT
- Криптография. История и классические шифры. Теория информации и криптоанализ. Блочные и поточные шифры.
- Введение в веб. Популярные веб-уязвимости.
- Стеганография. Разведка удаленной системы
- Wi-Fi
- Введение во вредоносное ПО.

Параллель Р

(промышленное программирование)

Параллель Промышленного программирования впервые появилась в ЛКШ в 2011 году. Она рассчитана на школьников, уже умеющих уверенно программировать сложные алгоритмы (окончивших параллель В и выше в прошлом году). В отличие от традиционных параллелей, сосредоточенных на изучении алгоритмов и решении олимпиадных задач, мы поставили задачу смоделировать работу группы разработчиков крупной IT-компании и научить вас работать в соответствии со стандартами, принятыми в промышленности.

Отметим несколько ключевых моментов, отличающих эту работу от написания олимпиадных задач:

1. Работа в команде: одновременное изменение кода несколькими разработчиками, использование системы контроля версий, создание кода, понятного другим членам команды.
2. Разработка системы тестов (unittests, mock-объекты) с целью дальнейшего безопасного изменения значительных объемов кода без потери его работоспособности.
3. Объектно-ориентированное проектирование и программирование.
4. Итерационная работа над проектом. Получение промежуточных версий проекта с последующим улучшением для доведения до идеала.

Также будет уделено внимание мультиплатформенности создаваемого приложения, особенностям эффективного использования возможностей среды разработки, другим важным особенностям работы профессионального разработчика.

За смену совместными усилиями всех школьников и преподавателей разрабатывается один серьезный программный продукт. Язык программирования: Python3. Выбор языка не случаен: с одной стороны, на каком бы языке вы не писали, вам не составит труда освоить python буквально за несколько дней. С другой стороны, python стал для большинства компаний (в т.ч. Яндекс, Google и т.п.) стандартом де факто при разработке приложений, не требующих максимальной производительности (а таких приложений в современной практике 95%). Для работы мы будем использовать среду программирования WING IDE 101 (бесплатную и мультиплатформенную) и последнюю версию python3.

В 2011 году в параллели преподавали специалисты компаний Яндекс, Google, Вконтакте, ...