

РОБОТ - МОДЕЛЬ ЦЕХОВОГО КРАНА

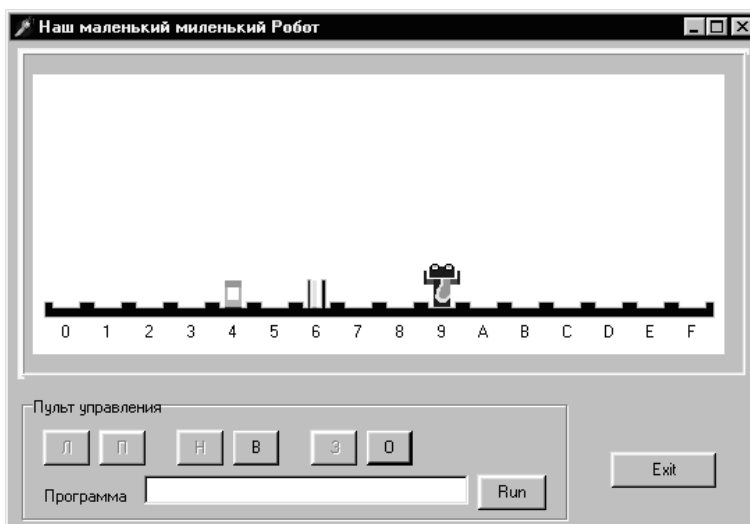
В преподавании информатики, мне кажется, целесообразно использовать так называемые «длинные задачи». В такой задаче некоторый сюжет развивается из занятия в занятие, обрастая все новыми подробностями. Такие задачи ближе подходят к практическому программированию, где человек или коллектив авторов долго работает в какой-нибудь «предметной области» и профессионализируется. Такая профессионализация должна быть одним из аспектов обучения программированию. Учащиеся должны привыкать к задаче. При этом они создают себе различные заготовки, которые используют в дальнейшем: инструменты, структуры данных, процедуры, просто заготовки текста, и то, что было сделано на прежних уро-

ках, пригодится на следующих. Кроме того, таким способом можно создавать более интересные вещи, чем при постоянной смене области.

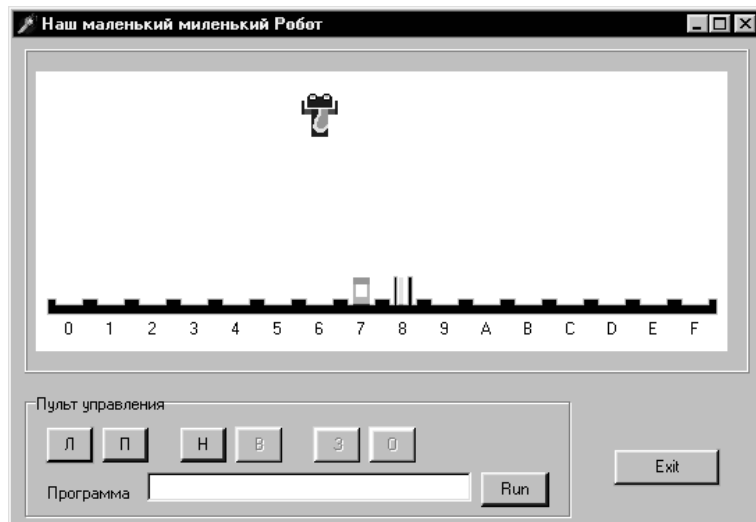
Я думал о нескольких разновидностях таких «длинных задач». Это может быть создание редактора текста: научите его сначала подравнивать края, затем переносить слова, затем искать «висячие строки». Много интересных задач возникает в вычислительной геометрии, если вспомнить те расчеты, которые проводили геодезисты во времена начального измерения Земли. Можно предложить и такую идею с моделью «робота».

Первая ситуация: представьте себе, что дискретные гнезда расположены на прямой линии на равных расстояниях. В каждом гнезде стоит (или может стоять) какой-то груз. Для удобства грузы можно занумеровать. Над этими гнездами перемещается «робот» - простейшая модель цехового крана. Этот робот может выполнять следующие действия: переместиться на шаг влево или вправо (точные положения - над этими гнездами), опуститься, захватить груз, подняться. С грузом он также может перемещаться вправо или влево.

Первоначально предлагается готовый имитатор,



На приведенной иллюстрации робот опустился вниз и захватил груз в гнезде № 9.



На этом рисунке робот выполняет задание: перенести груз из гнезда № 6 в гнездо № 9.

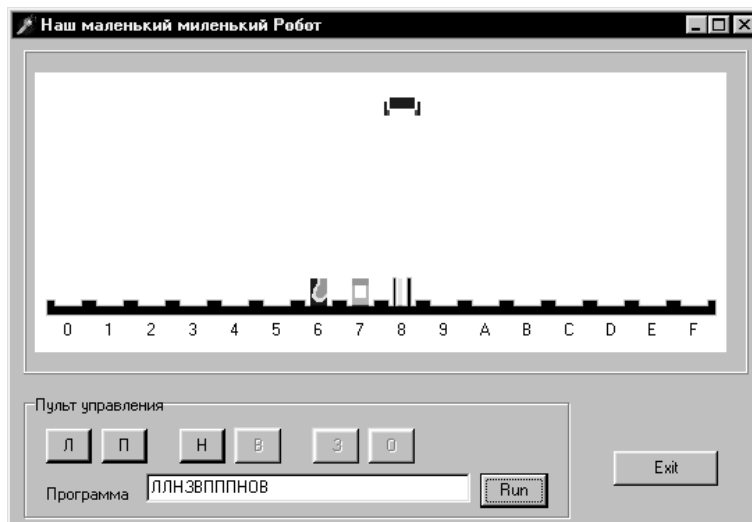
в котором ученик (практически любого возраста, хотя хорошо, если он может прочесть буквы на кнопках) может, нажимая кнопки, управлять роботом. И, немного с ним поиграв, он понимает, что робот воспринимает основные команды, соответствующие тем или иным клавишам.

Далее, расположив в гнездах некоторые детали, можно дать задание - перенести груз из одного гнезда в другое.

После этой предварительной подготовки задача ставится так: напишите программу, т.е. строку, состоящую из команд,

и предложите ее «роботу». Как видите, получается система, которая исполняет программы. Это очень полезно, поскольку ученик может быстро отладить такую программу, посмотреть, что получается, а что не получается. Можно добавить звуковые эффекты - например, в случае некорректной команды типа перемещения вправо, когда робот находится в крайнем правом положении.

После этого - следующий шаг: создать условный язык, который «укрупнял» бы простые команды. Например, вместо того, чтобы 3 раза писать букву «л» (сдвиг влево на 3 шага) можно написать «Зл» «лЗ» и преобразовать это сообщение в текст на базовом языке. Как видите, мы уже пишем маленький транслятор (точнее, интерпретатор с обработчиком макрокоманд). Далее можно описать несколько процедур и приучать школьника к правильной организации тела цикла - к тому, как нужно писать кратно используемый блок команд.



Если набрать программу ЛЛНЗВПППНОВ и нажать "Run", робот "самостоятельно" перенесет груз из гнезда № 6 в гнездо № 9. Если нажать "Run" еще раз, робот снова перенесет груз слева направо.

В развитии этой модели у учащегося должны выработаться и навыки «разумной лени»: он дол-

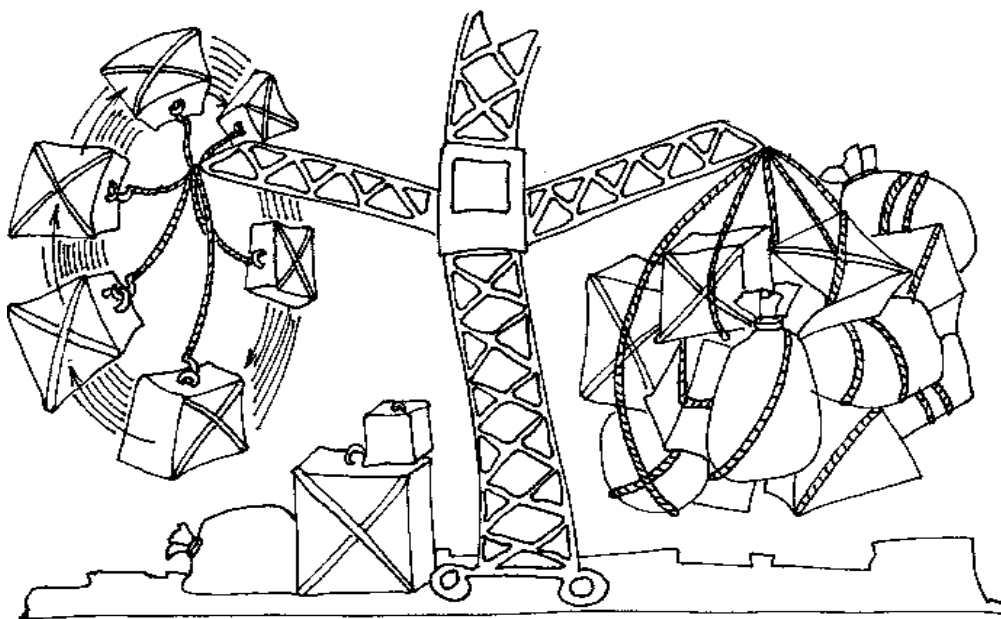
жен научиться перекладывать на машину длинную и скучную работу.

После этого можно идти дальше и заниматься «вычислением задания для робота». Представьте себе такую задачу: задано начальное и конечное расположение груза, и требуется написать программу, которая по этим данным выдавала бы программу для переноса этого груза. Здесь придется анализировать, где может быть робот, сколько раз надо переместиться вправо или влево, и так далее.

Теперь следующий шаг. Представьте себе, что для предыдущего случая надо решить так называемую задачу о коммивояжере, то есть перенести несколько грузов, причем самым «экономным» способом (с наименьшим числом перетаскиваний или с минимальной длиной программы в «тактах»). И результат оформляется в виде программы.

Это материал на первые 4-5 занятий, после чего вы можете усложнить задачу: пускать этого робота на плоскости, ставить ему препятствия, и так далее. Причем такой сюжет позволяет обеспечить интересным делом и, скажем так, «младших» школьников, которые осваивают робота, и «старших», которые добавляют ему все новые функции (вот пример «яркой картинки» - робот, вывешивающий на корабле флаги расцвечивания, составляющие требуемую фразу).

В принципе, аналогичными командами можно смоделировать не только кран (кстати, козловой кран на контейнерной площадке порта ездит в двух направлениях и должен иногда вытаскивать нижний контейнер из стопки), но и другие механизмы - скажем, фрезерный или сверлильный станок, графопостроитель («Привет, LOGO-Черепашка!»).



*Романовский Иосиф Владимирович,
профессор кафедры исследования
операций мат.-мех. факультета
СПбГУ.*