

УДК 519.685

К ВОПРОСУ СТАРТОВОГО ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ*Сандро Барнабишвили, Теодор Заркуа*

Грузинский университет им. Святого Андрея Первозванного Патриаршества Грузии

Аннотация

Описывается методика стартового обучения программированию, основанная на использовании модификации машины Тьюринга. Сформулированы требования к соответствующему программному обеспечению. Приводятся примеры, иллюстрирующие использование предлагаемой методики. Обосновываются преимущества рассматриваемой методики. Излагается взгляд на возможное развитие данной методики..

Abstract

This paper is intended to describe a concept of learning computer programming basics based on a modified version of the Turing machine. Main requirements to the corresponding software are formulated. The publication contains examples illustrating the use of the given approach. There are justified the benefits of the described methodology. Finally, considerations about the further development of the concept are stated.

Введение

Актуальность эффективной методики стартового обучения программированию сегодня очевидна. Развитие компьютерных технологий вызывает необходимость в уменьшении нижней границы возраста специалистов, способных участвовать в разработке современных программных продуктов. Растет число стран, которые бьют тревогу видя, что уже в очень недалеком будущем велика вероятность возникновения дефицита квалифицированных специалистов сферы компьютерных технологий. Развитие многих отраслей деятельности человека выдвигает качественно новые требования к программному обеспечению, используемому в этой области. Традиционная схема создания программного обеспечения, когда от программиста не требуется знания предметной области уже не может обеспечить требуемое качество продукта. Соответственно, знание основ программирования зачастую становится необходимо специалистам предметных областей далеких от физико-математических и инженерных наук. Между тем, в ряде государств, в том числе в Грузии, в программу общеобразовательных школ не входит обучение программированию и эта задача полностью возлагается на Вузы. Исключение составляет очень небольшое количество специальных школ и учебных центров. Ситуация усугубилась после прекращения поддержки развития систем программирования, основанных на Паскале (методики, основанные на обучении основам программирования на базе языка Паскаль были де факто стандартом). С другой стороны, молодые люди со школьных лет привыкли иметь дело с компьютером, но не зная основ программирования, начинают проявлять чрезмерное внимание к развлекательному программному обеспечению и этим самым наносят вред своему здоровью, бюджету и ресурсу времени. Зачастую это становится причиной серьезных неприятностей во вполне благополучных семьях.

Требования к методике стартового обучения программированию

В современных условиях методика обучения программированию должна позволять использовать компьютеры в учебном процессе как можно раньше без ущерба требованию эффективности его использования. Чисто психологически уже неприемлема методика обучения программированию без использования компьютера.

Ниже дается концепция методики стартового обучения программированию на базе модификации машины Тьюринга, которая была разработана исходя из следующих требований:

1. Сохранить основные части машины Тьюринга – информационную ленту, головку считывания-записи, а также эквивалент таблицы состояний;
2. Возложить функции таблицы состояний на программу, написанную в терминах модифицированной машины Тьюринга;
3. Сохранить сочетание минимума средств программирования с соблюдением условия универсальности при отсутствии ограничения на ресурсы;
4. Обеспечить простоту при использовании в учебном процессе;
5. Обеспечить максимальную наглядность.

Описание модифицированной машины Тьюринга и примеры ее использования

Исходя из вышесказанного, была разработана модификация машины Тьюринга, которая состоит из ленты, разделенной на однородные ячейки (естественно, конечной), головки считывания-записи, в каждый данный момент указывающей на текущую (актуальную) позицию, управляемой посредством всего 6 команд. Вот эти команды:

1. L – сдвиг головки на одну позицию влево;
2. R – сдвиг головки на одну позицию вправо;
3. S – прекращение выполнения программы;
4. W<c> – запись символа <c> в текущую позицию ленты;
5. G<m> – безусловный переход в программе на метку <m>;
6. I<c><m> – переход в программе на метку <m> при условии, что в текущей позиции символ <c>.

Описанная модификация машины Тьюринга дополнительно предусматривала, что каждая команда занимает отдельную строку, а в качестве метки можно использовать любое неотрицательное целое число, не превосходящее 99999, причем метка в программе также занимает отдельную строку – непосредственно перед помечаемой ею командой.

Очевидно, ознакомление с приведенной системой команд не требует длительного времени (как правило, для этого бывает достаточно 30 минут), после чего уже можно приступать к решению простейших задач. Условие каждой задачи включает описание стартового состояния машины, и описание финального состояния. Условие может включать пояснительные примеры.

Задача 1. Переход головки в правый конец последовательности символов, если на старте она находится в области, занятой последовательностью.

Стартовое состояние ленты. На ленте изначально любая непустая последовательность символов, не включающая пробелы. Все остальные позиции ленты заполнены пробелами.

Стартовое положение головки. Головка изначально может указывать на любое место вышеупомянутой последовательности.

Финальное состояние ленты. Финальное состояние ленты идентично стартовому.

Финальное положение головки. Головка должна указывать на крайне правый символ беспробельной последовательности символов, расположенной на ленте.

Пример 1.

Стартовое состояние

		I	O	N	1	4		
--	--	---	---	---	---	---	--	--

Финальное состояние

		I	O	N	1	4		
--	--	---	---	---	---	---	--	--

Пример 2.

Стартовое состояние

Финальное состояние



Решение может иметь следующий вид:

10	20
R	L
I 20	S
G10	

Для экономии места, программа выведена в два столбца (этот же прием использован в реализации при выводе на экран). Как видим, даже в такой простой задаче мы имеем возможность продемонстрировать ветвление и цикл. Слегка смодифицировав условие, а именно, допустив, что вначале головка может находиться не только на последовательности, но и слева от неё мы сразу получаем возможность продемонстрировать довольно полезную при обучении ошибку – зацикливание во вроде бы естественном с точки зрения преемственности решении.

5	I 20
R	G10
I 5	20
10	L
R	S

Тот, кто имеет некоторый опыт программирования может заметить неочевидную ошибку. Программа зацикливается, когда головка с самого начала находится на крайнем правом символе последовательности. Тут же можно предложить красивый выход из положения, организовав т.н. прецикл (аналог музыкального затакта) добавлением в самом начале одного сдвига влево.

Желание иметь возможности, эквивалентные подпрограммам (включая рекурсию) привело к небольшому расширению системы команд (в этом направлении есть риск переборщить). А именно, добавление символа R в конце обеих команд перехода (G,I) превратило их в переход с возвратом. А возврат был возложен на команду S. Эта команда осуществляет возврат на команду, непосредственно следующую за последней по времени выполнения командой перехода с возвратом, на которую возврата еще не было и завершает программу, если нет ожидающих возвратов.

Задача 2. Проверка корректности скобочной последовательности.

Стартовое состояние ленты. На ленте изначально любая непустая скобочная последовательность состоящая только из круглых скобок и не содержащая пробелы. Все остальные позиции ленты заполнены пробелами.

Стартовое положение головки. Головка изначально указывает на позицию, примыкающую слева к крайней левой позиции последовательности.

Финальное состояние ленты. Финальное состояние ленты идентично стартовому.

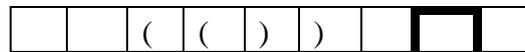
Финальное положение головки. Если последовательность корректна, то головка должна быть двумя позициями правее от самого правого символа последовательности, иначе – в любом другом месте.

Пример 1.

Стартовое состояние



Финальное состояние



Решение может быть таким.

10	I(20
R	R
20	S
I(10R	

Как видим, несмотря на лаконичность, решение явно не тривиально. Ту же самую задачу решить без применения расширенных возможностей неизмеримо труднее.

Заключение

Подытоживая вышеизложенное можно сказать, что имеет место одновременно и простота и универсальность. С другой стороны, удастся покрыть вопросы, связанные с такими элементами программирования, как ветвление, цикл, рекурсия. Сложность заданий практически неограниченна. Благодаря возможности иметь на экране одновременно как ленту с головкой, так и исполняемую программу, обеспечена наглядность. В дальнейшем планируется, с одной стороны, привести средства простого препроцессинга и этим самым получить возможность раскрыть элементы модульности, с другой стороны, развить программную реализацию в направлении создания арены, которая позволит брать задачи из сети и там же отлаживать их. Возможности учета работы пользователей позволят внести их ранжирование, что также в интересах учебного процесса. Ученик, успешно освоивший программирование на предлагаемой модификации машины Тьюринга, в дальнейшем изучает программирование на языке высокого уровня явно в комфортных условиях, соответственно, с большими шансами на успех в обучении.

Список использованных источников:

1. Пратт Т., Зелковиц М. Языки программирования. Разработка и реализация. СПб., Питер, 2002. – 688 с.
2. Успенский В.А. Машина поста. Москва, «Наука». 1979. – 96 с.

Статья получена: 2015-03-20