



**IT ШКОЛА SAMSUNG**

# Программа обучения для школьников

на 2019-2020 учебный год

Москва, 2019 г.

[Программа «IT ШКОЛА SAMSUNG»](#) компании Samsung Electronics - это долгосрочная инициатива, которая реализуется при поддержке Министерства Образования и Науки РФ. Программа работает с 2014 года. 3 тысячи школьников успешно завершили бесплатное обучение по программе дополнительного образования в области IT и программирования в 25 регионах России за 5 лет. В 2019 году Samsung инициировал новый проект «IT Академия Samsung», который объединил все образовательные программы, в том числе IT ШКОЛУ SAMSUNG.

Разработчики программы - Samsung R&D Institute Rus (Исследовательский Центр Samsung) при участии Московского физико-технического института (МФТИ).

Программа ориентирована на школьника 9-10 класса, который:

- имеет склонность к алгоритмическому мышлению, увлекается IT-технологиями;
- владеет хотя бы одним языком программирования на уровне знания условных и циклических конструкций;
- имеет устойчивые знания по школьному курсу математики.

Особенности программы:

- Комплексный подход в построении курса: охвачены минимально необходимые базовые знания из разных областей IT и программирования с учетом уровня подготовки школьников.
- Проектное обучение: разработка учащимся индивидуального проекта в виде приложения для мобильных платформ на ОС Android.
- Ориентация на группы/учеников с различным уровнем подготовки: выделение в программе основного «ядра» и тем/материалов для дополнительного углубленного изучения (\*).

Продолжительность обучения 1 учебный год, 132 ак. часа, из них: 122 ч. - аудиторные занятия по 4 часа в неделю, 2 часа – защита проекта, 8 часов – консультации по выполнению индивидуальных проектов (в программе рекомендуемые темы консультаций помечены \*\*).

Программа обучения состоит из пяти модулей:

1. Основы программирования на языке Java - 20 часов
2. Введение в объектно-ориентированное программирование - 24 часа
3. Основы программирования Android приложений - 24 часа
4. Алгоритмы и структуры данных на языке Java - 32 часов
5. Основы разработки серверной части мобильных приложений - 22 часа

Виды деятельности учащегося в процессе обучения:

- освоение теоретического и практического материала на занятиях
- самостоятельная практическая работа: выполнение домашних заданий, мини-проектов (небольшие приложения, которые реализуются учениками преимущественно на занятиях совместно с учителем с небольшими самостоятельными доработками в качестве домашнего задания)
- разработка индивидуального проекта
- участие в вебинарах
- промежуточная помодульная аттестация в форме электронного тестирования

Далее приведено тематическое содержание модулей.

Часов	№	Описание темы
20	1	<b>Модуль 1. Основы программирования на языке Java</b>
2	1.1	<b>Здравствуй мир!</b> Знакомство с системой обучения в IT ШКОЛЕ SAMSUNG. Порядок создания, компиляции, сборки и запуска в IDE приложения Java. Порядок инсталляции IDE в домашних условиях. Выбор IDE по усмотрению преподавателя: либо IntelliJ IDEA + Android Studio или Eclipse + ADT. Последний вариант не рекомендуется использовать при работе в ОС Windows
2	1.2	<b>Типы данных и операции:</b> переменные, примитивные типы данных, арифметические операторы, выражения и присваивания
	1.3* <sup>1</sup>	<b>Представление данных в памяти:</b> представление числовых (в прямом коде) и символьных данных
	1.4*	<b>Представление отрицательных и вещественных чисел в памяти.</b> Представление чисел в обратном и дополнительном коде <b>Поразрядные операции</b>
2	1.5	<b>Логические выражения.</b> Тип Boolean. Операции отношения, логические и тернарные операции
2	1.6	<b>Условные конструкции.</b> Понятие блока, область действия блоков. Условные конструкции: if-else, switch
2	1.7	<b>Итеративные конструкции while, do-while.</b> Безусловные операторы перехода break
2	1.8	<b>Итеративные конструкции for. Массивы.</b> Вложенные циклы. Безусловные операторы перехода break с меткой, continue. Одномерные массивы, цикл for each в Java. Разбор примеров нахождения максимума и минимума, поиска на числовых массивах
2	1.9	<b>Методы.</b> Изучение понятия функций на примере методов Java. Передача параметров, возвращение результата. Видимость переменных
	1.10*	<b>Указатели на примере языка Си:</b> понятие, использование в качестве аргументов функций. Методы передачи аргументов в функцию. Динамическое выделение памяти
2	1.11	<b>Многомерные массивы.</b> Неровные массивы*
2	1.12	<i>Практикум</i>
2	1.13	<i>Контрольное тестирование по модулю</i>

<sup>1</sup> Обозначение \*- дополнительные темы, которые могут быть включены в основную программу обучения по усмотрению преподавателя

2		Модуль 2. Введение в объектно-ориентированное программирование	
24			Android практикум
2	2.1	<b>Понятие класса и объекта.</b> Цели и задачи ОО-подхода к проектированию и разработке ПО. Объект, сообщение, класс, экземпляр объекта, метод. Общее понятие о парадигмах ООП: абстракция, инкапсуляция, наследование, полиморфизм. Описание протокола класса. Обзор классов, соответствующих примитивным типам	
4	2.2	<b>Работа с экземплярами класса</b> Конструкторы и деструкторы. Статические методы. Открытые и закрытые поля. Доступ к полям объекта. Инициализация переменных, массивов, полей классов в конструкторе. Конструктор по умолчанию. Перегрузка методов на примере конструкторов	
2	2.3	<b>Строки. Основы тестирования и отладки</b> Начальные приемы тестирования и отладки, сценарии тестирования на примерах со строками	
2	2.4	<b>Знакомство с Android разработкой</b> ОС Android. Среда разработки. Принципиальная архитектура Android-приложения	Создание первого Android приложения в IDE. Изучение жизненного цикла Activity
4	2.5	<b>Интерфейс Android приложения</b> Построение простейшего интерфейса пользователя. Язык разметки XML. Описание ресурсов Android с помощью XML. Разметки (Layouts) и их применение. Представления (Views)	Создание приложения с простейшим интерфейсом и обработкой событий
4	2.6	<b>Наследование, инкапсуляция и полиморфизм</b> Производные классы и наследование. Защищенные части классов и правила доступа для классов и объектов в Java. Сравнение иерархии классов и контейнеризации классов. Полиморфные методы и позднее связывание в Java. Абстрактные методы и классы, интерфейсы.	Создание класса, хранящего информацию о человеке (возраст, имя). Реализация иерархии классов «Геометрические фигуры»
2	2.7	<b>Context и Intent</b> Понятия контекста (Context) и намерения (Intent) в Android	Намерения (Intents) в Android
	2.8*	<b>Параметризованные типы (Generic) как пример статического полиморфизма в Java</b>	Использование Generic
2	2.9	Практикум	

	2.10 **2	Формирование постановки задачи на индивидуальный проект (срок публикации в системе обучения темы индивидуального проекта – окончание 2 модуля)
2	2.11	Контрольное тестирование по модулю

24	3	Модуль 3. Основы программирования Android приложений	
			Android практикум
4	3.1	<b>Практикум ООП проектирования.</b> Разбор кейсов проектирования архитектуры классов приложения. Диаграммы UML	Разбор задания минипроекта 3.1. Получение диаграммы классов мини-проекта
2	3.2	<b>Ввод-вывод в Java. Исключения</b> Библиотечные классы ввода-вывода. Обработка исключений и классы исключений. Стандартные исключения Java. Работа с файлами в Android	Работа с классом File, как пример необходимости обработки исключений.
4	3.3	<b>Внутренние и анонимные классы</b> на примерах обработчиков событий пользовательского интерфейса	Разбор примеров использования Listener Работа над мини-проектом 3.1
2	3.4	<b>Параллелизм и синхронизация.</b> Процессы и потоки в Android. Классы AsyncTask и Thread. Реализация логики потоков. Синхронизация потоков	Разбор примера использования AsyncTask Работа над мини-проектом 3.1
	3.5*	<b>Сервисы в Android.</b> Жизненный цикл сервисов и управление им. Класс IntentService	Разбор примеров
2	3.6	<b>Фрагменты (Fragments). Сенсоры</b> Создание и управление фрагментами. Класс Fragment и его методы. Взаимодействие фрагментов и активностей. Типы сенсоров и обработка событий	Разбор примера изменения интерфейса с фрагментами с учетом данных гироскопа
2	3.7	<b>Двумерная графика</b> в Android приложениях. Класс Canvas	Реализация простейшего приложения на Canvas
4	3.8	<b>Разработка игровых приложений.</b> Этапы проектирования и реализации. Профессии в мире индустрии игр. Понятие игрового движка. Реализация графики на основе SurfaceView	Разбор игры с анимацией на SurfaceView
	3.9*	<b>Разработка 3D игр с использованием фреймворка libGDX</b> <a href="https://habrahabr.ru/post/276139/">https://habrahabr.ru/post/276139/</a>	Простейший пример с анимацией, HUD, обработкой касаний

<sup>2</sup> Обозначение \*\* - занятия для организации работы учащихся по выполнению индивидуальных проектов (проводятся по усмотрению преподавателя)

2	3.10	Практикум	
	3.11**	<i>Защита прототипа индивидуального проекта</i>	
2	3.12	<i>Контрольное тестирование</i>	

32	4	Модуль 4. Алгоритмы и структуры данных	
			Практикум
4	4.1	<p><b>Массив</b>, как базовая структура данных. Алгоритм двоичного поиска: идея, применения, реализация. Классы Arrays (массивы), ArrayList (неограниченный массив)</p>	<p>Изучение класса Arrays: заполнение, копирование, сравнение, печать, методы поиска и др. общие методы. Изучение класса ArrayList. Создание итераторов, навигация</p>
4	4.2	<p><b>Список</b>, как базовая структура данных: стеки, очереди, односвязные и двусвязные списки</p>	<p>Изучение класса LinkedList, реализующему связанные списки</p>
2	4.3	<p><b>Адаптеры в Андроид</b> Назначение и применение. Стандартные адаптеры</p>	<p>Использование готовых адаптеров ArrayAdapter и SimpleAdapter для реализации ListView</p>
2	4.4	<p><b>Реляционная модель данных</b> и реляционные схемы. Представление данных в виде таблиц. Типы связей: один к одному, один ко многим, многие-ко-многим. Проектирование простейшей БД Необходимость возникновения и история развития СУБД. Обзор и классификация современных СУБД</p>	<p>Разбор задания на минипроект по БД 4.1 (Чемпионат по футболу) Проектирование схемы БД минипроекта</p>
4	4.5	<p><b>Локальная СУБД</b> на примере SQLite. Введение в SQL. Создание и наполнение таблиц: команды CREATE, INSERT. Команда SELECT для выборки данных, UPDATE для изменения, DELETE для удаления записей из таблицы. Дополнения к запросам SELECT. Ключевые слова ORDER BY, DISTINCT. Агрегация в SELECT в запросах: COUNT, SUM, AVG, MAX, MIN</p>	<p>Знакомство с SQLite. Разбор заготовки Android приложения. Реализация изученных команд SQL в минипроекте 4.1</p>
2	4.6	<p><b>Рекурсия</b> Линейная и ветвящаяся рекурсия. Стек вызовов</p>	<p>Пример использования при обходе дерева директорий</p>
2	4.7	<p><b>Дерево</b>, как базовая структура данных. Сбалансированные деревья. Двоичные деревья</p>	<p>Практическое занятие по библиотечному классу TreeSet</p>
2	4.8	<p><b>Обзор алгоритмов сортировок</b>: пузырьковая, вставкой и быстрая в сравнении трудоемкости. <b>Компараторы</b></p>	<p>Компаратор, интерфейс Comparable. Практическое занятие по использованию методов класса Arrays, реализующих сортировку</p>

2	4.9	<b>Хэш-таблица и функция хэширования</b> Поддержка хэширования в Java, метод hashCode. Семейства контейнеров Collections и Map	Практическое занятие по библиотечным классам HashSet и TreeSet
2	4.10	<b>Ассоциативные массивы.</b> Класс Map, контейнеры HashMap, TreeMap. Хранение данных в Android Preferences	Практическое занятие по библиотечным классам, реализующим ассоциативные контейнеры
	4.11*	<b>Контент-провайдеры</b> в Android. Назначение и применение. Стандартные контент-провайдеры ContactsContract, MediaStore	Практика использования стандартных контент-провайдеров на примере контактов и медиа-файлов
	4.12*	<b>Введение в криптографию и криптоанализ.</b> Простейшие алгоритмы подстановочного шифрования	Практическое занятие на распознавание сообщения, зашифрованного подстановочным шифром
	4.13*	<b>Введение в шифрование</b> Понятие ключа шифрования. Симметричные и асимметричные ключи. Алгоритм DES. Алгоритм RSA. Примеры использования.	Практическое занятие на освоение ключей шифрования.
4	4.14	Практикум	
	4.15**	<i>Предзащита индивидуального проекта</i>	
2	4.16	<i>Контрольное тестирование</i>	



22	5	Модуль 5. Основы разработки серверной части мобильных приложений	
			Практикум
2	5.1	<b>IP-сети</b> Адресация в IP-сетях. IPv4. Автоматизация назначения IP-адресов (DHCP). Доменные имена (DNS), URL-ссылки. Несколько IP адресов для одного сайта. Популярные сетевые команды ping, tracer, ipconfig. Сервисы работы с IP-адресами.	Работа с сетевыми командами и сервисами
4	5.2	<b>Web сервер. HTTP запросы и ответы</b> Протокол HTTP. Понятие web-сервера. Запросы клиента POST и GET, коды ответов сервера, заголовки запросов и ответов. Разбор запросов и ответов сервера.	Отправка запроса на сервер с помощью формы из браузера. Обработка методов на сервере средствами Java, *PHP.
4	5.3	<b>Клиент-серверная архитектура мобильных приложений:</b> структура, схема взаимодействия сервера и клиента мобильного приложения. Формат JSON и XML. Сериализация. Библиотека Retrofit.	Отправка запросов из Android приложения. Реализация сервера средствами Java, *PHP.
4	5.4	<b>Облачные платформы. REST взаимодействие</b> Облачные сервисы <sup>3</sup> для хостинга серверной части приложений. Реализация серверной части средствами Java, *PHP. Стиль взаимодействия REST.	Реализация Android приложения REST
	5.5*	<b>Серверные СУБД</b> Хранение данных на сервере в виде БД. Индексы. Знакомство с серверной СУБД на примере PostgreSQL/MySQL. Реализация back-end части web сервера средствами Java, PHP.	Реализация БД в СУБД PostgreSQL на <a href="https://www.heroku.com/">https://www.heroku.com/</a> и Android клиента к ней.
	5.6*	Дизайн программного обеспечения и приложений <b>Material Design</b>	Разбор примеров
2	5.7	<i>Контрольное тестирование</i>	
6	5.8	Практикум. Работа над индивидуальным проектом	
2		<b>Защита проектов</b>	

<sup>3</sup> В случае невозможности использования облачного сервиса из-за неустойчивого интернет-соединения в классе рекомендуем установить виртуальную машину со всем необходимым ПО. Образ доступен в курсе для преподавателей.