

3. МНЕНИЯ

3.1. ЦИФРОВАЯ ЭКОНОМИКА, ОБУЧЕНИЕ И УПРАВЛЕНИЕ МИРАМИ

Недоря А.Е., к.ф.-м.н., ведущий разработчик,
компания WEBaby

Эта статья, скорее философская, чем технологическая, родилась из раздумья, как учить сына (и вообще детей) программированию.

Вместо вступления

Сын уже делал игры на Scratch [Торгашева, 2016], программы на Вире [Недоря, 2018], рисовал диаграммы на языке Дракон [3]. И каждый раз я чувствовал, что это полезно, но не совсем то. Это не совсем то, чему я хочу его научить: слишком много деталей, отвлекающих от главного, слишком разные подходы, это большей частью обучение деталям, а не понятиям. Я хочу научить его не программированию на каком-то языке (в какой-то среде), а гораздо более общему: умению решать задачи, или точнее, умению думать в современном мире. А для этого научить его декомпозировать сложные задачи, видеть/создавать архитектуру (устройство), уметь использовать математические методы и рекурсию, уметь написать программу для решения рутинных задач, уметь облегчать свою жизнь за счет программирования и автоматизации процессов.

Всему этому я более-менее научился за почти 40 лет в IT, хотя все это было попутно, вместе с десятком компиляторов и других программных систем, написанных на паре десятков разных языков программирования. Но 40 лет случайного тыка – это слишком долго. Нетерпеливый читатель может задаться вопросом – а где же здесь «цифровая экономика»? С одной стороны – прямо здесь, так как «цифровую экономику» делают люди, а людей надо учить. Но это далеко не все, дальше будет прямой выход, но сначала все же поговорим об обучении.

Чему и как учить будущих создателей и пользователей цифровой экономики?

Первый вопрос: нужно ли учить программированию в современном смысле? Может быть. Но точно далеко не всех. Но вот чему точно надо учить – это управлению окружающей средой и людьми, а среда все больше становится цифровой. Как мы уже писали в [Недоря, Буняк, 2017]: *«Мы говорим об отражении реального мира в виртуальном мире, и может говорить об объединении двух миров – реального и виртуального. По сути, мы на пороге перехода от 3-х мерного мира к миру большей размерности, к новой экосистеме, к новому яркому и интересному миру».*

Впрочем, мир с победившей цифровой экономикой может быть не таким уж замечательным и даже пугающим. Вспомним, например, планету Солярия из романа «Обнаженное солнце», написанного Айзеком Азимовым в 1956 г. Цитаты привожу из перевода Н.И. Виленской:

«Численность населения намеренно поддерживается в пределах двадцати тысяч – подобную цифру сочли оптимальной сами соляриане.

– Здесь работает также около двухсот миллионов позитронных роботов, партнер Элайдж.

– Иосафат! Что же получается – десять тысяч роботов на человека?

... – Почему на Солярии всего двадцать тысяч населения?

.. – Планета не пустует, – ответил Дэниел. – Она поделена на поместья, и каждым управляет один человек.

– Каждый живет в своем поместье? Двадцать тысяч поместий, и в каждом по солярианину?

– Нет, поместий меньше. В них живут также и жены владельцев.

– А городов нет? – Бейли похолодел.

– Ни одного, партнер Элайдж. Они живут совершенно раздельно и никогда не встречаются друг с другом – только в самых чрезвычайных обстоятельствах».

Предельная «цифровая экономика» – 20 000 человек и 200 000 000 роботов, и люди, которые не встречаются друг с другом... Не идем ли мы в эту сторону? К сожалению, в отличие от мира «Обнаженного солнца», нам до звезд очень далеко. Впрочем, оставим страшилки политикам и социологам. Надеюсь, что они думают не только о положительных, но и об отрицательных сценариях.

Вернемся к вопросу – «Чему и как учить?». На вопрос «Чему учить» есть простой ответ. К сожалению, этот ответ будет работать только в идеальном мире. Он состоит из двух частей:

- •Философская часть: учить надо умению решать задачи (бытовые, профессиональные и любые другие) в цифровом мире.
- •Технологическая часть: Учить надо использованию таких инструментов решения задач (в узком смысле инструментов программирования), которые можно использовать максимально широко, желательно для решения любых задач.

Технологическая часть ответа не работает в нашем мире. Просто потому, что таких инструментов пока не существует. Подумаем о том, можем ли мы хотя бы представить/придумать такие инструменты. Но немного позже.

Напомню, что я, в первую очередь, думаю об обучении детей. Хотя то, что пригодно для обучения детей, скорее всего, будет работать и для других возрастов. И второе замечание: я говорю не об обязательной, школьной программе, а о некоей подготовке к жизни, можно сказать - о профориентации, условно о «профессиональной» подготовке. При этих ограничениях, общий ответ на вопрос «как учить» становится очевидным: учить надо в такой обстановке, в которой дети сами хотят учиться: а именно, в игре и играючи. Почему? Просто потому, что это эффективно.

Идея обучения

Идея проста – берем игры, в которые дети прямо сейчас играют, и вставляем в них задачи, подсказки и все необходимое для того, чтобы дети учились не только решению конкретных задач, но и более общим понятиям и подходам – учились думать.

Какие игры можно взять? Для начала ограничимся компьютерными играми. Для того, чтобы учить на примере футбола, волейбола или другой игры реального мира надо быть профессионалом в этих играх. Для меня использовать компьютерные игры – проще и понятней.

Мы с сыном играем в Factorio [Недоря, 2017]. Это хорошая игра, которая сама по себе обучающая. Игрок попадает на планету с некоторыми начальными способами изготовления инструментов, например, кирпичи из железной руды. Он должен развить технологии вплоть до создания космической ракеты. Фишка игры в автоматизации: на каждом шаге надо строить все более сложные кластеры фабрик, каждая из которых из компонент собирает более сложные компоненты. Несмотря на упор на автоматизацию, есть действия, которые хочется дополнительно запрограммировать, но, увы, нельзя в рамках того, что дает игра. Например, деревья приходится рубить вручную.

Решение есть – надо написать плагин. То есть надо поставить задачу сыну: написать плагин для автоматизации какого-то действия, например, заготовки древесины. Но этого мало, надо еще сделать эту задачу разрешимой, то есть выстроить среду, которая позволит ему написать плагин в достаточно короткое время и с пользой в обучении думать.

Смотрим на руководство по разработке плагинов. Плагины пишутся на Lua – и это хорошо, это простой и используемый язык. Это плюс. При разработке плагина надо учитывать множество деталей (за рамками языка), которые специфичны для плагинов Factorio. Это большой минус. А что могло быть в идеале? Некая обобщенная среда разработки, которая позволяет делать плагины для разных игр. Почему я считаю, что это возможно? Просто потому, что любая игра – это набор объектов, свойств и алгоритмов поведения – иначе говоря, в основе любой игры лежит описание мира.

Представим себе среду разработки, с условным названием TotalRefiner, в которую загружается описание мира игры, например, Factorio. Назовем такое описание Метаописанием цифрового мира. После загрузки Метаописания, разработчику плагина доступны простые возможности:

1. Задать изменение свойств какого-то объекта.
2. Изменить/добавить алгоритм действия.
3. Задать создание нового объекта (по одному из шаблонов).
4. Подключить плагин к игре.

Список, очевидно, неполон, но достаточен для разработки плагина Заготовка древесины. Для этого надо добавить алгоритм действия, который запускается по нажатию новой кнопки интерфейса, примерно так: «*while «запас древесины» < 10 do «Найти дерево»; «Срубить дерево» end*». При этом все необходимое – объекты, элементарные действия, запросы свойств должны быть доступны из Метаописания. Говоря попросту, ребенок открывает раздел описания и вытаскивает нужную часть мышкой. А если какого-то действия нет, например, «найти дерево», то собирает его из частей более низкого уровня.

Возникает вопрос об используемом языке программирования. Не претендуя на единственность решения, я бы использовал синергетическое решение из Дракона и Scratch: алгоритмы записываются Дракон диаграммами, но в качестве кубиков используются компоненты в духе Scratch. А для сложных плагинов может использоваться явная схема программы (см. [Недоря, 2018]). Ниже уровнем можем быть какой-то язык программирования, но это для тех детей, которые захотят программировать. А для решения большинства задач диаграммы будет достаточно.

Подведем итоги. Мы можем представить себе решение, позволяющее обучать ребенка на любых компьютерных играх, если у нас есть инструмент сборки плагинов, мета описание целевого цифрового мира и набор методически выстроенных задач, решение которых подготовлено.

Прыжок в цифровую экономику

Есть ли принципиальная разница между реальным миром и виртуальным миром игры? Безусловно, есть. В реальном мире, кроме цифровых объектов, есть разумные существа, эмоции, обиды, женская/мужская логика, две беды нашей страны и так далее.

Тем не менее, возможно ли построить мета описание части реального мира, которое позволит разрабатывать полезные плагины? Ответ: безусловно, да. Например, в [Недоря, 2017] описано несколько примеров полезных плагинов реального мира. Более того, огромное количество людей занимается (осо-

знаявая это или нет) построением цифрового описания части мира. Замечу, что без такого описания невозможно создание настоящего (сильного) искусственного интеллекта (ИИ). Я не говорю про создание AID – artificial idiots, создание которых уже возможно, например, AID, играющий в Го или шахматы.

Будем ли мы догонять или попробуем возглавить? Вот в чем вопрос. Если хотим возглавить, то надо «тренироваться на кошках».

Литература

1. Торгашева Ю.В. «Первая книга юного программиста. Учимся писать программы на Scratch» – СПб: Питер, 2016.
2. Недоря А.Е. «Технология разработки мультиплатформенных программ на основе явных схем программ» // <http://digital-economy.ru/stati/tehnologiya-razrabotki-multiplatformennykh-programm-na-osnove-yavnykh-skhem-programm>, 2018
3. Паронджанов В.Д. «Дружелюбные алгоритмы, понятные каждому» – М.: ДМК Пресс, 2016.
4. Недоря А.Е., Буняк В.В. «Интернет — в поиске чистого воздуха» // <http://digital-economy.ru/stati/internet-v-poiske-chistogo-vozdukha>, 2017
5. Сайт игры Factorio // <https://www.factorio.com>
6. Недоря А.Е. «Забывтое 40 лет назад новое, и как оно может изменить нашу жизнь» // Сборник трудов SoRuCom-2017, М.:ФГБОУ ВО «РЭУ им. Г. В. Плеханова», 2017, стр. 243-250.

Недоря Алексей Евгеньевич

Ключевые слова

цифровое будущее, обучение детей

Nedorya Aleksey, Digital economy, training and management the worlds

Keywords

digital future, training of children

Abstract

This article, rather philosophical, than technological, was born from thought how to teach the son (and in general children) to programming.

DOI: 10.34706/DE-2018-04-08