

такого мира, в котором эта система будет работать. Чем больше тавтологий мы отыщем в нашей системе, тем больше катастроф этого мира мы сможем предсказать.

Данный тип задач требует особого внимания и сосредоточенности, так как одна ошибка может привести к краху всей системы. И тогда дальнейшие подсчеты и поиски тавтологий будут бессмысленны.

Список литературы

1. А. С. Карпенко. Фатализм и случайность будущего: логический анализ. – М.: Наука, 1990.

УДК 168.1

ОБЩАЯ ХАРАКТЕРИСТИКА ЗАДАЧ НА ПРОГРАММНУЮ ОПРЕДЕЛИМОСТЬ

С.В. Павленко
Симферополь, Украина

Проблема программной определимости в рамках теории определений открыта сравнительно недавно. В наиболее развернутой форме эта тема разработана в работе Николко В. Н. «Аналитическая определимость явлений»[1], где для примера приведены задачи на программирование работы машины Поста.

Однако, прежде чем говорить о программной определимости нужно выяснить, что собой представляет термины «определимость» и «определение». Довольно часто эти два термина используют как синонимы и нечетко различают, что, по сути, является не вполне корректно.

«Под определением понимается обычно логический прием, позволяющий: а) отличать, отыскивать интересующий нас предмет; б) уточнять значение уже введенного в науку термина, а также сформировать значение вновь вводимого термина»[2].

Что касается определимости, то здесь можно дать следующее философское обобщение: «отношение между фрагментами действительности X и A, состоящее в том, что X может являться как A, в качестве A, выступать в ипостаси A, принимать форму и облик A, эквивалентно замещаться в ряде случаев на A, не будучи A, назовем относением определимости, что будем записывать как X//A, где X – определяемый, а A – определяющий фрагмент действительности»[1, с. 4].

Программная определимость является разновидностью аналитической определимости. Этот термин впервые введен в литературу Николко В. Н. в работе «Аналитическая определимость явлений»[1].

В процессе пояснения программной определимости стоит дать некоторые допущения.

«Пусть имеется некоторое множество |M| с элементами, комбинациями, конфигурациями. Пусть, далее, имеется некоторое явление (природа которого может быть произвольной), способное перемещать элементы |M| внутри комбина-

ций, конструкций, конфигураций; конструкции внутри конфигураций и т. д. Иными словами: имеется группа способностей, перемещающая одни фрагменты внутри других фрагментов в $[M]$ » [1, с. 38].

Свыше пятидесяти лет назад, в 1935 году, выдающийся американский математик Эмиль Л. Пост опубликовал в «Журнале символической логики» статью «Финитные комбинаторные процессы. Формулировка 1» [3, с. 83–88].

В 1967 году Успенский В.А. пересказывает статью с новых позиций. Он вводит термин «машины Поста». Машина Поста – абстрактная машина, которая работает по алгоритмам, разработанным человеком.

Машина Поста состоит из ленты и каретки (считывающая и записывающая головка). Лента бесконечна и разделена на ячейки одинакового размера. В каждую ячейку ленты заносится один символ двоичного кода информации, который подлежит обработке. Один из символов двоичного алфавита – метка «V», другой – пустота. Если в ячейку занесена «V» – ячейка отмеченная, если в ячейке «V» нет – ячейка пустая, или неотмеченная. Каретка может передвигаться вдоль ленты влево (команда обозначается символом – «←») и вправо («→»). Когда она неподвижна и стоит против ровно одной ячейки ленты, принято говорить, что каретка обозревает эту ячейку. А такая ячейка называется текущей, или обозреваемой.

За единицу времени, которая называется шагом, каретка может сдвинуться на одну ячейку влево или вправо. Кроме того, каретка может также распознать, стоит или нет метка в обозреваемой ею ячейке (« \langle »), может заносить метку в пустую секцию («V») и может удалять метку из отмеченной ячейки (« ξ »). В программе также должна быть команда результативной остановки («Stop»), уже не предусматривающей отсылку к следующей команде.

Команды, по которым каретка должна занести метку в отмеченную секцию или удалить метку из пустой ячейки, являются недопустимыми.

Командой является последовательность символов nKm , где n – номер текущей команды, K – команда из системы команд машины Поста, m – отсылка – номер команды, которая будет выполняться следующей.

Все виды задач, пожалуй, можно разделить на несколько видов:

1) Задачи, где требуется определить границу заданного массива пустых или занятых ячеек без установок новых меток. Здесь основные команды, которые будут использоваться – это движение каретки влево («←») и вправо («→»), результативная остановка («Stop»), а также команда обозрения наличия или отсутствия в текущей ячейке метки « $n\left(\frac{k}{m} 1\right)$ », где n – номер текущей команды, k – номер команды, куда произойдет отсылка, если ячейка пуста и m – номер команды, которая будет выполняться, если ячейка занята.

2) Следующий вид заданий заключается в установке меток в каких-либо пустых ячейках. Здесь уже используется команда установки метки («V») в незанятой ячейке.

3) Третий вид задач сводится к преобразованиям массивов занятых и свободных ячеек на ленте, исходя из заданных условий. Здесь уже используются все виды вышеперечисленных команд, включая команду стирания метки (« ξ ») в занятой ячейке.

Третий вид задач включает две предыдущие. Большинство задач на программную определимость относятся именно к третьему виду.

В качестве примера можно представить задачу, в которой требуется определить местонахождение массива занятых ячеек. Известно, что положение массива меток и каретки на ленте произвольно.

1 $\leftarrow \frac{2}{15}$	- определение: имеется ли в этой ячейке метка, которую мы ищем	6 \rightarrow 7	- движемся вправо к «служебной» метке, удаляем ее и ставим в соседней ячейке метку, если она свободна	12 \leftarrow 13 13 $\leftarrow \frac{12}{14}$	- движемся к левой «служебной» метке, удаляем ее и повторяем команды начиная с 3
2V3 3 \leftarrow 4	-делаем метку на первоначальном месте каретки и движемся налево, чтобы поставить метку, если ячейка пуста	7 $\leftarrow \frac{6}{8}$ 8 ξ 9 9 \rightarrow 10		14 ξ 3	
4 $\leftarrow \frac{5}{15}$ 5V6		10 $\leftarrow \frac{11}{15}$ 11Vi2		15Stop	- результативная остановка на которую ссылаются

В заключение можно сказать, что эти задачи на программирование машины Поста на сегодняшний день служат только для развития навыков построения программ на начальном уровне. Также они дают представление о программной определимости как виде аналитической определимости.

Список литературы

1. *Николко В. Н.* Аналитическая определимость явлений. – Симферополь: издательство ТНУ, 2004.
2. *Горский Д. П.* Определение: (Логико-методологические проблемы). – М.: Мысль, 1974.
3. *Успенский В. А.* Машина Поста. – М.: Наука, 1973.

УДК 168.1

КЛАССЫ ЗАДАЧ НА ОПРЕДЕЛИМОСТЬ*

В.Н. Николко
Симферополь, Украина

1. Определимость как явление – шире определения: определимость – это возможность, а определение – действительность определительных процессов. Это отношение между некоторым явлением X и множеством M , в которое X не входит. Единственное условие, предъявляемое к M , состоит в том, что M – структура, т.е. множество, в котором имеют место ностадийные преобразования как элементов M , так и самих результатов преобразований. Для этого предполагается: в M , помимо элементов, входят преобразующие средства L .

Преобразования L , как и всякие преобразующие силы, превращают любой элемент, принадлежащий M , в элемент, не обязательно принадлежащий M . Будем говорить, что X определимо в структуре M , если и только если предъявлен процесс $s_1, s_2, s_3, \dots, s_n$, такой, что:

1. Любое S_j – либо элемент M , либо получено из предшествующих S_j элементов (т.е. S_{j-1}, S_{j-2} и т.д.) посредством L ;