

1	2	3	4	5	6	7	8	9	10

1. Дана грамматика  $G$ :

$S \rightarrow Sb|aSb|ab|aAb$   
 $aAb \rightarrow aDAb|aab$   
 $aD \rightarrow Da$   
 $D \rightarrow a$

(а) Описать язык  $L(G)$  в виде теоретико-множественной формулы или исчерпывающего словесного описания (не более 300 печатных знаков включая пробелы).

(б) Каким из перечисленных классов грамматик принадлежит  $G$ ? **Ответ:**

Класс $\Pi$	$G \in \Pi?$ (да/нет)
регулярные	
контекстно-зависимые	
контекстно-свободные	
грамматики типа 0	
неукорачивающие	

(в) Тип языка: найти такое целое  $k$ , что язык  $L(G)$  является языком типа  $k$  и не является языком  $k+1$ .

2. Среди правил грамматик  $G_1$  и  $G_2$  найти и вычеркнуть одно правило (альтернативу) так, чтобы  $G_1$  и  $G_2$  стали эквивалентными. Ответ обосновать.

$G_1:$   $S \rightarrow ITI | ISIU$   
 $B \rightarrow 0C | 0$   
 $C \rightarrow \varepsilon$   
 $T \rightarrow TI | UT | US$

$G_2:$   $S \rightarrow ITI | 10T | 10UC | U | IU | IS$   
 $C \rightarrow 0TZ$   
 $U \rightarrow 0T | 0UC | \varepsilon$   
 $Z \rightarrow ZI | 0 | ZS$

3. а) Построить по заданной грамматике  $G$  конечный автомат (в виде ДС).

б) Если автомат оказался недетерминированным, преобразовать его в соответствии с алгоритмом преобразования НКА в эквивалентный ДКА.

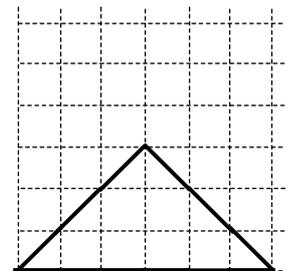
в) По ДКА построить праволинейную грамматику.

$G:$   $S \rightarrow A\perp | B\perp | \perp$   
 $A \rightarrow Aa | Ba | Bb | a$   
 $B \rightarrow Bb | Aa | Ab | b$

4. Имеется клетчатый лист бумаги, бесконечный вправо и вверх. В левом нижнем углу расположено перо, оставляющее на бумаге след при перемещении. Перо управляется интерпретатором-графопостроителем. На вход интерпретатору подается последовательность символов-команд из алфавита  $\{a, b, d\}$ :

- $a$  — переместить перо на одну клетку вправо;
- $d$  — переместить перо по диагонали на одну клетку влево-вверх;
- $b$  — переместить перо по диагонали на одну клетку влево-вниз.

Например, по последовательности  $aaaaadddbbb$  интерпретатор построит прямоугольный равнобедренный треугольник с основанием длины 6 (см. рисунок).



Построить грамматику с терминальным алфавитом  $\{a, b, d\}$ , описывающую подобные изображенному на рисунке треугольники с основанием длины  $2k$ ,  $k \geq 1$ . В грамматике должно быть **не более 4** правил вывода (считая альтернативы).

5. Дана однозначная КС-грамматика  $G_{parenthesis}$ , порождающая язык  $L$  сбалансированных скобочных систем. *Неделимой* называется скобочная система, которую нельзя представить в виде конкатенации двух других скобочных систем. Например, цепочка  $(( ))( )$  – неделимая скобочная система. Вставить в грамматику действия вида  $\langle cout \ll 'символ'; \rangle$  так, чтобы в процессе рекурсивного спуска был реализован перевод  $\tau = \{(x, 1^{f(x)}) \mid x \in L, f(x) - \text{количество неделимых скобочных подсистем в } x\}$ . Например, для  $x = (( ))( )(( ))( )$   $f(x) = 5$ .

$$G_{parenthesis}: \\ S \rightarrow TS \mid \epsilon \\ T \rightarrow (S)$$

6. От каких компонентов интегрированных систем программирования получают и каким компонентам передают данные для обработки входящие в них редакторы связей? Что подвергается обработке в редакторах связей и с чем связана необходимость этой обработки?

7. Какие области памяти выделяются компиляторами для хранения локальных данных процедур? Какая дисциплина распределения памяти при этом реализуется? Какой способ использования выбирается для этих областей? Какая ещё информация должна размещаться в тех же областях памяти, с той же дисциплиной распределения и с тем же способом использования?

8. По заданной грамматике  $G = \{\{a, b, c\}, \{S, T, U\}, P, S\}$  получить эквивалентную неукорачивающую контекстно-свободную грамматику (использовать алгоритм устранения правил с пустой правой частью).

$$P: \\ S \rightarrow aTb \mid UT \mid cS \\ T \rightarrow aT \mid bU \mid \epsilon \\ U \rightarrow bT \mid aU \mid \epsilon$$

9. Дана грамматика  $G$ . Применим ли к ней метод рекурсивного спуска? Ответ обосновать.

$$G: \\ S \rightarrow aSb \mid Yb \mid bb \\ Y \rightarrow cSY \mid B \mid \epsilon \\ B \rightarrow d \mid \epsilon$$

10. По приведенной обратной польской записи фрагмента программы, написанной на языке Си++, восстановите этот фрагмент двумя способами: сначала, пользуясь операторами условного и безусловного перехода на метку, а затем (если это возможно), пользуясь только операторами структурного программирования без переходов. Операция ПОЛИЗ '@' соответствует унарной операции изменения знака.

ПОЛИЗ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
	a	23	!F	&a	&a	b	8	—	7	*	b	a	@	23	/	/	—

18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
=	+=	;	28	!	&a	&b	+#	=	;	&b	#+	a	b	—	<	38	!F	1	!	}