

СИНТЕЗ РАСПОЗНАВАТЕЛЕЙ С МАГАЗИННОЙ ПАМЯТЬЮ ПО  
ДЕТЕРМИНИРОВАННЫМ СИНТАКСИЧЕСКИМ ДИАГРАММАМ

Ю. Д. Рязанов

Белгородский государственный технологический университет им. В.Г. Шухова

Поступила в редакцию 05.01.2014 г.

**Аннотация.** Рассматриваются вопросы использования синтаксических диаграмм при проектировании программ обработки формальных языков. Предложены алгоритмы синтеза МП-распознавателей с одним и с конечным множеством состояний по детерминированной синтаксической диаграмме.

**Ключевые слова:** МП-распознаватель, синтаксическая диаграмма, множество выбора, состояние.

**Annotation.** Problems of syntax diagram using in design of programs for processing formal languages considered in paper. Author suggest algorithms for the synthesis of pushdown recognizers with one and finite set of states from deterministic syntax diagram.

**Keywords:** Pushdown recognizers, syntax diagram, set of choice, state.

ВВЕДЕНИЕ

Классическая технология проектирования программ обработки контекстно-свободных языков предполагает задание языка в виде КС-грамматики, преобразование ее в МП-распознаватель и его программную реализацию [1–2]. Более наглядным способом описания языка являются синтаксические диаграммы (СД) [3]. В данной работе предлагаются алгоритмы преобразования детерминированных СД в МП-распознаватели с одним и с конечным множеством состояний.

ОСНОВНЫЕ ПОНЯТИЯ

Синтаксическую диаграмму будем задавать четверкой  $D = (T, N, G, S)$  [4], где

$T$  – конечное множество терминалов;

$N$  – конечное множество нетерминалов;

$S \in N$  – начальный нетерминал;

$G = (V, E)$  – ориентированный граф, где

$$V = V_T \cup V_N \cup V_u \cup V_{вход} \cup V_{выход},$$

где

$V_{вход}$  – множество точек входа,  $|V_{вход}| = |N|$ ;

$V_T$  – множество терминальных вершин;

$V_N$  – множество нетерминальных вершин;

$V_u$  – конечное множество узлов;

$V_{выход}$  – множество точек выхода,  $|V_{выход}| = |N|$ ;

$$E = E_1 \cup E_2 \cup E_3 \cup E_4 \cup E_5,$$

где

$E_1 \subseteq \{(a, b) | a \in V_{вход}, b \in V_u\}$  – множество входных дуг;

$E_2 \subseteq \{(a, b) | a \in V_u, b \in V_{выход}\}$  – множество выходных дуг;

$E_3 \subseteq \{(a, b) | a \in V_u, b \in V_T \cup V_N\}$  – множество дуг, выходящих из узлов;

$E_4 \subseteq \{(a, b) | a \in V_T \cup V_N, b \in V_u\}$  – множество дуг, входящих в узлы;

$E_5 \subseteq \{(a, b) | a \in V_u, b \in V_u\}$  – множество  $\varepsilon$ -дуг, соединяющих узлы.

Каждому нетерминалу соответствует связанная компонента графа. Компонента именуется соответствующим нетерминалом, имеет только одну точку входа и одну точку выхода и конечное множество вершин других типов. Точки входа и выхода на диаграмме компоненты не изображаются. Терминальная вершина изображается кружочком, в который вписан терминальный символ. Нетерминальная вершина изображается прямоугольником, в который вписан нетерминальный символ. Узел изображается на диаграмме жирной точкой. В точку входа не входит ни одна дуга и выходит конечное множество дуг (входные дуги компоненты). Узлы, в которые входят входные дуги, называются начальными. Из точки выхода не выходит ни одна дуга и входит конечное множество дуг (выходные дуги компоненты). Узлы, из которых выходят выходными дугами, называются заключительными. Каждая дуга, за исключением входных и выходных дуг, может выходить из узла и входить в терминальную или нетерминальную вершину или другой узел, либо выходить из терминальной или нетерминальной вершины и входить в узел. В каждую терминальную и нетерминальную вершину входит только одна дуга и выходит только одна дуга. На количество дуг, входящих в узлы и выходящих из них, ограничений нет.

Цепочку языка можно получить, «двигаясь» по дугам СД от точки входа начальной компоненты к ее точке выхода. При этом если дуга идет в терминальную вершину, то вписанный в нее символ добавляем в цепочку, если дуга идет в нетерминальную вершину, то переходим в соответствующую компоненту и движемся по ней аналогичным образом до точки выхода, после чего возвращаемся в предыдущую компоненту и продолжаем движение. После прохождения выходной дуги начальной компоненты в цепочку добавляем конечный маркер ( $\dagger$ ).

Символ  $x$ , который может быть добавлен в цепочку после прохождения выходящей из узла дуги  $e$ , принадлежит множеству выбора дуги  $e$ . Узел  $u$  называется детерминированным, если множества выбора любых двух дуг,

выходящих из узла  $u$ , не пересекаются. СД является детерминированной, если в ней все узлы детерминированные. В работе [4] описаны алгоритмы вычисления множеств выбора дуг, выходящих из узлов, и определения принадлежности СД классу детерминированных СД.

Определим интуитивно множества выбора дуг, выходящих из узлов, в СД на рис. 1 и припишем их к соответствующим дугам. Множества выбора дуг, входящих в терминальную вершину, очевидно, содержат только терминал, записанный в этой вершине. Множество выбора дуг, входящих в вершину с нетерминалом  $B$ , равно  $\{d, e\}$ , т. к. из начального узла 9 компоненты  $B$  дуги идут только в вершины с терминалами  $d$  и  $e$ . Множество выбора дуги  $(1, A)$  равно  $\{b, d, e, c\}$ . Принадлежность символа  $b$  этому множеству объясняется дугой  $(5, b)$ , символов  $d$  и  $e$  – дугой  $(5, B)$  и дугами, выходящими из узла 9, символа  $c$  – дугами  $(5, \text{выход})$  и  $(2, c)$ . Множество выбора дуги  $(4, \text{выход})$  содержит только конечный маркер  $\{\dagger\}$ , т. к. после прохождения этой дуги цепочка заканчивается. Множество выбора дуги  $(8, \text{выход})$  содержит только символ  $c$  потому, что после выхода из вершины с нетерминалом  $A$  (дуга  $(A, 2)$ ) попадаем в узел 2, из которого выходит единственная дуга  $(2, c)$ . Множество выбора дуги  $(11, \text{выход})$  равно  $\{a, c, d, e, \dagger\}$ . Это объясняется последовательностями дуг  $((B, 8), (8, a))$ ,  $((B, 2), (2, c))$ ,  $((B, 4), (4, B), (9, d))$  и  $((B, 4), (4, \text{выход}))$ . Выполнив анализ множеств выбора дуг, выходящих из узлов, можно сделать вывод, что СД детерминированная.

МП-распознаватель определяется пятью объектами [2]:

- 1) конечным множеством входных символов, в которое входит и конечный маркер;
- 2) конечным множеством магазинных символов, включающим маркер дна ( $\nabla$ );
- 3) конечным множеством состояний, включающим начальное состояние;
- 4) управляющим устройством, которое каждой комбинации входного символа, магазинного символа и состояния ставит в соответствие переход или выход. Переход заключа-

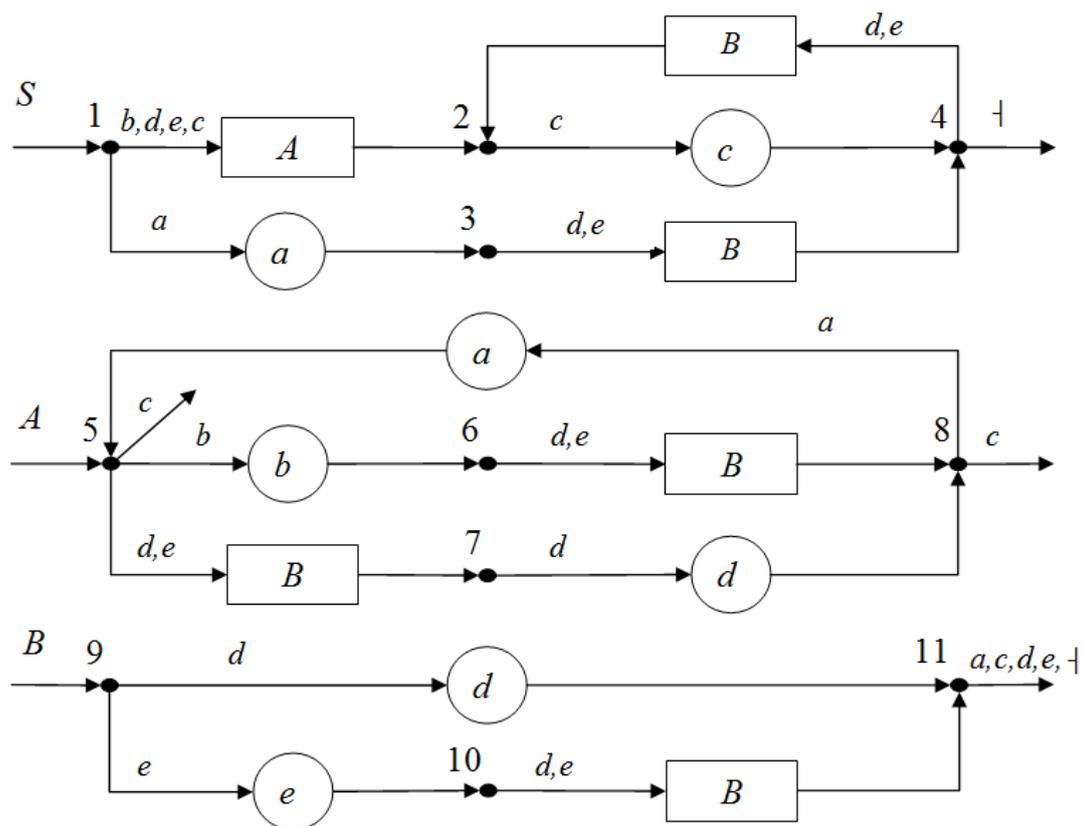


Рис. 1. Синтаксическая диаграмма

### СИНТЕЗ МП-РАСПОЗНАВАТЕЛЯ С КОНЕЧНЫМ МНОЖЕСТВОМ СОСТОЯНИЙ

ется в выполнении операций над магазином, состоянием и входом. МП-распознаватель имеет два выхода – ДОПУСТИТЬ и ОТВЕРГНУТЬ;

5) начальным содержимым магазина, которое представляет собой маркер дна, за которым следует (возможно, пустая) цепочка других магазинных символов. Мы будем строить МП-распознаватели, в которых над магазином выполняются операции ВТОЛКНУТЬ( $m$ ), где  $m$  –магазинный символ, ВЫТОЛКНУТЬ и расширенная операция ЗАМЕНИТЬ, которая заключается в выталкивании верхнего символа магазина и последующем выполнении нескольких вталкиваний. Над состоянием выполняется единственная операция – переход в состояние. В МП-распознавателе с одним состоянием операции над состоянием не выполняются. Над входом возможны операция СДВИГ, которая заключается в переходе к следующему символу в обрабатываемой цепочке, и ДЕРЖАТЬ, которая указывает, что на следующем шаге работы МП-распознавателя будет обрабатываться тот же символ входной цепочки.

Определим МП-распознаватель с конечным числом состояний, который распознает цепочки языка, заданного детерминированной СД:

1) конечное множество входных символов представляет собой множество терминалов СД, дополненное конечным маркером;

2) конечное множество магазинных символов представляет собой множество узлов СД, в которые входят дуги, выходящие из нетерминальных вершин, дополненное маркером дна;

3) конечное множество состояний представляет собой множество узлов СД, дополненное допускающим состоянием «доп». Начальным состоянием является состояние, соответствующее начальному узлу начальной компоненты;

4) управляющее устройство будем представлять таблицей из пяти столбцов, в которой каждая строка соответствует допустимо-

му переходу. Строки в таблице попарно различны. В первом столбце строки указывается состояние  $a_m$ , из которого выполняется переход, во втором – состояние перехода  $a_s$ , в третьем – множество входных символов, при которых возможен переход из  $a_m$  в  $a_s$ , в четвертом – верхний магазинный символ, при котором возможен переход из  $a_m$  в  $a_s$ , в пятом – действия, выполняемые при переходе из  $a_m$  в  $a_s$  над магазином и входом или выход ДОПУСТИТЬ. Если содержимое магазина на переходе не изменяется, то операции над магазином не указываются. Не указывается и операция ДЕРЖАТЬ над входом. Если возможность перехода из  $a_m$  в  $a_s$  не зависит от верхнего символа в магазине, то четвертый столбец не заполняется.

Таблица МП-распознавателя формируется по следующим правилам.

Если из узла  $a_m$  СД существует путь в узел  $a_s$  через терминальную вершину с терминалом  $t$ , то в таблицу добавляется строка, содержащая в первом столбце состояние  $a_m$ , во втором – состояние  $a_s$ , в третьем – терминал  $t$ , в четвертом – пусто, в пятом – СДВИГ.

Если из узла  $a_m$  СД существует путь в узел  $a_s$  через нетерминальную вершину с нетерминалом  $X$  и узел  $a_n$  является начальным в компоненте  $X$ , то в таблицу добавляется строка, содержащая в первом столбце состояние  $a_m$ , во втором – состояние  $a_s$ , в третьем – множество выбора дуги  $(a_m, X)$ , в четвертом – пусто, в пятом – ВТОЛКНУТЬ( $a_n$ ).

Если узел  $a_m$  заключительный в компоненте  $X$ , и в СД существует дуга  $(X, a_s)$ , то в таблицу добавляется строка, содержащая в первом столбце состояние  $a_m$ , во втором – состояние  $a_s$ , в третьем – множество выбора дуги  $(a_m, \text{выход})$ , в четвертом –  $a_s$ , в пятом – ВЫТОЛКНУТЬ.

Если узел  $a_m$  заключительный в начальной компоненте, то в таблицу добавляется строка, содержащая в первом столбце состояние  $a_m$ , во втором – состояние «доп», в третьем – концевой маркер, в четвертом – маркер дна магазина, в пятом – ДОПУСТИТЬ.

Формирование таблицы продолжается, пока по указанным выше правилам можно получить строку, которой еще нет в таблице;

5) начальное содержимое магазина представляет собой только маркер дна (магазин пуст).

В табл. 1 представлена таблица МП-распознавателя, построенная по СД (рис. 1).

Алгоритм работы МП-распознавателя следующий. Если для текущего состояния истинно условие некоторого перехода (определяется третьим и четвертым столбцом), то МП-распознаватель переходит в новое состояние и выполняются соответствующие переходу действия, иначе цепочка отвергается. МП-распознаватель моделирует получение обрабатываемой цепочки в СД. Текущее состояние представляет собой узел СД, из которого нужно продолжить «движение» при обработке символа цепочки. Если «движение» проходит через терминальную вершину, то распознаватель переходит в состояние, в которое идет дуга из этой вершины, после чего обрабатывается следующий символ входной цепочки. Если «движение» проходит через нетерминальную вершину  $X$ , то распознаватель переходит в состояние, соответствующее начальному узлу компоненты  $X$ , а узел, в который идет дуга из вершины с нетерминалом  $X$  запоминается в магазине. Если достигнут заключительный узел компоненты, то выполняется переход на узел, который был запомнен в магазине. Если дошли до заключительного узла начальной компоненты, цепочка закончилась и магазин пуст, то цепочка допускается.

В табл. 2 представлен протокол работы МП-распознавателя (табл. 1) при обработке цепочки  $adedc\downarrow$ . Просматривая второй столбец в табл. 2 сверху вниз, увидим последовательность узлов СД (рис. 1), которую нужно пройти, чтобы получить цепочку  $adedc\downarrow$ .

## СИНТЕЗ МП-РАСПОЗНАВАТЕЛЯ С ОДНИМ СОСТОЯНИЕМ

Теперь определим МП-распознаватель с одним состоянием, который распознает цепочки языка, заданного детерминированной СД:

1) конечное множество входных символов представляет собой множество терминалов СД, дополненное концевым маркером;

МП-распознаватель с конечным множеством состояний

Текущее состояние	Состояние перехода	Входные символы	Верх магазина	Действия
1	3	<i>a</i>		СДВИГ
1	5	<i>b, c, d, e</i>		ВТОЛКНУТЬ(2)
2	4	<i>c</i>		СДВИГ
3	9	<i>d, e</i>		ВТОЛКНУТЬ(4)
4	9	<i>d, e</i>		ВТОЛКНУТЬ(2)
4	«доп»		∇	ДОПУСТИТЬ
5	6	<i>b</i>		СДВИГ
5	2	<i>c</i>	2	ВЫТОЛКНУТЬ
5	9	<i>d, e</i>		ВТОЛКНУТЬ(7)
6	9	<i>d, e</i>		ВТОЛКНУТЬ(8)
7	8	<i>d</i>		СДВИГ
8	2	<i>c</i>	2	ВЫТОЛКНУТЬ
8	5	<i>a</i>		СДВИГ
9	10	<i>e</i>		СДВИГ
9	11	<i>d</i>		СДВИГ
10	9	<i>d, e</i>		ВТОЛКНУТЬ(11)
11	2	<i>a, c, d, e,  </i>	2	ВЫТОЛКНУТЬ
11	4	<i>a, c, d, e,  </i>	4	ВЫТОЛКНУТЬ
11	7	<i>a, c, d, e,  </i>	7	ВЫТОЛКНУТЬ
11	8	<i>a, c, d, e,  </i>	8	ВЫТОЛКНУТЬ
11	11	<i>a, c, d, e,  </i>	11	ВЫТОЛКНУТЬ

Таблица 2

Протокол работы МП-распознавателя

Шаг	Состояние	Символ	Магазин	Действие
1	1	<i>a</i>	∇	СДВИГ
2	3	<i>d</i>	∇	ВТОЛКНУТЬ(4)
3	9	<i>d</i>	∇ 4	СДВИГ
4	11	<i>e</i>	∇ 4	ВЫТОЛКНУТЬ
5	4	<i>e</i>	∇	ВТОЛКНУТЬ(2)
6	9	<i>e</i>	∇ 2	СДВИГ
7	10	<i>d</i>	∇ 2	ВТОЛКНУТЬ(11)
8	9	<i>d</i>	∇ 2 11	СДВИГ
9	11	<i>c</i>	∇ 2 11	ВЫТОЛКНУТЬ
10	11	<i>c</i>	∇ 2	ВЫТОЛКНУТЬ
11	2	<i>c</i>	∇	СДВИГ
12	4		∇	ДОПУСТИТЬ

2) конечное множество магазинных символов представляет собой множество узлов СД, дополненное маркером дна;

3) состояние одно;

4) управляющее устройство будем представлять таблицей  $T$ , в которой строки соответствуют магазинным символам, а столбцы – входным символам. Клетки таблицы  $T$  заполняются по следующим правилам.

Если из узла  $u$  СД дуга ведет в терминальную вершину с терминалом  $t$ , а из нее выходит дуга в узел  $u'$ , то в клетку  $T_{u,t}$  записать операции ЗАМЕНИТЬ( $u'$ ) и СДВИГ.

Если из узла  $u$  СД дуга ведет в нетерминальную вершину с нетерминалом  $X$ , а из нее выходит дуга в узел  $u'$  и терминал  $t$  принадлежит множеству выбора дуги  $(u, X)$ , то в клетку  $T_{u,t}$  записать операции ЗАМЕНИТЬ( $u'v$ ) и ДЕРЖАТЬ, где  $v$  – начальный узел компоненты  $X$ .

Если узел  $u$  является заключительным узлом компоненты  $X$  и терминал  $t$  принадлежит множеству выбора дуги  $(u, \text{выход})$ , то в

клетку  $T_{u,t}$  записать операции ВЫТОЛКНУТЬ и ДЕРЖАТЬ.

В клетку  $T_{\nabla+}$  записать выход ДОПУСТИТЬ.

Формирование таблицы продолжается, пока по указанным выше правилам можно заполнить пустую клетку.

Пустым клеткам соответствует выход ОТВЕРГНУТЬ;

5) начальное содержимое магазина представляет собой маркер дна и начальный узел начальной компоненты.

В табл. 3 представлена таблица МП-распознавателя, построенная по СД (рис. 1). В ней названия операций над магазином сокращены, а операция ДЕРЖАТЬ – опущена.

Алгоритм работы МП-распознавателя следующий. По верхнему символу магазина и входному символу определяется клетка таблицы и выполняются записанные в ней действия.

МП-распознаватель моделирует получение обрабатываемой цепочки в СД. Верхний символ магазина представляет собой узел СД,

Таблица 3

МП-распознаватель с одним состоянием

	$a$	$b$	$c$	$d$	$e$	$\dagger$
1	ЗАМ(3) СДВИГ	ЗАМ(2 5)	ЗАМ(2 5)	ЗАМ(2 5)	ЗАМ(2 5)	
2			ЗАМ(4) СДВИГ			
3				ЗАМ(4 9)	ЗАМ(4 9)	
4				ЗАМ(2 9)	ЗАМ(2 9)	ВЫТОЛК
5		ЗАМ(6) СДВИГ	ВЫТОЛК	ЗАМ(7 9)	ЗАМ(7 9)	
6				ЗАМ(8 9)	ЗАМ(8 9)	
7				ЗАМ(8) СДВИГ		
8	ЗАМ(5) СДВИГ		ВЫТОЛК			
9				ЗАМ(11) СДВИГ	ЗАМ(10) СДВИГ	
10				ЗАМ(11 9)	ЗАМ(11 9)	
11	ВЫТОЛК		ВЫТОЛК	ВЫТОЛК	ВЫТОЛК	ВЫТОЛК
$\nabla$						ДОП

из которого нужно продолжить «движение» при обработке символа цепочки. Если «движение» проходит через терминальную вершину, то верхний символ магазина заменяется на узел, в который идет дуга из этой вершины, после чего обрабатывается следующий символ входной цепочки. Если «движение» проходит через нетерминальную вершину  $X$ , то верхний символ магазина заменяется на узел  $u'$ , в который идет дуга из этой вершины и выталкивается начальный узел компоненты  $X$  (ЗАМЕНИТЬ( $u'v$ )), т. к. из него должно быть продолжено «движение», направление которого определяется тем же символом цепочки (ДЕРЖАТЬ). Возвращение в узел  $u'$  происходит при достижении заключительного узла в компоненте  $X$ . В этот момент выталкивается верхний символ магазина, под которым находится узел  $u'$ . Направление «движения» из узла  $u'$  определяется символом цепочки, который анализировался в заключительном узле компоненты  $X$  (ДЕРЖАТЬ). Если в магазине окажется только заключительный узел начальной компоненты и цепочка закончится ( $\downarrow$ ), то она допускается. Это обнаруживается выталкиванием верхнего символа магазина, после чего магазин должен стать пустым (клетка  $T_{v+}$  содержит выход ДОПУСТИТЬ).

В табл. 4 представлен протокол работы МП-распознавателя (табл. 3) при обработке цепочки  $adedc\downarrow$ . Просматривая на каждом шаге работы МП-распознавателя верхний символ магазина, увидим последовательность узлов СД (рис. 1), которую нужно пройти, чтобы получить цепочку  $adedc\downarrow$ .

## ЗАКЛЮЧЕНИЕ

В работе представлены алгоритмы синтеза детерминированных МП-распознавателей с одним и с конечным множеством состояний по детерминированным синтаксическим диаграммам, неформально обоснована корректность алгоритмов, приведены примеры синтеза МП-распознавателей и протоколы обработки ими входной цепочки. Эти алгоритмы могут быть использованы в системах автоматизированного построения трансляторов на основе синтаксических диаграмм.

Таблица 4

Протокол работы МП-распознавателя с одним состоянием

Шаг	Символ	Магазин	Действие
1	$a$	$\nabla 1$	ЗАМ(3) СДВИГ
2	$d$	$\nabla 3$	ЗАМ(4 9) ДЕРЖАТЬ
3	$d$	$\nabla 4 9$	ЗАМ(11) СДВИГ
4	$e$	$\nabla 4 11$	ВЫТОЛК ДЕРЖАТЬ
5	$e$	$\nabla 4$	ЗАМ(2 9) ДЕРЖАТЬ
6	$e$	$\nabla 2 9$	ЗАМ(10) СДВИГ
7	$d$	$\nabla 2 10$	ЗАМ(11 9) ДЕРЖАТЬ
8	$d$	$\nabla 2 11 9$	ЗАМ(11) СДВИГ
9	$c$	$\nabla 2 11 11$	ВЫТОЛК ДЕРЖАТЬ
10	$c$	$\nabla 2 11$	ВЫТОЛК ДЕРЖАТЬ
11	$c$	$\nabla 2$	ЗАМ(4) СДВИГ
12	$\downarrow$	$\nabla 4$	ВЫТОЛК ДЕРЖАТЬ
13	$\downarrow$	$\nabla$	ДОПУСТИТЬ

## СПИСОК ЛИТЕРАТУРЫ

1. Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции. – М. : Мир, 1978.

2. Льюис Ф., Розенкранц Д., Стирнз Р. Теоретические основы проектирования компиляторов. – М. : Мир, 1979.

3. Йенсен К., Вирт Н. Паскаль. Руководство для пользователя и описание языка. – М. : Финансы и статистика, 1982.

**Рязанов Юрий Дмитриевич** – Белгородский государственный технологический университет им. В. Г. Шухова, доцент кафедры программного обеспечения вычислительной техники и автоматизированных систем, тел.: +7 (910) 325 – 73 – 75.  
E-mail: Ryazanov.iurij@yandex.ru.

4. Рязанов Ю. Д., Севальнева М. Н. Анализ синтаксических диаграмм и синтез программ-распознавателей линейной сложности // Научные ведомости БелГУ. Сер. История. Политология. Экономика. Информатика. 2013. № 8 (151). Вып. 26/1. С. 128–136.

**Ryazanov Y. D.** – BSTU after V.G. Shukhov, PhD in engineering, assistant professor of the department of software computer and automated systems, phone: +7 (910) 325 – 73 – 75.  
E-mail: Ryazanov.iurij@yandex.ru.