

**ОТВЕТЫ :: Варианты 6 и 8 :: грамматики**

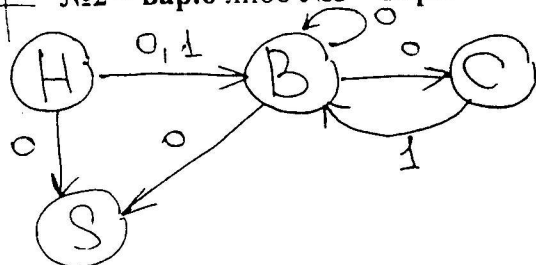
**№1 – вар.6 либо №2 – вар.8**

неукорачивающая грамматика

Опр. Грамматика  $G = (VT, VN, P, S)$  называется *неукорачивающей грамматикой*, если каждое правило из  $P$  имеет вид  $\alpha \rightarrow \beta$ , где  $\alpha, \beta \in (VT \cup VN)^+$  и  $|\alpha| \leq |\beta|$ .

$L(G) = \{ a^n b^n c^{3n}, n \geq 1 \}$  – язык типа 1

**№2 – вар.6 либо №5 – вар.8**



$S \rightarrow 0 | Bc$   
 $B \rightarrow 0 | 1 | Bc | cA$   
 $c \rightarrow Bc$

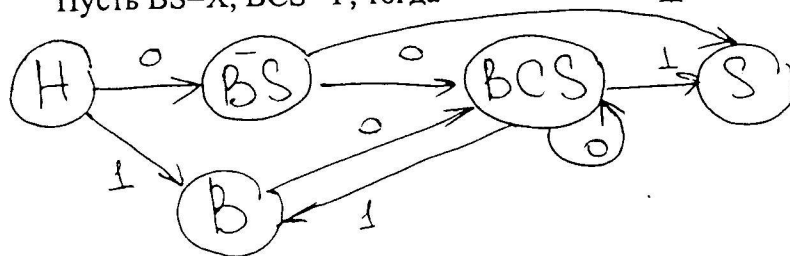
$F(H,0)=BS$   
 $F(H,1)=B$

$F(BS,0)=BCS$   
 $F(BS,1)=$

$F(B,0)=BCS$   
 $F(B,1)=$

$F(BCS,0)=BCS$   
 $F(BCS,1)=B$

Пусть  $BS=X$ ,  $BCS=Y$ ; тогда



$G: S \rightarrow X1 | Y1 \quad X \rightarrow 0 \quad Y \rightarrow Y0 | X0 | B0 \quad B \rightarrow Y1 | 1$

**№3 – вар.6 либо №1 – вар.8**

$FIRST(A) = \{a,c\}$ ;  $FOLLOW(A) = \{b\}$  → пока хорошо

$FIRST(D) = \{b\}$ ;  $FOLLOW(D) = \{a,b,c\}$  → PC-метод неприменим

**№4 – вар.6 либо №4 – вар.8**

$G: S \rightarrow 1S0 | A$   
 $A \rightarrow 0A | 0$

$G: S \rightarrow 1S0 \langle \text{putchar}('0'); \rangle | A$   
 $A \rightarrow 0 \langle \text{putchar}('1'); \text{putchar}('1'); \rangle A1 | 0 \langle \text{putchar}('1'); \text{putchar}('1'); \rangle$

**№5 – вар.6 либо №3 – вар.8**

$i, 0, =, S, 0, =, i, 5, <, 34, !F, 18, !, i, @, 7, !, S, S, a, i, [], x, \sin, S, -, 2, +, *, +, =, 14, !$

```

for (i=0, S=0; i<5; i++)
{
    S = S + a[i] * (sin x - S + 2);
}
  
```

20012

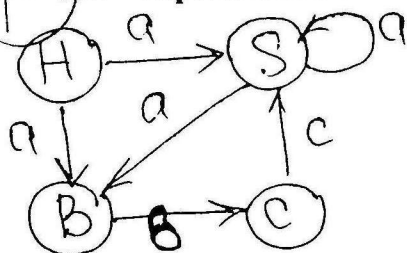
# ОТВЕТЫ :: Варианты 5 и 7 :: грамматики

- №1 – вар.5 либо №2 – вар.7  
грамматика типа 0

Опр. Грамматика  $G = (VT, VN, P, S)$  называется грамматикой типа 0, если на правила из  $P$  не накладывается никаких ограничений, кроме тех, что заданы в определении грамматики.

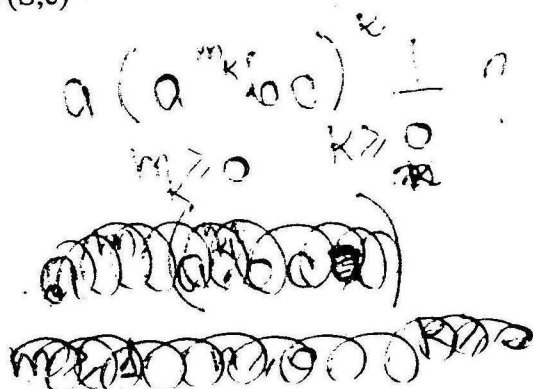
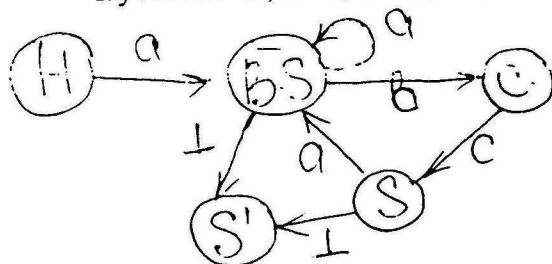
$$L(G) = \{ (ab)^n c^n, n \geq 0 \} \rightarrow L(G) - \{ \epsilon \} - \text{КС - язык}$$

- №2 – вар.5 либо №5 – вар.7



|                |                 |               |                |
|----------------|-----------------|---------------|----------------|
| $F(H, a) = BS$ | $F(BS, a) = BS$ | $F(C, a) =$   | $F(S, a) = BS$ |
| $F(H, b) =$    | $F(BS, b) = C$  | $F(C, b) =$   | $F(S, b) =$    |
| $F(H, c) =$    | $F(BS, c) =$    | $F(C, c) = S$ | $F(S, c) =$    |

Пусть  $BS = X$ ,  $S = Y$ ,  $S' = S$ ; тогда



$$G: \quad S \rightarrow X1 \mid Y1 \quad X \rightarrow Xa \mid Ya \mid a \quad Y \rightarrow Cc \quad C \rightarrow Xb$$

- №3 – вар.5 либо №1 – вар.7

$$\text{FIRST}(A) = \{a, c\}; \text{FOLLOW}(A) = \{b\}$$

$$\text{FIRST}(D) = \{d\}; \text{FOLLOW}(D) = \{a, d\} \rightarrow \text{PC-метод неприменим}$$

- №4 – вар.5 либо №4 – вар.7

$$G: \quad S \rightarrow 1S0 \mid A$$

$$A \rightarrow 0A1 \mid \epsilon$$

$$G: \quad S \rightarrow 1 \langle \text{putchar}('1'); \rangle S0 \langle \text{putchar}('0'); \text{putchar}('0'); \rangle \mid A$$

$$A \rightarrow 0 \langle \text{putchar}('1'); \rangle A1 \mid \epsilon$$

- №5 – вар.1 либо №3 – вар.7

$i, 0, =, \underline{a}, a, b, 0, 10, -, b, *, \text{exp}, *, -, =, i, @, a, b, >, 4, !T$

## ОТВЕТЫ : : : Варианты 2 и 4 : : : грамматики

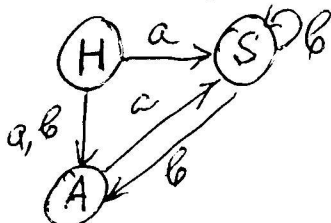
**№1 – вар.2 либо №5 – вар.4**

тип 1;-неукорачивающая, но не КЗ (из-за альтернативы 01)

$L(G) = \{ 1^{n-1}010^n, n \geq 1 \}$  – КС – язык

$S \rightarrow 1S0 \mid 010$

**№2 – вар.2 либо №6 – вар.4**



$F(H, a) = AS$

$F(H, b) = A$

$F(AS, a) = S$

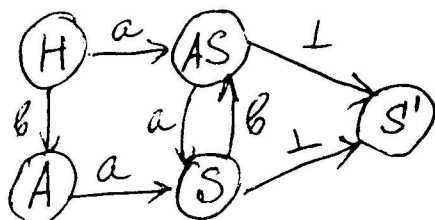
$F(AS, b) = AS$

$F(A, a) = S$

$F(A, b) = \emptyset$

$F(S, a) = \emptyset$

$F(S, b) = AS$



Пусть  $AS=X$ ,  $S=Y$ ,  $S'=S$ ; тогда

G:  $S \rightarrow X1 \mid Y1$   
 $Y \rightarrow Aa \mid Xa$   
 $X \rightarrow Xb \mid Yb \mid a$   
 $A \rightarrow b$

**№3 – вар.2 либо №7 – вар.4**

$FIRST(B) = \{a, b\}$ ;  $FOLLOW(B) = \{b, c\}$  → PC-метод неприменим

(нетерминализованная альтернатива A в нетерминале S допустима, т.к. она не является конкурентом другой терминализованной альтернативе)

**№4 – вар.2 либо №8 – вар.4**

G:  $S \rightarrow aS \mid bS \mid cA$   
 $A \rightarrow cA \mid \epsilon$

G':  $S \rightarrow aS \text{ <putchar('c')>} \mid bS \mid c \text{ <putchar('a')>} A$   
 $A \rightarrow c \text{ <putchar('a')>} A \mid \epsilon$

**№5 – вар.2 либо №9 – вар.4**

Не является, т.к. изменен порядок операндов в исходном тексте программы → такая запись не может быть сгенерирована автоматически в процессе синтаксического анализа. Однако указанная запись семантически эквивалентна исходной.

ПОЛИЗом исходной конструкции является запись

$\underline{i}, 1, =, i, n, <, 37, !F, 18, !, i, i, 2, +, =, 4, !, a, a, b, +, 1, -, x, y, y, 2, +, /, *, 5, +, *, =, 11, !$

## **ОТВЕТЫ : : : Варианты 10 и 12 : : : СП и STL**

### **№6– вар.10 либо №1 – вар.12**

- пошаговое выполнение программы (шаг=строка; с трассировкой внутри вызываемой функции и без нее)
- выполнение программы до строки, в которой в редакторе стоит курсор
- выделение выполняемой в данный момент строки
- приостановка выполнения программы:
  - можно запросить значение переменной
  - заказать вычисление некоторого выражения
  - можно изменить значения переменной и продолжить выполнение программы (! но редко какой отладчик позволяет изменять программный код, т.е. поддерживает частичную перекомпиляцию)
- расставить/снять точки останова, которые визуализируются в текстовом редакторе
- вся информация должна выдаваться в терминах исходной программы

### **№7 – вар.10 либо №2 – вар.12**

исх.пр-ма -> ЛА->посл.лексем+таблицы->СА->внутр.предст.->КУ->ГЕН->объект.пр-ма  
Функции ЛА:

- объединить литералы в лексемы – минимальные элементы программы, обладающие определенным смыслом в анализируемом языке
- сформировать таблицы, которые будут использоваться и пополняться на каждом из последующих этапов
- удалить комментарии

### **№8 – вар.10 либо №8 – вар.12**

```
#include <iostream>
#include <vector>
using namespace std;
```

```
void g(vector<double> &v)
{vector<double>::const_iterator p=v.begin();
double s=1;
while (p!=v.end())
    { s=s**p; p++;}
v.push_back(s);
};
```

либо

```
void f(vector<double> &v)
{double s=1; int i;
for (i=0; i<v.size(); i++) s=s*v[i];
v.push_back(s);
};
```

## ОТВЕТЫ : : : Варианты 9 и 11 : : : СП и STL

### №6 – вар.9 либо №1 – вар.11

- подготовка текста программы (обычные действия по созданию, редактированию, сохранению файла с текстом программы)
- визуализация текста с выделением лексем
- помощь в написании кода («интуитивный помощник»- Code Insight):
  - дополнение кода (а. => получаем список всех членов класса)
  - шаблоны кода (на «горячих» клавишах – часто используемые программные конструкции)
  - контекстный список параметров (f( => список параметров f )
  - всплывающие подсказки об атрибутах идентификаторов, если на него установить курсор
- перенос и копирование текста не только через буфер, но и с помощью drag&drop
- отображение ошибок, обнаруженных на этапе компиляции, в тексте программы
- визуализация контрольных точек останова при отладке

### №7 – вар.9 либо №2 – вар.11

исх.пр-ма -> ЛА->посл.лексем+таблицы->СА->внутр.предст.->КУ->ГЕН->объект.пр-ма  
Основная функция семантического анализатора:

- контроль контекстных условий, выполнение которых не может быть проверено на этапе синтаксического анализа, т.к. они не могут быть заданы с помощью КС/УКС - грамматик.

Примеры:

- в программе используется неописанная / описанная дважды переменная
- список фактических параметров функции не соответствует по количеству и/или типам списку формальных параметров
- тип выражения в условии операторов if, while в Паскале не является булевым

### №8 – вар.9 либо №8 – вар.11

```
#include <iostream>
```

```
#include <list>
```

```
using namespace std;
```

```
void g(list<double> &lst)
```

```
{list<double>::const_iterator p=lst.begin();
```

```
double s=0; int n=0;
```

```
while (p!=lst.end())
```

```
{ s=s+*p; n++; p++;}
```

```
if (n!=0) lst.push_back(s/n);
```

```
};
```

*also ... lst.push\_back(s/lst.size());*

## ОТВЕТЫ : : : Варианты 2 и 4 : : : СП и STL

### №6 – вар.2 либо №1 – вар.4

- пошаговое выполнение программы (шаг=строка; с трассировкой внутри вызываемой функции и без нее)
- выполнение программы до строки, в которой в редакторе стоит курсор
- выделение выполняемой в данный момент строки
- приостановка выполнения программы:
  - можно запросить значение переменной
  - заказать вычисление некоторого выражения
  - можно изменить значения переменной и продолжить выполнение программы (! но редко какой отладчик позволяет изменять программный код, т.е. поддерживает частичную перекомпиляцию)
- расставить/снять точки останова, которые визуализируются в текстовом редакторе
- вся информация должна выдаваться в терминах исходной программы

### №7 – вар.2 либо №2 – вар.4

исх.пр-ма -> ЛА->посл.лексем+таблицы->СА->внутр.предст.->КУ->ГЕН->объект.пр-ма  
 Функции СА:

- определить, имеет ли цепочка лексем структуру, заданную синтаксисом языка программирования
- зафиксировать эту структуру во внутреннем представлении программы

### №8 – вар.2 либо №3 – вар.4

Контейнер – это класс, объекты которого предназначены для хранения других объектов. В библиотеку STL входят следующие контейнеры:

|                                        |                                                                                 |
|----------------------------------------|---------------------------------------------------------------------------------|
| vector – одномерный вектор             | list – двусвязный список                                                        |
| queue - очередь                        | deque – очередь с двумя концами                                                 |
| priority_queue - очередь с приоритетом | stack - стек                                                                    |
| map – ассоциативный массив             | multimap – ассоциативный массив, элементы которого могут иметь одинаковые ключи |
| set - множество                        | multiset – множество, которое может содержать одинаковые элементы               |

### №9 – вар.2 либо №4 – вар.4

```
#include <iostream>
#include <vector>
#include <list>
using namespace std;
void g(vector<int> &v, list<int> &lst)
{ list<int>::const_iterator p=lst.begin();
  while (p!=lst.end())
    {if (!(*p%2)) v.push_back(*p); p++;}
  for (int i=0; i<v.size(); i++)
    cout<<v[i]<<" "<<endl;
  cout<<endl;
};
int main()
{vector<int> v; list<int> lst;
  for (int i=0; i<20; i++) lst.push_back(i);
  cout<<"list is created"<<endl;
  g(v,lst);
  return 0;
}
```