

СТАТЬЯ

УДК 004.08

ИССЛЕДОВАНИЕ АРИФМЕТИЧЕСКИХ ПРОГРАММ

Попов С.В.

ООО «Научно-внедренческая фирма БП+», Москва, e-mail: s-v-popov@yandex.ru

Исследуются вычисления арифметических программ в зависимости от вида предметной области, над которыми вычисления осуществляются. Свойства предметных областей, определяющие сложность вычисления программ, описываются путем выделения так называемых разделимых означиваний префикса входной переменной программ. Каждое неполное означивание входной переменной программы в общем случае задает новую программу, вычисляемые значения которой зависят от вида окончательного продолжения означивания. Разделимые означивания префикса характеризуются различными спектрами возможных продолжений вычислений. Тем самым разнообразие разделимых означиваний префиксов входной переменной программы служит характеристикой разнообразия вычислений над предметной областью. Из этого следует нижняя оценка сложности максимального вычисления в результате произвольного окончательного означивания входной переменной. Для исследования свойств вычислений программ используется сведение программ к так называемым обобщенным формулам, представляющим функции, вычисляемые программами. Каждая такая формула строится по программе единственным образом и позволяет исследовать ее вычисления логическими средствами. Обобщенная формула строится над расширенным логическим базисом, который помимо обычных логических связей содержит бесконечные дизъюнкцию и конъюнкцию. Обобщенные формулы представляют собой полное множество в том смысле, что всякая вычисляемая функция представима обобщенной формулой.

**Ключевые слова:** арифметические программы, регулярные выражения, вычисления, сложность вычислений, обобщенные логические формулы

THE STUDY OF ARITHMETIC PROGRAMS

Popov S.V.

LLC «Nauchno-vnedrencheskaya firma BP+», Moscow, e-mail: s-v-popov@yandex.ru

The calculations of arithmetic programs depending on the type of the domain over which the calculations are carried out are investigated. The properties of the subject areas that determine the complexity of the calculation of programs are described by highlighting the so-called separable meanings of the prefix of the input variable of programs. Each incomplete denotation of the input variable of the program, in General, sets a new program, the calculated values of which depend on the type of the final continuation of the denotation. The separated values of the prefix are characterized by different spectra of possible extensions of calculations. Thus, the variety of separable meanings of prefixes of the input variable of the program is a characteristic of the variety of calculations over the subject area. This implies a lower bound on the complexity of the maximum computation as a result of arbitrary final signification of the input variable. To study the properties of program calculations, we use the reduction of programs to the so-called generalized formulas representing the functions calculated by programs. Each such formula is built according to the program in a unique way and allows to investigate its calculations by logical means. The generalized formula is constructed over an extended logical basis which, in addition to the usual logical bundles, contains infinite disjunction and conjunction. Generalized formulas are a complete set in the sense that every computable function is represented by a generalized formula.

**Keywords:** arithmetic programs, regular expressions, calculations, complexity of calculations, generalized logical formulas

Определение сложности вычислений актуально в связи с моделированием технологических, научных и социальных процессов и активным развитием искусственного интеллекта [1, 2]. Поэтому практически важно установление оценок сложности вычисления в зависимости от особенностей предметной области, над которой вычисление осуществляется [3, 4]. В статье рассматриваются арифметические двоичные программы над полным базисом  $v \leftarrow v + 1, v \leftarrow 0, v = u$  операторов, обозначаемые соответственно  $s(x), 0(x), E(v, u)$ . Каждая программа представляется оргграфом с арифметическими вершинами  $s(x), 0(x)$  и логическими ( $E$ ), обладает одной входной переменной  $x$  и конечным набором  $y_1, y_2, \dots, y_m$  рабочих, среди которых одна – вы-

ходная. При этом одна вершина – входная; одна – выходная; в арифметических вершинах используются только рабочие переменные; в логических могут использоваться как рабочие, так и входная. Все определения, необходимые для детального понимания содержания статьи, приведены в [5]. В силу ограниченного объема статьи здесь приведены лишь пояснения к некоторым определениям, чтобы сделать изложение понятнее.

Пусть  $\tau = v_1, v_2, \dots, v_l$  – путь в программе  $\pi$ . Заменим каждый его предикат  $P(z_1, z_2)$  предикатом  $P^\sigma(z_1, z_2)$ , где  $\sigma \in \{0, 1\}$  так, что следующая в  $\tau$  за  $P(z_1, z_2)$  вершина является его  $\sigma$ -последователем. Полученную последовательность назовем *размеченным путем* и обозначим  $\tau$ . Содержательно размеченный

путь представляет вычисление в программе. Если он начинается во входной вершине и заканчивается в выходной, то он представляет *полное вычисление* программы.

Рассуждения в настоящей статье базируются на результате Ю.И. Янова (Проблемы кибернетики, вып. 32), который показал, что каждую арифметическую программу можно представить в виде конечного автомата, состояния которого суть логические и заключительная вершины, а входной алфавит образован следующими «буквами». Если из логической вершины  $E$  в логическую вершину  $E_1$  ведет путь  $E, v_1, v_2, \dots, v_l, E_1$ , где  $v_1$  есть  $\sigma$ -последователь вершины  $E, v_1, v_2, \dots, v_l$  суть арифметические вершины и  $l \geq 0$ , то переход из состояния  $E$  в состояние  $E_1$  происходит под воздействием автоматной буквы  $E^\sigma, v_1, v_2, \dots, v_l$ . Представимое таким автоматом событие состоит из всех полных размеченных путей программы. Регулярное выражение  $R_\pi$ , представляющее множество всех полных путей, представимо в виде суммы  $R_1 \vee R_2 \vee \dots \vee R_r$  членов, не содержащих операции суммирования. Очевидно, что *каждое выражение  $R_i, i = 1, 2, \dots, r$  представляет только полные вычисления, а всякое его регулярное подвыражение определяет совокупность целочисленных функций.*

В [5] показано, как по регулярному выражению  $R_1 \vee R_2 \vee \dots \vee R_r$  построить обобщенную логическую формулу, представляющую функцию, вычисляемую программой  $\pi$ . Построение основывается на том, что все базисные операторы представимы обобщенными формулами. Построенная формула вместо переменных содержит так называемые *метAPERменные*, которые устанавливают соответствие между этой формулой и вычислениями. Дальнейшее содержание статьи состоит в исследовании свойств этой формулы.

Цель статьи состоит в установлении длины вычисления программы в зависимости от свойств предметной области, над которой вычисляется программа.

*Соотношение обобщенных формул, регулярных выражений и размеченных путей*

Построим *обобщенную ДНФ* (ОДНФ) по формуле, представляющей функцию, определяемой регулярным выражением  $R_\pi$ . Для этого отметим, что каждое регулярное выражение  $A^*$  определяет бесконечную дизъюнкцию. Если  $A$  не содержит оператора  $*$ , то соответствующая формула уже представлена в ОДНФ, с бесконечным числом конъюнктов, каждый из которых обладает конечной длиной. Если выражение  $A$  содержит оператор  $*$ , то индукцией по числу операторов  $*$  показывается, что определяемое им выра-

жение эквивалентно преобразуется в бесконечную дизъюнкцию конъюнктов конечной длины, атомарными формулами в которых служат формулы  $\mathbf{0}, \mathbf{S}$  и  $\mathbf{E}$ . Эту дизъюнкцию назовем обобщенной ДНФ.

Установим соответствие между ОДНФ и вычислениями программы, определяемыми размеченными путями. Индукцией по длине регулярного выражения доказывается теорема.

**Теорема 1.** *Пусть формула  $F$  определяется регулярным выражением. Тогда по ней единственным образом строится ОДНФ, в которой каждый конъюнкт представляет в точности одну функцию, определяемую единственным размеченным путем программы.*

Будем исследовать так называемые *частичные вычисления*, которые определяются означиванием лишь конечного префикса входной переменной программы. В результате такого означивания  $\sigma_x$  программа  $\pi(x)$  превращается в новую программу  $\pi'$ , а первоначальная формула  $F$ , построенная по регулярному выражению  $R_\pi$ , в новую формулу  $F'$ , представляющую функцию, вычисляемую программой  $\pi'$ . Преобразование формулы  $F$  в  $F'$  базируется на следующих эквивалентных преобразованиях.

Пусть обобщенная формула  $G$  построена по регулярному выражению. Введем так называемое *правильное означивание* метAPERменных формулы  $G$ . Допустим, что означен некоторый префикс входной переменной, а рабочие метAPERменные означиваются следующим образом.

1. Если в  $G$  встречается формула  $\mathbf{0}(\alpha)$ , где  $\alpha = \alpha_0, \dots, \alpha_m, \dots$ , то  $\alpha_0 = \dots = \alpha_m = \dots = 0$ . Очевидно, это единственное означивание метAPERменных, при котором формула  $\mathbf{0}(\alpha)$  истинная.

2. Если в  $G$  встречается формула  $S'(\alpha, \beta)$ ,  $\alpha = \alpha_0, \dots, \alpha_m, \dots, \beta = \beta_0, \dots, \beta_m, \dots$ , то  $\alpha, \beta$  – рабочие метAPERменные. Следовательно, имеется самая правая единица в означивании компонентов  $\alpha_0, \dots, \alpha_m, \dots$ , за которой следуют бесконечный нулевой суффикс. Тогда означивание компонентов  $\beta_0, \dots, \beta_m, \dots$  получается из двоичного представления числа  $|\alpha| + 1$ . Здесь  $|\alpha|$  – это двоичное число, определяемое значениями метAPERменных  $\alpha_0, \dots, \alpha_m, \dots$  отбрасыванием бесконечного нулевого суффикса правее последнего единичного компонента. При таком означивании метAPERменных  $\alpha, \beta$  формула  $S'$  истинная, при любом другом означивании метAPERменных  $\beta_0, \dots, \beta_m, \dots$  она ложна.

3. Если  $\alpha$  и  $\beta$  суть рабочие метAPERменные, то формула  $E(\alpha, \beta)$  истинная при означиваниях, соответственно  $\sigma_\alpha$  и  $\sigma_\beta$  лишь в случае, когда  $\sigma_\alpha = \sigma_\beta$ , т.е.  $\sigma_\alpha$  и  $\sigma_\beta$  суть рав-

ные двоичные числа. А формула  $E^{-1}(\alpha, \beta)$  истинная лишь в случае, когда  $\sigma_\alpha \neq \sigma_\beta$ , т.е.  $\sigma_\alpha$  и  $\sigma_\beta$  суть равные двоичные числа.

Пусть  $G(\alpha_1, \alpha_2, \dots, \alpha_q)$  есть конъюнкция формул, представляющих арифметические и логические операторы, содержащая только рабочие метапеременные  $\alpha_1, \alpha_2, \dots, \alpha_q$ . Тогда при означивании ее метапеременных она истинная тогда и только тогда, когда означивания *правильные*, т.е. удовлетворяют условиям из пунктов 1–3. Если конъюнкция  $G$  построена по регулярному выражению, определяемому размеченным путем, то значения ее метапеременных полностью соответствуют вычисленным значениям рабочих переменных. То есть вычисление, определяемое размеченным путем, совпадает с правильным означиванием, и наоборот, всякое правильное означивание совпадает с вычислением, определяемым размеченным путем. Такой вывод справедлив лишь при условии, когда размеченный путь не включает логической вершины, содержащей входную переменную программы. Рассмотрим, что происходит, если в размеченном пути встречаются логические операторы, содержащие входную переменную программы. Легко показать, что *формула  $E(\alpha, \beta)$  может быть либо ложной, либо неопределенной, а  $E^{-1}(\alpha, \beta)$  – либо истинной либо неопределенной. Формула  $E^\sigma(\alpha, \beta)$  после означивания ее метапеременных представляет собой конъюнкт или дизъюнкт литер, лишь когда она содержит частично означенную входную переменную и означенные префиксы метапеременных  $\alpha$  и  $\beta$  совпадают.* Таким образом, если логический оператор не содержит частично означенной переменной, то его логическое значение совпадает с логическим значением представляющей формулы. Но если логический оператор содержит частично означенную входную переменную, то не является константой, и его окончательное значение зависит от дальнейшего означивания входной переменной.

*Расширение означивания префикса входной переменной программы*

Естественно возникает вопрос, что происходит с формулой, которая строится по регулярному выражению  $R_n$ , если расширяется означиваемый префикс входной переменной. В [5] показано, что при расширении  $\alpha_2$  начального означивания  $\alpha_1$  входной переменной вычисление для  $\alpha_2$  включает вычисление для  $\alpha_1$ . При этом возможны следующие случаи.

1. Оно либо завершается с тем же результатом, что и при вычислении с означиванием  $\alpha_1$ . Это происходит в случае, когда

вычисление с начальным означиванием  $\alpha_1$  завершилось.

2. Оно может завершиться, хотя вычисление с начальным означиванием  $\alpha_1$  не завершилось.

3. Оно может не завершиться, при этом вычисление с означиванием  $\alpha_1$  также не завершается.

Действительно, увеличение длины означиваемого префикса входной переменной программы приводит к исключению ряда конъюнктов из ОДНФ, построенной по формуле, определяемой регулярным выражением.

**Пример 1.** Рассмотрим означивания префикса длины  $n > 0$  входной переменной  $x = x_0 x_1 x_2 x_3 \dots x_i \dots$  в формуле  $0(\alpha_0) (E(\alpha_0, x) E(\alpha_0, \mathbf{Out}) \vee \bigcup_{j=1, \omega} (\bigcap_{h=0, j-1} E(\alpha_h, x) S(\alpha_h, \alpha_{h+1})) E(\alpha_j, x) E(x, \mathbf{Out}))$ .

Нетривиальным значением эта формула обладает, когда формула  $0(\alpha_0)$  истинная, т.е. значение метапеременной  $\alpha_0$  нулевое.

*Базис индукции.* При  $n = 1$  возможны только два случая  $x_0 = 0 / x_0 = 1$ . При  $x_0 = 0$  формула  $E(\alpha_0, x) = \bigcap_{i=1, \omega} x_i$ . Выходная метапеременная  $\mathbf{Out}$  в этом случае нулевая. В оставшейся бесконечной дизъюнкции каждый конъюнктивный член содержит выражение  $E(\alpha_j, x)$ ,  $j = 1, 2, \dots$ . Каждый такой член ложный в том случае, когда  $j$  – нечетно. Следовательно, бесконечная дизъюнкция эквивалентно преобразуется в бесконечную дизъюнкцию, в которые входят выражения  $E(\alpha_j, x)$ , где  $j$  – четно. Выражение  $\alpha_j$  представляет число  $j$ . Следовательно, результирующая бесконечная дизъюнкция представляет предикат «число  $x$  – четно», и эквивалентно преобразуется в  $x_0$ . Случай, когда  $x_0 = 1$ , эквивалентно преобразует исходное выражение в формулу, представляющую предикат «число  $x$  – нечетно», представляемый формулой  $x_0$ . Итак, начальные означивания  $x_0 = 0 / x_0 = 1$  порождают две формулы:  $x_0$  и  $x_0$ .

*Индукционные рассуждения.* Если  $n > 0$ , то, рассуждая как в предыдущем случае, можно показать, что при означивании  $\sigma_x = \sigma_0 \sigma_1 \dots \sigma_{n-1}$  префикса переменной  $x$  исходное выражение эквивалентно преобразуется в формулу  $x_0^{\sigma_0} x_1^{\sigma_1} \dots x_{n-1}^{\sigma_{n-1}}$ , представляющую предикат «число  $x$  имеет своим префиксом  $\sigma_x$ ». Таким образом, все эти наборы образуют  $2^n$  классов эквивалентности.

**Пример 2.** Рассмотрим программу, вычисляющую функцию  $x + y$ , в которой  $u, v$  суть рабочие переменные, причем  $u$  – входная. Первая вершина есть функция присваивания  $u \leftarrow x$ , следующий за ней цикл осуществляет прибавление к переменной  $u$  еще  $y$  единиц. Таким образом, на выходе программы формируется сумма  $x + y$ . Регу-

лярное выражение, порожаемое этой программой, таково:  $0(v) E(u, x) (E(v, y) s(v) s(u))^* E(v, y)$ . Нетрудно убедиться, что для него справедливы приведенные выше рассуждения.

*Классы неполных означиваний  
входной переменной программы*

В этом разделе введем классификацию означиваний префиксов входной переменной программы. Справедлива следующая теорема.

**Теорема 2.** Пусть  $F(x)$  есть формула, представляющая функцию, вычисляемую программой  $\pi(x)$ , и означивания  $\alpha_1$  и  $\alpha_2$  префиксов входной переменной находятся в отношении  $\alpha_1 < \alpha_2$ . Тогда справедлива импликация  $F(\alpha_2) \supset F(\alpha_1)$ .

**Следствие.** Пусть  $\mathcal{S}_1$  есть множество вычислений, которые определяются программой  $\pi(\alpha_1)$  и  $\mathcal{S}_2$  – множество вычислений, которые определяются программой  $\pi(\alpha_2)$ . Тогда справедливо включение  $\mathcal{S}_2 \subseteq \mathcal{S}_1$ .

Таким образом, программа  $\pi(\alpha_1)$  вычисляет функцию, которая является обобщением функции, вычисляемой программой  $\pi(\alpha_2)$ .

Говорим, что логические формулы  $A$  и  $B$  находятся в отношении  $A \# B$ , если неверны обе импликации  $A \supset B$  и  $B \supset A$ .

**Теорема 3.** Пусть формула  $F(x)$  представляет функцию, вычисляемую программой  $\pi(x)$ , и  $\alpha_1, \alpha_2$  суть означивания префиксов входной переменной, причем формулы  $F(\alpha_1), F(\alpha_2)$  находятся в отношении  $F(\alpha_1) \# F(\alpha_2)$ . Тогда:

1) имеются размеченные пути  $t_1$  и  $t_2$ , такие, что  $t_1$  принадлежит программе  $\pi(\alpha_1)$  и не принадлежит программе  $\pi(\alpha_2)$ , а  $t_2$  – принадлежит программе  $\pi(\alpha_2)$  и не принадлежит  $\pi(\alpha_1)$ ;

2) имеются конъюнкты  $C_1$  и  $C_2$ , такие, что  $C_1$  принадлежит ОДНФ формулы  $F(\alpha_1)$  и не принадлежит ОДНФ формулы  $F(\alpha_2)$ , а  $C_2$  – принадлежит ОДНФ формулы  $F(\alpha_2)$  и не принадлежит ОДНФ формулы  $F(\alpha_1)$ .

**Доказательство.** Упомянутые конъюнкты  $C_1$  и  $C_2$  различаются подформулами  $E^\sigma$ , в которых один аргумент есть не полностью означенная входная переменная программы, а второй – рабочая метапеременная. Эти же конъюнкты определяют различные размеченные пути  $t_1$  и  $t_2$ .

*О длине вычисления программ*

В этом разделе покажем, что длина вычисления программы определяется ее свойствами образовывать различные вычисления при означивании префикса ее входной переменной. Введем такое определение.

Пусть  $\sigma'_x$  есть означивание префикса входной переменной программы  $\pi(x)$ . Тогда

двоичное число  $\sigma_x$  называется *окончательным продолжением* означивания  $\sigma'_x$ , если  $\sigma'_x \sigma_x$  есть двоичное число. Два означенных префикса  $\sigma'_x$  и  $\sigma''_x$  входной переменной программы  $\pi$  называются *разделимыми*, если имеется окончательное продолжение  $\sigma_x$  такое, что вычисления  $\pi(\sigma'_x \sigma_x)$  и  $\pi(\sigma''_x \sigma_x)$  различные.

**Теорема 4.** Пусть  $\sigma'_x$  и  $\sigma''_x$  суть два разделимых означивания и формула  $F(x)$  получена по регулярному выражению  $R_x$ . Тогда выполняется отношение  $F(\sigma'_x) \# F(\sigma''_x)$ .

**Доказательство.** Пусть окончательные означивания  $\sigma'_x \sigma_x$  и  $\sigma''_x \sigma_x$  определяют разные вычисления программы, т.е. им соответствуют разные размеченные пути. Тогда этим путям в ОДНФ формул  $F(\sigma'_x)$  и  $F(\sigma''_x)$  соответствуют две разные конъюнкции, которые истинны при различных означиваниях  $\sigma'_x \sigma_x$  и  $\sigma''_x \sigma_x$  входной переменной и при соответствующих правильных означиваниях метапеременных.

**Следствие.** Пусть формула  $F(x)$  получена по регулярному выражению  $R_x$ , и  $\{\sigma'_x, \sigma''_x, \dots, \sigma_x^{(m)}\}$  попарно разделимые означивания префикса входной переменной программы. Тогда формулы  $F(\sigma'_x), F(\sigma''_x), \dots, F(\sigma_x^{(m)})$  попарно находятся в отношении  $\#$ .

Справедлива теорема.

**Теорема 5.** Пусть  $\{\sigma'_x, \sigma''_x, \dots, \sigma_x^{(m)}\}$  попарно разделимые означивания префикса входной переменной программы  $\pi(x)$ . Тогда при всяком окончательном продолжении  $\sigma_x$  средняя длина вычислений  $\pi(\sigma_x^{(i)} \sigma_x)$  не меньше  $\log_2 m$ ,  $i = 1, 2, \dots, m$ .

**Доказательство.** Пусть формула  $F(x)$  представляет функцию, вычисляемую программой  $\pi(x)$ . Каждое означивание  $\sigma_x^{(i)}$  определяет совокупность конъюнктов в ОДНФ формулы  $F(\sigma_x^{(i)})$ , которые содержат в качестве переменных не означенный суффикс входной переменной программы. При окончательном продолжении  $\sigma_x$  в точности один из них становится тождественно истинным и определяет значение программы  $\pi(\sigma_x^{(i)} \sigma_x)$ . Из соответствия конъюнктов ОДНФ формул  $F(\sigma_x^{(i)})$  и размеченных путей следует, что при окончательном продолжении  $\sigma_x$  существуют не менее  $m$  размеченных путей соответственно  $t_1, t_2, \dots, t_m$ , начинающихся во входной вершине программы и оканчивающихся в заключительной. В совокупности они определяют однокорневое бинарное дерево  $T(\sigma_x)$  вычислений, в котором будет не менее  $m$  бинарных узлов, разделяющих эти размеченные пути  $t_1, t_2, \dots, t_m$ . Поэтому средняя длина всех путей в нем не менее  $\log_2 m$ .

**Следствие.** Пусть  $\{\sigma'_x, \sigma''_x, \dots, \sigma_x^{(m)}\}$  попарно разделимые означивания префикса входной переменной программы  $\pi(x)$ . Тогда

для всякого окончательного продолжения  $\sigma_x$  длина некоторого вычисления программы  $\pi(\sigma_x^{(i)} \sigma_x)$  не меньше  $t$ ,  $i = 1, 2, \dots, t$ .

**Доказательство.** Для получения 1 в  $l$ -м разряде, начиная с 0, оператору  $x + 1$  требуется не менее  $2^l$  применений. В каждой совокупности конъюнктов, выделяемых означиваниями  $\sigma_x', \sigma_x'', \dots, \sigma_x^{(m)}$ , имеется по меньшей мере одна подформула, которая имеет вид  $E^\sigma(\sigma_x^{(i)} x', \alpha)$ , где  $x'$  – не означенный суффикс входной переменной программы и  $\alpha$  – рабочая метапеременная, означенный префикс которой совпадает с  $\sigma_x^{(i)}$ .

### Заключение

Приведена нижняя оценка вычислений арифметических программ в зависимости от предметной области, над которой вычисление осуществляется. Свойства предметной области описываются как совокупность попарно разделимых означиваний префикса

входных переменных программ. При такой характеристике найдется вычисление, длина которого не меньше, чем мощность множества попарно разделимых означиваний префикса входа программы.

### Список литературы

1. Кормен Т.Х., Лейзерсон Ч.И., Ривест Р.Л., Штайн К. Алгоритмы: построение и анализ. М.: Вильямс, 2016.
2. Хопкрофт Д., Мотвани Р., Ульман Д. Введение в теорию автоматов, языков и вычислений. М.: Вильямс, 2012. 528 с.
3. Верещагин Н.К., Успенский В.А., Шень А. Колмогоровская сложность и алгоритмическая случайность. М.: МЦНМО, 2013. [Электронный ресурс]. URL: [http:// www.mcsme.ru/free-books/shen/kolmbook.pdf](http://www.mcsme.ru/free-books/shen/kolmbook.pdf) (дата обращения: 27.06.2019).
4. Goldreich O. P, NP, and NP-Completeness: The Basics of Complexity Theory. Cambridge University Press, 2011.
5. Попов С.В., Брошкова Н.Л. Длина вычисления арифметических программ. LAP Lambert Academic Publishing, Saarbrücken, Deutschland. 2014. 250 с.