

УДК 519.683

В.Г. Акуловский, А.Е. Дорошенко

## СОСТАВ И СВОЙСТВА ДАННЫХ, СПЕЦИФИЦИРУЕМЫХ В КОМПОЗИЦИОННЫХ СХЕМАХ АЛГОРИТМОВ

Рассмотрены данные, специфицируемые в композиционных схемах алгоритмов. Показан их состав, структура, свойства и возможность контроля корректности их описания.

### Введение

Достаточно давно осознана принципиальная важность роли, которую данные играют в программировании [1, 2]. Более того, часто утверждается [3, 4], что информация о данных для понимания программ важнее, чем информация об управлении.

Таким образом, в описании алгоритма наряду с описанием потоков управления должны быть отображены и данные. При этом роль данных и их влияние на процесс проектирования алгоритмов в рамках известных алгебраических аппаратов изучены и используются далеко недостаточно.

Для изучения влияния данных на процесс проектирования алгоритмов они были формализованы следующим образом [5].

**Определение 1.** Данными называется пара  $D = \langle N, Z \rangle$ , где  $N$  – носитель данных,  $Z$  – кортежей значений, носителем которых является  $N$ . На каждом шаге вычислительного процесса носитель содержит (хранит) некоторый (текущий) кортеж значений данных, в частности, эти значения могут быть неопределенными.

Для того, что бы “имплантировать” данные в алгебраический аппарат, известная модель ЭВМ Глушкова дополнена внешней средой (ВС).

Внешняя среда – память и внешние устройства представляют собой множество данных  $D^{BC} = (D^I \cup D^{BY}) = \{D_1, \dots, D_n\}$ , характеризующиеся в каждый момент времени некоторым множеством значений, которые, изменяются в ходе вычислительного процесса.

На основе модифицированной модели ЭВМ авторами предложена система алгоритмических алгебр (САА/Д) [5 – 7], представляющая собой трех-основную алгебраическую систему  $\langle U, L, D, \Omega \rangle$ , основами которой являются множество  $D$ -операторов  $U$ , множество логических условий  $L$  и множество данных  $D$ , а  $\Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3$  – её сигнатура, состоящая из  $\Omega_1$  – логических операций, принимающих значения на множестве  $L$ , множества  $\Omega_2$  – операций, принимающих значения на множестве  $D$ -операторов, и  $\Omega_3$  – операций, принимающих значения на множестве данных  $D$ .

$D$ -операторы – это операторы вида  $(D)O(D')$  со специфицированными на входе и выходе данными такими, что  $D \subseteq D^{BC}$  и  $D' \subseteq D^{BC}$ , чем и обусловлено использование названия  $D$ -оператор. Такой  $D$ -оператор обрабатывает данные, т.е. анализирует и преобразует, изменяя их значения.

Одна из форм описания алгоритмов – композиционные схемы (КС) [6], в которых для описания алгоритмов используется операция (в САА/Д операция обозначается “\*”) композиции  $D$ -операторов

$$(D_1)X_1(D'_1) * (D_2)X_2(D'_2),$$

означающая последовательное выполнение сначала  $D$ -оператора  $(D_1)X_1(D'_1)$  затем  $D$ -оператора  $(D_2)X_2(D'_2)$ .

С помощью КС реализуются две стратегии проектирования алгоритмов нисходящая и восходящая [8, 9]. В данном случае рассмотрим первую – более распространенную стратегию, в рамках

которой исходный Д-оператор  $(D)O(D')$  декомпозируется, а специфицированные на его входе и выходе данные детализируются, в результате чего он представляется в виде КС:

$$(D)O(D') = (D_1)O_1(D'_1) * (D_2)O_2(D'_2) * \dots * (D_k)O_k(D'_k),$$

где  $(D_1)O_1(D'_1), (D_2)O_2(D'_2), \dots, (D_k)O_k(D'_k)$  – производные Д-операторы, полученные в результате декомпозиции исходного.

Одна из основных задач, решаемых в процессе описания алгоритмов в виде КС – это спецификация данных на входах и выходах производных Д-операторов. Актуальность этой задачи обусловлена тем, что наличие таких спецификаций для каждого Д-оператора на каждом уровне представления алгоритма являются необходимым условием для перехода к следующему его уровню, т. е. необходимым условием для определения функциональности производных Д-операторов, получаемых в результате декомпозиции исходного. При этом данные на входе и выходе Д-оператора специфицируются однократно, т.е. дублирование спецификаций недопустимо.

Сложность указанной задачи обусловлена тем, что производные Д-операторы, входящие в КС, обрабатывают глобальные, специфицированные на входе и выходе исходного Д-оператора, данные как “автономно”, так и “коллективно”. То есть, в общем случае каждый из производных Д-операторов, с одной стороны, обрабатывает данные независимо от других производных Д-операторов. С другой – обработка множества (подмножества) глобальных данных, начатая некоторым Д-оператором  $(D_i)O_i(D'_i)$ , продолжается и/или завершается в результате выполнения другого Д-оператора  $(D_j)O_j(D'_j)$ , где  $j > i$ . Таким образом, в последнем случае обработка данных осуществляется некоторой последовательностью Д-операторов, в которой результат выполнения  $(D_i)O_i(D'_i)$  подвергается дальнейшей обработке Д-оператором  $(D_j)O_j(D'_j)$  и может продол-

жаться несколькими следующими Д-операторами.

Не все и не всегда специфицируемые данные изменяются в результате обработки. Это обусловлено тем, что изменение данных зависит от некоторых условий, полученных в результате их анализа и “скрытых” внутри Д-оператора.

Кроме того, для обработки глобальных данных могут использоваться и, как правило, используются вспомогательные (локальные) данные, которые служат для преобразования входных глобальных данных в выходные, но сами таковыми не являются. То есть, производные Д-операторы обрабатывают и продуцируют как глобальные, так и локальные данные в любых сочетаниях. Источниками и приемниками (носителями) этих данных часто выступают внешние устройства (ВУ), в частности, внешняя память (файлы).

Из вышеизложенного очевидно, что решение указанной задачи не тривиально и предполагает изучение состава, структуры и свойств специфицируемых данных. Этому и посвящена данная работа, из дальнейшего изложения которой будет видно, что специфицируемые данные имеют достаточно сложный состав и структуру.

Очевидно, что спецификация данных должна быть выполнена таким образом, чтобы обеспечить корректность перехода от исходного Д-оператора к производным, т.е. возможность получения результата совпадающего с результатом, продуцируемым исходным Д-оператором. Поскольку на рассматриваемом уровне представления алгоритма решить задачу корректности такого перехода не представляется возможным, в данной работе рассматриваются один из её аспектов, а именно, контроль корректности спецификации данных.

### **Свойства данных в композиционных схемах алгоритмов**

Прежде чем приступить к изучению состава и структуры данных в КС, оговорим некоторые особенности обозначений, которые будем далее использовать.

В соответствии с определением 1, значения специфицируемых данных изменяются (могут изменяться) в ходе вычислительного процесса, а на некоторых его этапах могут быть не определены. Отсюда следует, что специфицируются, фактически, носители данных. Учитывая это в данной работе под теоретико-множественными операциями, определенными на множествах данных, будем понимать операции над их носителями. В частности, под соотношением  $D = D'$  будем понимать совпадение множеств носителей данных  $D$ ,  $D'$ . В том случае, когда будем говорить о равенстве данных, понимая под этим равенство как множества носителей, так и множества значений, будем использовать символ “ $\equiv$ ” ( $D \equiv D'$ ). В случае совпадения множеств носителей при несовпадении множеств значений – символ “ $\neq$ ” ( $D \neq D'$ ).

Теперь определим некоторые виды специфицируемых данных, которые в дальнейшем будем использовать.

**Определение 2.** Данные, специфицируемые на входе и выходе Д-оператора  $(D)O(D')$ , будем называть:  $D$  входными или обрабатываемыми,  $D'$  – выходными или продуцируемыми. Для множеств, специфицируемых данных, допустимо как соотношение  $(D \cap D') \neq \emptyset$  (в частности,  $D \subseteq D'$  и  $D' \subseteq D$ ), так и  $(D \cap D') = \emptyset$ . Обрабатываемые данные изменяются (могут изменяться) в результате обработки. Входные данные  $D^C \subseteq D$  такие, что  $D^C \cap D' = \emptyset$  и  $D^C$  не изменяется Д-оператором  $(D)O(D')$  ни при каких условиях и ни при каких значениях данных  $D$ , будем называть используемыми (в отличие от обрабатываемых).

**Определение 3.** Данные, носителями которых являются некоторые множества ВУ  $R$ , с которых хранимые значения переносятся (копируются) в память, будем обозначать  $D^R$  и называть вводимыми, и  $W$ , в которые хранимые значения переносятся (копируются) из памяти на ВУ, –  $D^W$  и выводимыми. Вводимые и выводимые данные специфицируются, соответ-

ственно, на входе  $D^R \subseteq D$  и выходе  $D^W \subseteq D'$  произвольного Д-оператора. Множества  $D^R$  и\или  $D^W$  могут быть пустыми ( $D^R = \emptyset$  и\или  $D^W = \emptyset$ ), т.е. вводимые и\или выводимые данные могут отсутствовать.

Общий случай спецификации данных в КС определим исходя из вышеприведенных аспектов обработки данных и определения 2.

**Определение 4.** Глобальными для КС  $(D)O(D') = (D_1)O_1(D'_1) * (D_2)O_2(D'_2) * \dots * (D_k)O_k(D'_k)$ , являются данные  $D^G = D \cup D'$ , специфицированные на входе и выходе исходного Д-оператора. На входе и выходе любого производного Д-оператора  $(D_i)O_i(D'_i)$ , специфицируются как локальные, так и глобальные данные, то есть

$$D_i = D_i^G \cup D_i^L, D'_i = D_i'^G \cup D_i'^L,$$

где  $D_i^G, D_i'^G \subseteq D^G$ ,  $D_i^L, D_i'^L \subseteq D^L$ .

Глобальные данные такие, что  $(D_1^G \cup \dots \cup D_k^G) \subseteq D$ ,  $(D_1'^G \cup \dots \cup D_k'^G) \subseteq D'$ , а локальные – такие, что

$$D^L = (D_1^L \cup D_1'^L \cup \dots \cup D_k^L \cup D_k'^L).$$

При этом локальные данные инкапсулированы в КС, то есть определяются в данной КС, невидимы и недоступны в других КС данного уровня и на предшествующих уровнях описания алгоритма, и  $D^L \cap D^G = \emptyset$ .

По поводу данных, введенных в определении 4, отметим следующее.

Исходный Д-оператор на предшествующем уровне детализации алгоритма рассматривался как производный, а производные Д-операторы на следующем уровне детализации будут играть роль исходных. При этом, глобальные данные, являющиеся таковыми на данном уровне описания алгоритма, на предшествующем уровне являлись локальными. Локальные данные на следующем уровне детализации алгоритма играют роль глобальных.

Исходя из того, что, в соответствии с определением 3, данные могут являться или содержать вводимые и выводимые

данные, определим спецификации вводимых и выводимых данных с учетом определения 4.

**Определение 5.** Вводимые  $D^R = D^R \cup D^r$  и выводимые  $D^W = D^W \cup D^w$  данные делятся на глобальные  $(D^R \cup D^W) \subseteq D^G$  и локальные  $(D^r \cup D^w) \subseteq D^L$ . Глобальные, специфицируемые на входе и выходе как исходного  $D^R \subseteq D$ ,  $D^W \subseteq D'$ , так и любого производного Д-оператора

$$D_i^R \subseteq D_i, D_i^W \subseteq D_i'$$

такие, что  $(D_1^R \cup \dots \cup D_k^R) = D^R$  и  $(D_1^W \cup \dots \cup D_k^W) = D^W$ . Локальные  $D_i^r, D_i^w \subseteq D^L$ , специфицируемые только на входе и выходе производных Д-операторов  $D_i^r \subseteq D_i$ , такие, что  $(D_1^r \cup \dots \cup D_k^r) = D^r$ ,  $(D_1^w \cup \dots \cup D_k^w) = D^w$ .

Исходя из того, что Д-операторы взаимодействуют в процессе обработки данных, определим понятие информационной связи.

**Определение 6.** В КС Д-операторы  $(D_i)O_i(D_i')$  и  $(D_j)O_j(D_j')$  ( $j > i$ ) назовем информационно связанными (в дальнейшем связанными), если для специфицированных у них данных выполняется соотношение  $D_i' \cap D_j \neq \emptyset$ , в противном случае эти Д-операторы не связаны. Данные  ${}_i D_j = (D_i' \cap D_j)$  назовем связывающими, если при обозначении их на выходе  $i$ -го Д-оператора  ${}_i \bar{D}_j$ , а на входе  $j$ -го –  ${}_i \bar{D}_j$ , выполняется соотношение  ${}_i \bar{D}_j \equiv {}_i \bar{D}_j$ . Если это соотношение не выполняется, то  ${}_i D_j$  не являются связывающими данными. Данные, связывающие Д-оператор  $(D_i)O_i(D_i')$  со всеми предшествующими  ${}_1 D_i^{CB} = {}_1 D_i \cup \dots \cup {}_{i-1} D_i$  и всеми последующими  ${}_i D_k^{CB} = {}_i D_{i+1} \cup \dots \cup {}_i D_k$  Д-операторами, назовем связывающими Д-оператор  $(D_i)O_i(D_i')$ , соответственно, слева и справа.

ва. Данные  ${}_1 D_k^{CB} = {}_1 D_2 \cup \dots \cup {}_{k-1} D_k$  назовем связывающими Д-операторы в КС.

Отметим, что в данной работе главным образом рассматриваются локальные связи, т.е. связи внутри КС.

Теперь, определим множества глобальных данных, обрабатываемых производными Д-операторами автономно. Эти данные назовем собственными.

**Определение 7.** Собственные данные  $D_i^{CB} \subseteq D_i$  и  $D_i'^{CB} \subseteq D_i'$ , специфицированные на входе и выходе любого  $i$ -го Д-оператора такие, что  $D \supseteq D_i^{CB} \equiv D_i^{CB} \subseteq D_i$ ,  $D' \supseteq D_i'^{CB} \equiv D_i'^{CB} \subseteq D_i'$  и  $(D_1^{CB} \cup \dots \cup D_k^{CB}) = D^{CB} \subseteq D$ ,  $(D_1'^{CB} \cup \dots \cup D_k'^{CB}) = D'^{CB} \subseteq D'$ .

По поводу определения 7 отметим, что данные  $D$  и  $D'$  на предшествующем уровне детализации алгоритма включали данные, связывающие исходный Д-оператор  $(D)O(D')$  с другими Д-операторами этого предшествующего уровня. При переходе к текущему уровню описания алгоритма все Д-операторы предшествующего уровня детализуются и представляют собой композиции производных Д-операторов с сохранением вышеупомянутых связей. Функцию связывающих производные Д-операторы данной КС с производными Д-операторами других КС на данном уровне детализации алгоритма выполняют собственные данные  $D_i^{CB}$  и  $D_i'^{CB}$ . Спецификации собственных данных, таким образом, позволяет специфицировать глобальные связи между всеми КС, описывающими алгоритм на данном уровне его детализации. В данной работе глобальные связи детально не рассматриваются.

Будем полагать, исходя из определений 4 – 7, что данные, специфицируемые на входе и выходе Д-оператора, представляют собой семейства множеств. Исходный Д-оператор при этом записывается в виде:

$$(D, D^R)O(D', D^W), \quad (1)$$

а производный – в виде:

$$(D_i^R, D_i^{CB}, D_i^{CB})O_i(D_i^W, D_i^{CB}, D_k^{CB}).$$

Для того, чтобы обеспечить упомянутое во введении требование об отсутствии дублирования в спецификациях данных, необходимо эти данные детализовать таким образом, чтобы для любых множеств данных, полученных в результате детализации и специфицируемых на входе и выходе любого производного  $i$ -го Д-оператора, выполнялось  $(D_i^b \cap D_i^a) = \emptyset$ .

Для детализации состава данных, специфицируемых на входе и выходе  $i$ -го производного Д-оператора, покажем наличие некоторых их свойств, предварительно записав производный Д-оператор с учетом определения 5, в виде:

$$(D_i^R, D_i^r, D_i^{CB}, D_i^{CB})O_i(D_i^W, D_i^w, D_i^{CB}, D_k^{CB}). \quad (2)$$

Начнем рассмотрение с выводимых данных, для общего случая которых будем утверждать следующее.

**Утверждение 1.** Выводимые данные  $D_i^W, D_i^w \subseteq D_i'$  включают подмножества  ${}_i D_k^{W-CB} \subseteq D_i^W, {}_i D_k^{w-CB} \subseteq D_i^w$ , которые входят в состав данных, связывающих Д-оператор  $(D_i)O_i(D_i')$  справа

$${}_i D_k^{W-CB}, {}_i D_k^{w-CB} \subseteq {}_i D_k^{CB}.$$

*Доказательство.* Если для данных, специфицированных на входе любого Д-оператора  $(D_s)O_s(D_s')$  ( $s > i$ ), выполняется  $(D_i^W \cap D_s) = {}_i D_s^{W-CB}$  и  $(D_i^w \cap D_s) = {}_i D_s^{w-CB}$ , то, поскольку, в соответствии с определением 3,  $D_i^W$  и  $D_i^w$  копируются из памяти на ВУ и, таким образом, их значения не изменяются, то есть выполняется  ${}_i \bar{D}_s^{W-CB} \equiv {}_i \bar{D}_s^{W-CB}, {}_i \bar{D}_s^{w-CB} \equiv {}_i \bar{D}_s^{w-CB}$ , в соответствии с определением 6, эти данные являются связывающими.

Так как очевидно, что приведенные рассуждения справедливы для всех  $s > i$ , то

$${}_i D_k^{W-CB} = ({}_i D_{i+1}^{W-CB} \cup \dots \cup {}_i D_s^{W-CB} \cup \dots \cup {}_i D_k^{W-CB})$$

и

$${}_i D_k^{w-CB} = ({}_i D_{i+1}^{w-CB} \cup \dots \cup {}_i D_s^{w-CB} \cup \dots \cup {}_i D_k^{w-CB}),$$

в соответствии с определением 6, входят в состав данных, связывающих Д-оператор  $(D_i)O_i(D_i')$  справа  ${}_i D_k^{W-CB}, {}_i D_k^{w-CB} \subseteq {}_i D_k^{CB}$ .

Утверждение доказано.

**Следствие.** Из соотношения

$${}_i D_k^{CB} = {}_i D_{i+1} \cup \dots \cup {}_i D_k$$

(см. определение 6) следует, что для любых  $i$  и  $j$  выполняется  ${}_i D_j^{W-CB}, {}_i D_j^{w-CB} \subseteq {}_i D_j$ .

Теперь покажем наличие некоторых свойств, которыми в общем случае обладают собственные данные.

**Утверждение 2.** Собственные данные  $D_i^{CB} \subseteq D_i'$  включают подмножество данных  ${}_i D_k^{CB-CB} \subseteq D_i^{CB}$ , входящие в состав связывающих Д-оператор  $(D_i)O_i(D_i')$  справа  ${}_i D_k^{CB-CB} \subseteq {}_i D_k^{CB}$  при условии, что эти данные являются используемыми. При этом, данные  ${}_i D_s^{CB-CB} \subseteq {}_i D_k^{CB-CB}$  ( $s > i$ ) связывают  $i$ -ый Д-оператор с любыми следующими за ним Д-операторами, на входе которых эти данные специфицированы.

*Доказательство.* Пусть для данных, специфицированных на входе Д-оператора  $(D_s)O_s(D_s')$  ( $s > i$ ), выполняется

$${}_i D_s^{CB-CB} = D_i^{CB} \cap D_s.$$

Поскольку  ${}_i D_s^{CB-CB}$ , в соответствии с условием утверждения, используемые данные, т.е. в соответствии с определением 2, они не изменяются никаким Д-оператором, то выполняется

$$D_s \supseteq {}_i \bar{D}_s^{CB-CB} \equiv {}_i \bar{D}_s^{CB-CB} \subseteq D_i^{CB}$$

и, в соответствии с определением 6, эти данные являются связывающими.

Очевидно, что приведенные рассуждения справедливы для любого  $s > i$ , откуда следует

$$({}_i D_{i+1}^{CB-CB} \cup \dots \cup {}_i D_s^{CB-CB} \cup \dots \cup {}_i D_k^{CB-CB}) = {}_i D_k^{CB-CB}.$$

В соответствии с определением 6, данные  ${}_i D_k^{CB-CB}$  входят в состав связывающих Д-оператор  $(D_i)O_i(D_i')$  справа  ${}_i D_k^{CB-CB} \subseteq {}_i D_k^{CB}$ .

Теперь предположим, что на входе  $(D_p)O_p(D'_p)$  и  $(D_s)O_s(D'_s)$  специфицированы одни и те же данные, связывающие эти Д-операторы с  $(D_i)O_i(D'_i)$ . То есть  ${}_iD_p^{CB-cs}, {}_iD_s^{CB-cs} \subseteq {}_iD_k^{CB-CB}$  и  ${}_iD_p^{CB-cs} = {}_iD_s^{CB-cs}$  при  $i < p < s$ . Поскольку из определения 6 известно, что  ${}_i\bar{D}_p^{CB-cs} \equiv {}_i\bar{D}_p^{CB-cs}$ , а по условию утверждения  ${}_iD_p^{CB-cs}$  – используемые данные, то из  ${}_iD_p^{CB-cs} = {}_iD_s^{CB-cs}$  следует,  ${}_i\bar{D}_p^{CB-cs} \equiv {}_i\bar{D}_s^{CB-cs}$ . Откуда, в свою очередь, следует, что  $i$ -ый Д-оператор, на выходе которого данные обозначены  ${}_iD_{p,s}^{CB-cs}$ , связан с  $p$ -ым и  $s$ -ым

$${}_i\bar{D}_{p,s}^{CB-cs} \equiv {}_i\bar{D}_p^{CB-cs} \equiv {}_i\bar{D}_s^{CB-cs}.$$

Очевидно, что полученный результат легко обобщить на случай, когда данные  ${}_iD_{i+1,\dots,k}^{CB-cs}$  связывают  $i$ -ый Д-оператор с любыми (в частности, со всеми) следующими за ним Д-операторами, на входах которых эти данные специфицированы.

Утверждение доказано.

**Следствие.** Из соотношения  ${}_iD_k^{CB} = {}_iD_{i+1} \cup \dots \cup {}_iD_k$  (см. определение 6) следует, что для любых  $i$  и  $j$  выполняется  ${}_iD_j^{CB-cs} \subseteq {}_iD_j$ .

Отметим, что данные  ${}_iD_k^{CB-CB}$  играют как роль собственных (глобальных связывающих) данных, так и локальных связывающих данных.

Для дальнейшей детализации данных на выходе Д-оператора покажем наличие ещё одного свойства выводимых данных.

**Утверждение 3.** Глобальные выводимые данные Д-оператора  $(D_i)O_i(D'_i)$  входят в состав собственных  $(D_i^W \cap D_i^{CB}) \neq \emptyset$ .

Доказательство аналогично доказательству утверждения 1 с тем отличием, что собственные данные  $D_i^{CB}$  рассматриваются как глобальные связывающие дан-

ные (о чем выше упоминалось), а  $(D_i^W \cap D_i^{CB}) = D_i^{W-CB}$ .

**Следствие.** Из утверждения 3 с учетом утверждений 1, 2 и следствий из них легко увидеть, что имеют место глобальные данные, являющиеся выводимыми, собственными и связывающими  $D_i^{W-CB-CB} = (D_i^W \cap D_i^{CB} \cap {}_iD_k^{CB})$  и для любых  $i$  и  $j$  выполняется  ${}_iD_j^{W-CB-cs} \subseteq {}_iD_j$ .

Для исключения дублирования специфицируемых данных, исходя из утверждений 1 – 3, введем следующие их множества:

$$D_i^{W'} = (D_i^W \setminus ({}_iD_k^{CB} \cup D_i^{CB})), \quad (3)$$

$$D_i^{w'} = (D_i^{w'} \setminus {}_iD_k^{CB}), \quad (4)$$

$$D_i^{CB'} = (D_i^{CB} \setminus (D_i^W \cup {}_iD_k^{CB})), \quad (5)$$

а из следствий из этих утверждений множество

$${}_iD_k^{CB'} = ({}_iD_{i+1} \cup \dots \cup {}_iD_k), \quad (6)$$

где

$${}_iD_j' = ({}_iD_j \setminus ({}_iD_j^{CB-cs} \cup {}_iD_j^{W-cs} \cup {}_iD_j^{w-cs} \cup {}_iD_j^{W-CB-cs}))$$

( $i < j$ ).

В соответствии с утверждениями 1 – 3, учетом (3) – (6), данные, специфицируемые на выходе  $i$ -го Д-оператора, представим в виде следующего семейства множеств:

$$D_i' = \{D_i^{W'}, D_i^{w'}, D_i^{CB'}, {}_iD_k^{W-CB}, {}_iD_k^{CB-CB}, {}_iD_k^{W-CB}, {}_iD_k^{w-CB}, {}_iD_j^{W-CB-CB}, {}_iD_k^{CB'}\}, \quad (7)$$

а данные, связывающие Д-оператор справа, представим в виде:

$${}_iD_k^{CB} = ({}_iD_k^{CB} \cup {}_iD_k^{CB-CB} \cup {}_iD_k^{W-CB} \cup {}_iD_k^{w-CB} \cup {}_iD_j^{W-CB-CB}). \quad (8)$$

Полученный результат позволяет перейти к рассмотрению данных на входе  $i$ -го производного Д-оператора по поводу которых будем утверждать следующее.

**Утверждение 4.** Данные  ${}_1D_i^{CB}$ , связывающие  $i$ -ый Д-оператор слева, включают множество связывающих данных

$${}_1D_i^{W-CB}, {}_1D_i^{w-CB}, {}_1D_i^{CB-CB}, {}_iD_j^{W-CB-CB}.$$

*Доказательство.* В соответствии с (8) и следствиями из утверждений 1 – 3, на выходе Д-оператора  $(D_p)O_p(D'_p)$  ( $p < i$ ) присутствует множество данных  ${}_pD_i^{W-cb}$ ,  ${}_pD_i^{w-cb}$ ,  ${}_pD_i^{CB-cb}$ ,  ${}_iD_j^{W-CB-cb} \subseteq {}_pD_k^{CB}$ ,

связывающих этот Д-оператор справа. Исходя из определения 6, легко увидеть, что при  $p < i$   ${}_pD_k^{CB} \cap {}_1D_i^{CB} \neq \emptyset$  и  ${}_pD_i^{W-cb}$ ,  ${}_pD_i^{w-cb}$ ,  ${}_pD_i^{CB-cb}$ ,  ${}_pD_i^{W-CB-cb} \subseteq {}_1D_i^{CB}$ .

Если рассмотреть все Д-операторы, для которых выполняется  $p < i$ , получаем:

$$\left( {}_1D_i^{W-cb} \cup \dots \cup {}_pD_i^{W-cb} \cup \dots \cup {}_{i-1}D_i^{W-cb} \right) = {}_1D_i^{W-CB},$$

$$\left( {}_1D_i^{w-cb} \cup \dots \cup {}_pD_i^{w-cb} \cup \dots \cup {}_{i-1}D_i^{w-cb} \right) = {}_1D_i^{w-CB},$$

$$\left( {}_1D_i^{CB-cb} \cup \dots \cup {}_pD_i^{CB-cb} \cup \dots \cup {}_{i-1}D_i^{CB-cb} \right) = {}_1D_i^{CB-CB},$$

$$\left( {}_1D_i^{W-CB-cb} \cup \dots \cup {}_pD_i^{W-CB-cb} \cup \dots \cup {}_{i-1}D_i^{W-CB-cb} \right) = {}_1D_i^{W-CB-CB}.$$

Поскольку, в соответствии со следствиями из утверждений 1 – 3, для любых  $i$  и  $j$  выполняется

$${}_iD_j^{W-cb}, {}_iD_j^{w-cb}, {}_iD_j^{CB-cb}, {}_iD_j^{W-CB-cb} \subseteq {}_iD_j$$

то, очевидно,

$${}_1D_i^{W-CB}, {}_1D_i^{w-CB}, {}_1D_i^{CB-CB}, {}_1D_i^{W-CB-CB} \subseteq {}_1D_i^{CB}.$$

Утверждение доказано.

Чтобы исключить дублирование специфицируемых данных, исходя из утверждений 1 – 3 и следствий из них, введем следующее их множество:

$${}_1D_i^{CB} = ({}_1D'_i \cup \dots \cup {}_{i-1}D'_i),$$

где

$${}_pD'_i = \left( {}_pD_i \setminus \left( {}_pD_i^{CB-cb} \cup {}_pD_i^{W-cb} \cup {}_pD_i^{w-cb} \cup {}_pD_i^{W-CB-cb} \right) \right).$$

В соответствии с утверждением 4, данные, связывающие  $i$ -ый Д-оператор слева, представим в виде:

$${}_1D_i^{CB} = \left( {}_1D'_i{}^{CB} \cup {}_1D'_i{}^{CB-CB} \cup {}_1D'_i{}^{W-CB} \cup {}_1D'_i{}^{w-CB} \cup {}_1D'_i{}^{W-CB-CB} \right). \quad (9)$$

В соответствии с (2) и (9), данные, специфицируемые на входе  $i$ -го Д-оператора, представим в виде следующего семейства множеств:

$$D_i = \left\{ D_i^R, D_i^r, D_i^{CB}, {}_1D_i^{CB-CB}, {}_1D_i^{W-CB}, {}_1D_i^{w-CB}, {}_1D_i^{W-CB-CB} \right\}. \quad (10)$$

Исходя из соотношений (7), (10), запишем Д-оператор, со специфицированными на входе и выходе данными,

$$\left( D_i^R, D_i^r, D_i^{CB}, {}_1D_i^{CB-CB}, {}_1D_i^{W-CB}, {}_1D_i^{w-CB}, {}_1D_i^{W-CB-CB} \right) O_i \left( D_i^W, D_i^w, D_i^{CB}, D_i^{W-CB}, {}_iD_k^{CB-CB}, {}_iD_k^{W-CB}, {}_iD_k^{w-CB}, {}_iD_k^{W-CB-CB}, {}_iD_k^{CB} \right). \quad (11)$$

При этом по построению и в соответствии с определениями 4 – 7, для специфицируемых данных выполняются соотношения:

$$\left( D_i^R \cap D_i^r \cap D_i^{CB} \cap {}_1D_i^{CB-CB} \cap {}_1D_i^{W-CB} \cap {}_1D_i^{w-CB} \cap {}_1D_i^{W-CB-CB} \cap {}_1D_i^{CB} \right) = \emptyset,$$

$$\left( D_i^W \cap D_i^w \cap D_i^{CB} \cap D_i^{W-CB} \cap {}_iD_k^{CB-CB} \cap {}_iD_k^{W-CB} \cap {}_iD_k^{w-CB} \cap {}_iD_k^{W-CB-CB} \cap {}_iD_k^{CB} \right) = \emptyset,$$

т.е. все данные специфицируются однократно (не дублируются).

В итоге покажем наличие особенностей в спецификации данных для первого и последнего в КС Д-операторов.

**Утверждение 5.** На входе первого  $(D_1)O_1(D'_1)$  и последнего  $(D_k)O_k(D'_k)$  в КС Д-операторов спецификации связывающих данных отсутствуют.

Доказательство очевидно и вытекает из определения 6, так как для первого Д-оператора отсутствуют предшествующие, а для последнего последующие Д-операторы.

Исходя из (1), (2), (11) и утверждения 5 запишем КС в виде:

$$\begin{aligned}
 & (D, D^R)O(D', D^W) = \\
 & = (D_1^R, D_1^r, D_1^{CB})O_1(D_1^W, D_1^w, D_1^{rCB}, D_1^{w-CB}, \\
 & \quad , {}_1D_k^{CB-CB}, {}_1D_k^{w-CB}, {}_1D_k^{rCB}, {}_1D_k^{CB})^* \\
 & \dots\dots\dots \\
 & * (D_i^R, D_i^r, D_i^{CB}, {}_1D_i^{CB-CB}, {}_1D_i^{w-CB}, {}_1D_i^{w-CB}, \\
 & \quad , {}_1D_i^{rCB})O_i(D_i^W, D_i^w, D_i^{rCB}, D_i^{w-CB}, {}_iD_k^{CB-CB}, \\
 & \quad , {}_iD_k^{w-CB}, {}_iD_k^{w-CB}, {}_iD_k^{rCB})^* \\
 & \dots\dots\dots \\
 & * (D_k^R, D_k^r, D_k^{CB}, {}_1D_k^{CB-CB}, {}_1D_k^{w-CB}, {}_1D_k^{w-CB}, \\
 & \quad , {}_1D_k^{rCB})O_k(D_k^W, D_k^w, D_k^{CB}, D_k^{w-CB}).
 \end{aligned} \tag{12}$$

В результате выполненных построений показана возможность спецификации данных на входах и выходах производных Д-операторов, входящий в КС, соответствующей сформулированной задаче.

Кроме того, могут быть специфицированы связи между Д-операторами. В этом случае, рассматривая связывающие данные, в соответствии с определением 6, выражение (12) запишем в виде:

$$\begin{aligned}
 & (D, D^R)O(D', D^W) = \\
 & = (D_1^R, D_1^r, D_1^{CB})O_1(D_1^W, D_1^w, D_1^{rCB}, D_1^{w-CB}, \\
 & \quad , {}_1\bar{D}_2, \dots, {}_1\bar{D}_i, \dots, {}_1\bar{D}_k)^* \\
 & \dots\dots\dots \\
 & * (D_i^R, D_i^r, D_i^{CB}, {}_1\bar{D}_i, \dots, {}_{i-1}\bar{D}_i)O_i(D_i^W, D_i^w, D_i^{rCB}, \\
 & \quad , D_i^{w-CB}, {}_i\bar{D}_{i+1}, \dots, {}_i\bar{D}_k)^* \\
 & \dots\dots\dots \\
 & * (D_k^R, D_k^r, D_k^{CB}, {}_1\bar{D}_k, \dots, {}_{k-1}\bar{D}_k)O_k(D_k^W, D_k^w, \\
 & \quad , D_k^{CB}, D_k^{w-CB})
 \end{aligned}$$

Отметим, что спецификации данных, в частности, связывающих могут иметь различную степень детализации, что легко увидеть из изложенного.

Помимо вышеприведенных свойств специфицируемых данных имеют место такие свойства, которые ограничивают возможности спецификаций. Перейдем к рассмотрению данных свойств, которое начнем с вводимых и выводимых данных.

**Утверждение 6.** Вводимые данные не могут быть связывающими  $D^R \cap_1 D_k^{CB} = \emptyset$ .

Доказательство построим от противного и рассмотрим некоторые связывающие данные  ${}_p D_{i \subseteq 1} D_i^{CB}$ , продуцируемые Д-оператором  $(D_p)O_p(D'_p)$  ( $p < i$ ), для которых, в соответствии с определением 6, выполняется соотношение  ${}_p \bar{D}_i \equiv_p \bar{D}_i$ . Предположим, что  $(D_i^R \cap_p D_i) \neq \emptyset$  и  $(D_i^r \cap_p D_i) \neq \emptyset$ . Поскольку, в соответствии с определением 3, данные  $D_i^R$  копируются с ВУ в память, значения подмножества данных  $D_i^R \cap_p D_i$ , продуцируемых Д-оператором  $(D_p)O_p(D'_p)$ , будут изменены. То есть, при сделанном предположении приходим к противоречию, так как получаем  ${}_p \bar{D}_i \not\equiv_p \bar{D}_i$  и подмножество данных  $D_i^R \cap_p D_i$ , в соответствии с определением 6, не являющихся связывающими.

Аналогично рассуждая, к такому же противоречию придем в случае локальных вводимых данных  $D_i^r$ .

В результате сделанные предположения недопустимы, т.е.  $(D_i^R \cap_p D_i) = \emptyset$  и  $(D_i^r \cap_p D_i) = \emptyset$ .

Очевидно, что, таким же образом рассуждая, аналогичный результат получим для всех связывающих данных  ${}_j D_{i \subseteq 1} D_i^{CB}$  при любых  $j$  и  $i$  ( $j < i$ ). Таким образом,  $D^R \cap_1 D_i^{CB} = \emptyset$ ,  $D^r \cap_1 D_i^{CB} = \emptyset$  и  $D^R \cap_1 D_i^{CB} = \emptyset$ , т.е. вводимые данные не бывают связывающими слева.

Поскольку, в соответствии с определением 5, вводимые данные не специфицируются на выходе Д-оператора, то они для любого Д-оператора не бывают связывающими справа  $D^R \cap_i D_k^{CB} = \emptyset$ , откуда следует, что  $D^R \cap_1 D_k^{CB} = \emptyset$ .

Утверждение доказано.

**Утверждение 7.** Вводимые данные не могут быть собственными  $D^R \cap D^{CB} = \emptyset$ .

*Доказательство.* Если предположить, что  $D_i^R \cap D_i^{CB} \neq \emptyset$ , то, поскольку, в соответствии с определением 3, данные  $D_i^R$  копируются с ВУ в память, значения подмножества данных  $D_i^R \cap D_i^{CB}$  будет изменено. В результате чего получим противоречие с определением 7, так как в этом случае  $D \supseteq D_i^{CB} \neq D_i^{CB} \subseteq D_i$ . К такому же противоречию придем при всех  $i$ , и, таким образом,  $D^R \cap D^{CB} = \emptyset$ .

Утверждение доказано.

**Утверждение 8.** Локальные выводимые данные не могут быть собственными  $D^w \cap D'^{CB} = \emptyset$ .

*Доказательство* очевидно, так как  $D^w \subseteq D^L$ ,  $D'^{CB} \subseteq D^G$  и  $D^G \cap D^L = \emptyset$  (см. определения 4, 5, 7).

Теперь остановимся на локальных данных.

**Утверждение 9.** Локальные данные бывают только вводимыми, выводимыми и связывающими.

*Доказательство* очевидно и следует из (11) и утверждения 8.

**Утверждение 10.** Любые связывающие данные, за исключением  ${}_i D_k^{CB-CB}$ , не могут связывать  $i$ -ый Д-оператор более чем с одним следующим за ним Д-оператором.

*Доказательство.*  ${}_i D_k^{CB-CB}$  является исключением, в соответствии с утверждением 2.

Предположим, что для связывающих данных  ${}_i \bar{D}_p, {}_i \bar{D}_s$  ( $i < p < s$ ), специфицированных на входах соответствующих Д-операторов, выполняется соотношение  ${}_i \bar{D}_p = {}_i \bar{D}_s$ , т.е. специфицируются одни и те же данные.  ${}_i \bar{D}_p$  специфицированы на входе  $p$ -го Д-оператора и для них в соответствии с определением 6 выполняется  ${}_i \bar{D}_p \equiv {}_i \bar{D}_p$ . Поскольку, в соответствии с

определением 2,  ${}_i \bar{D}_p$  обрабатываемые и могут изменяться, то, в случае такого изменения,  ${}_i \bar{D}_p \neq {}_i \bar{D}_s$ , откуда следует,  ${}_i \bar{D}_p \neq {}_i \bar{D}_s$  и, в соответствии с определением 6,  ${}_i \bar{D}_s$  не будут данными, связывающими  $i$ -ый и  $s$ -ый Д-операторы. Таким образом, пришли к противоречию, к которому, очевидно, будем приходить при всех  $s > p$ . К тому же противоречию, очевидно, придем при рассмотрении всех остальных (см. (11)) связывающих данных

$${}_1 D_i^{CB-CB}, {}_1 D_i^{W-CB}, {}_1 D_i^{W-CB}, {}_1 D_i^{W-CB-CB}.$$

Поскольку во всех возможных случаях связывающих данных приходим к противоречию, то сделанное предположение не соответствует действительности и утверждение доказано.

Полученные в утверждениях 6 – 10 свойства специфицируемых данных будем трактовать как ограничения на спецификации.

## Заключение

В процессе выполнения данной работы рассматривался общий случай спецификации данных на входах и выходах производных Д-операторов, входящих в КС. Результатом является возможность записывать КС алгоритмов со специфицированными данными. Наличие спецификаций позволяет определить функциональность каждого Д-оператора и осуществить их декомпозицию, т.е. позволяет выполнить переход к следующему уровню представления алгоритма. При этом детализация рассматриваемых данных доведена до такой степени, при которой спецификации данных не дублируются. Иначе говоря, сформулированная задача решена с учетом специфики обработки данных.

Выше отмечено, что спецификация данных может осуществляться с различной степенью детализации, причем данная степень может задаваться в зависимости от решаемой задачи. Кроме того, весьма существенным результатом является возможность спецификации информационных

связей между Д-операторами, которые так же могут быть детализованы.

Что касается корректности выполнения спецификации данных, то для контроля над ней могут использоваться как проверка соответствия свойствам, заданным определениями 1 – 7 и утверждениями 1 – 5, так и проверка удовлетворения ограничений, заданных утверждениями 6 – 10.

Дальнейшее изучение свойств специфицируемых данных, в частности, дальнейшая их детализация, распространение полученных результатов на описание алгоритмов, выполненное в виде совокупности КС, является направлением будущих исследований.

1. *Данные в языках программирования: абстракция и типология.* Сб. статей / Под ред. В. Агафонова. – М.: Мир, 1982. – 328 с.
2. *Турский В.* Методология программирования. – М.: Мир, 1981. – 264 с.
3. *Шнейдерман Б.* Психология программирования: человеческие факторы в вычислительных и информационных системах. – М.: Радио и связь, 1984. – 304 с.
4. *Bastani F.B., Iyengar S.S.* The effect of data structures on the logical complexity of programs // *SACM.* – 1987. Vol. 30, N 3. – P. 250–259.
5. *Акуловский В.Г.* Основы алгебры алгоритмов, базирующейся на данных // *Проблемы програмування.* – 2010. – № 2–3. – С. 89–96.
6. *Дорошенко А.Е., Акуловский В.Г.* Алгебра алгоритмов с данными и прогнозирование вычислительного процесса // *Проблемы програмування.* – 2011. – № 3. – С. 3–10.
7. *Акуловский В.Г.* Алгебра для описания данных в композиционных схемах алгоритмов // *Проблемы програмування.* – 2012. – № 2–3. – С. 234–240.
8. *Дорошенко А.Е., Акуловский В.Г.* Нисходящее проектирование алгоритмов в рамках алгеброалгоритмического подхода // *Математические машины и системы.* – 2012. – № 3. – С. 97–102.

9. *Дорошенко А.Ю., Акуловський В.Г.* Вихідне проектування алгоритмів при алгеброалгоритмічному підході // *Вісник Київського національного університету імені Тараса Шевченка.* Серія фізико-математичні науки. – 2012. – Вип. 1. – С. 167–172.

Получено 26.07.2012

### **Об авторах:**

*Акуловский Валерий Григорьевич,*  
кандидат технических наук,  
доцент кафедры информационных систем  
и технологий,

*Дорошенко Анатолий Ефимович,*  
доктор физико-математических наук,  
профессор,  
заведующий отделом  
теории компьютерных вычислений.

### **Место работы авторов:**

Академия таможенной службы Украины,  
Тел. 050 941 05 66,  
E-mail: valeryakulovskiy@rambler.ru

Институт программных  
систем НАН Украины.  
Тел. (044) 526 3559,  
E-mail: dor@isofts.kiev.ua