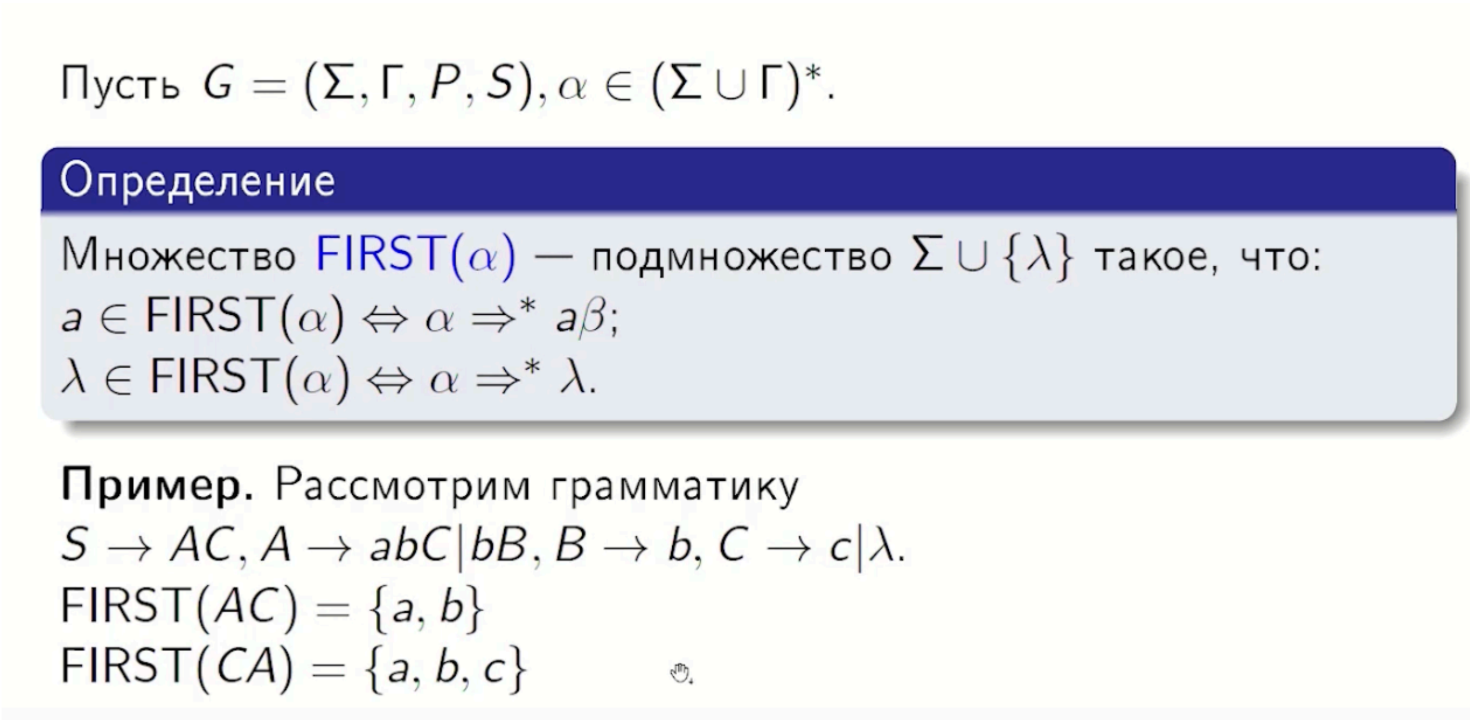
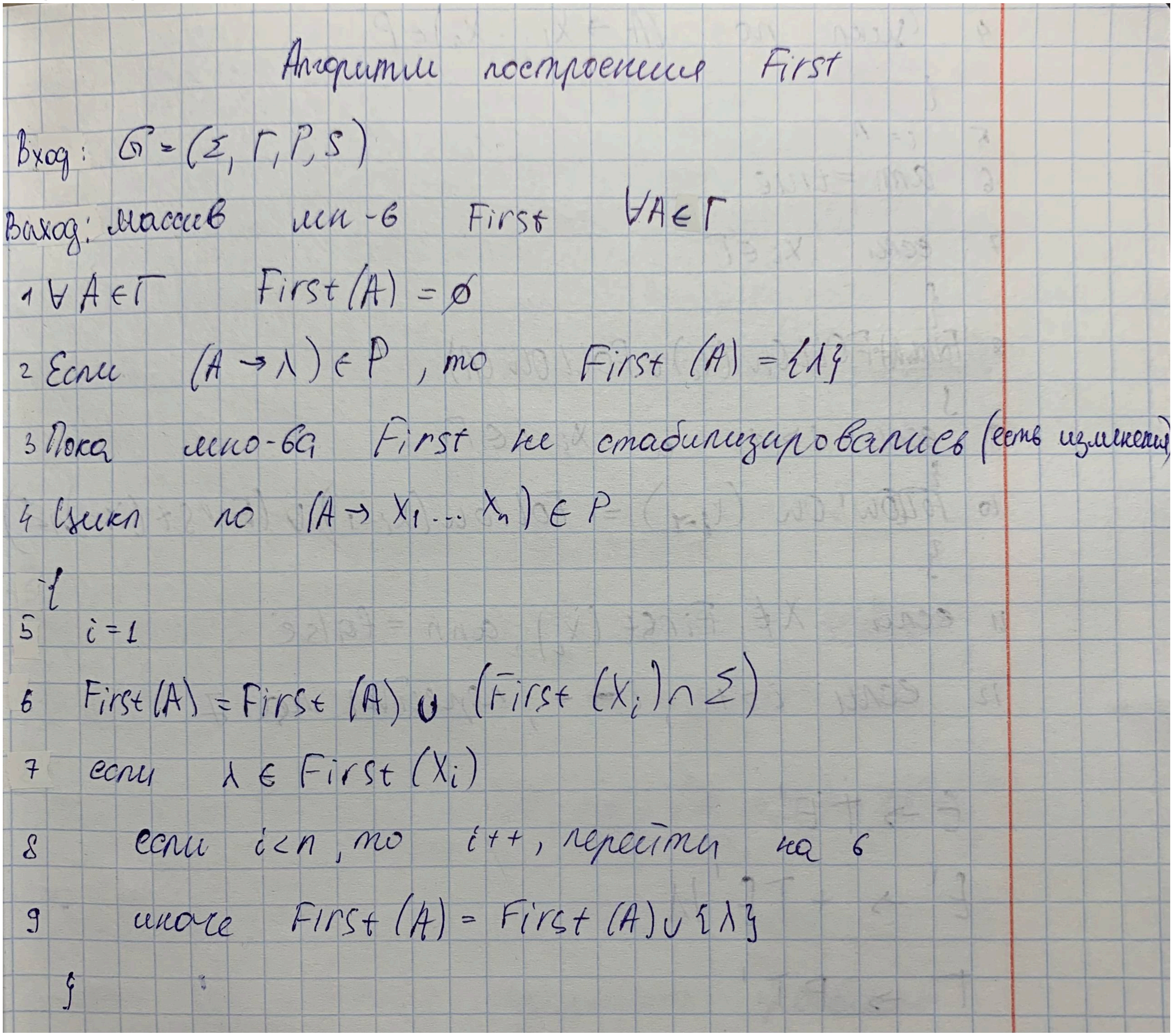
## 11. Множества FIRST, FOLLOW и SELECT. Алгоритмы их построения.

### Множество FIRST

Для **произвольной цепочки** (не обязательно один нетерминал) $\alpha$, множество $FIRST(\alpha)$ — это подмножество терминалов, с которых могут начинаться терминальные цепочки, выводимые из $\alpha$.
Если из $\alpha$ можно вывести $\lambda$, то $\lambda \in FIRST(\alpha)$.



**Алгоритм построения:**



Алгоритм построения First

Вход: $G = (\Sigma, \Gamma, P, S)$

Выход: массив мн-в First $\forall A \in \Gamma$

1 $\forall A \in \Gamma$ $First(A) = \varnothing$

2 Если $(A \to \lambda) \in P$, то $First(A) = \{\lambda\}$

3 Пока мн-ва First не стабилизировались (есть изменения)

4 Цикл по $(A \to X_1 \dots X_n) \in P$

{

5 $i = 1$

6 $First(A) = First(A) \cup (First(X_i) \cap \Sigma)$

7 если $\lambda \in First(X_i)$

8 если $i < n$, то $i{+}{+}$, перейти на 6

9 иначе $First(A) = First(A) \cup \{\lambda\}$

}

**Текстом:**

1. Для каждого терминала $a$: $FIRST(a) = \{a\}$.
2. Для каждого нетерминала $A$: если есть правило $A \to \lambda$, добавить $\lambda$ в $FIRST(A)$.
3. Итеративно для каждого правила $A \to X_1 X_2 ... X_n$:
   - Добавлять в $FIRST(A)$ все неаннулирующие символы из $FIRST(X_1)$.
   - Если $X_1$ может быть аннулирован ($\lambda \in FIRST(X_1)$), переходить к $FIRST(X_2)$ и так далее.
   - Если все символы в правой части — аннулирующие, добавить $\lambda$ в $FIRST(A)$.

## Множество FOLLOW

Для нетерминала $A$, множество $FOLLOW(A)$ — это подмножество терминалов, которые могут следовать непосредственно за $A$ в цепочках, выводимых из аксиомы.

### Определение

Множество FOLLOW($A$) — подмножество $\Sigma \cup \{\dashv\}$ такое, что:
$a \in \text{FOLLOW}(A) \Leftrightarrow S \Rightarrow^* \alpha A a \beta$;
$\dashv \in \text{FOLLOW}(A) \Leftrightarrow S \Rightarrow^* \alpha A$.

Для грамматики $S \to AC, A \to abC | bB, B \to b, C \to c | \lambda$:
$\text{FOLLOW}(A) = \{c, \dashv\}$, $\text{FOLLOW}(B) = \{c, \dashv\}$

**Алгоритм построения:**

Алгоритм построение FOLLOW

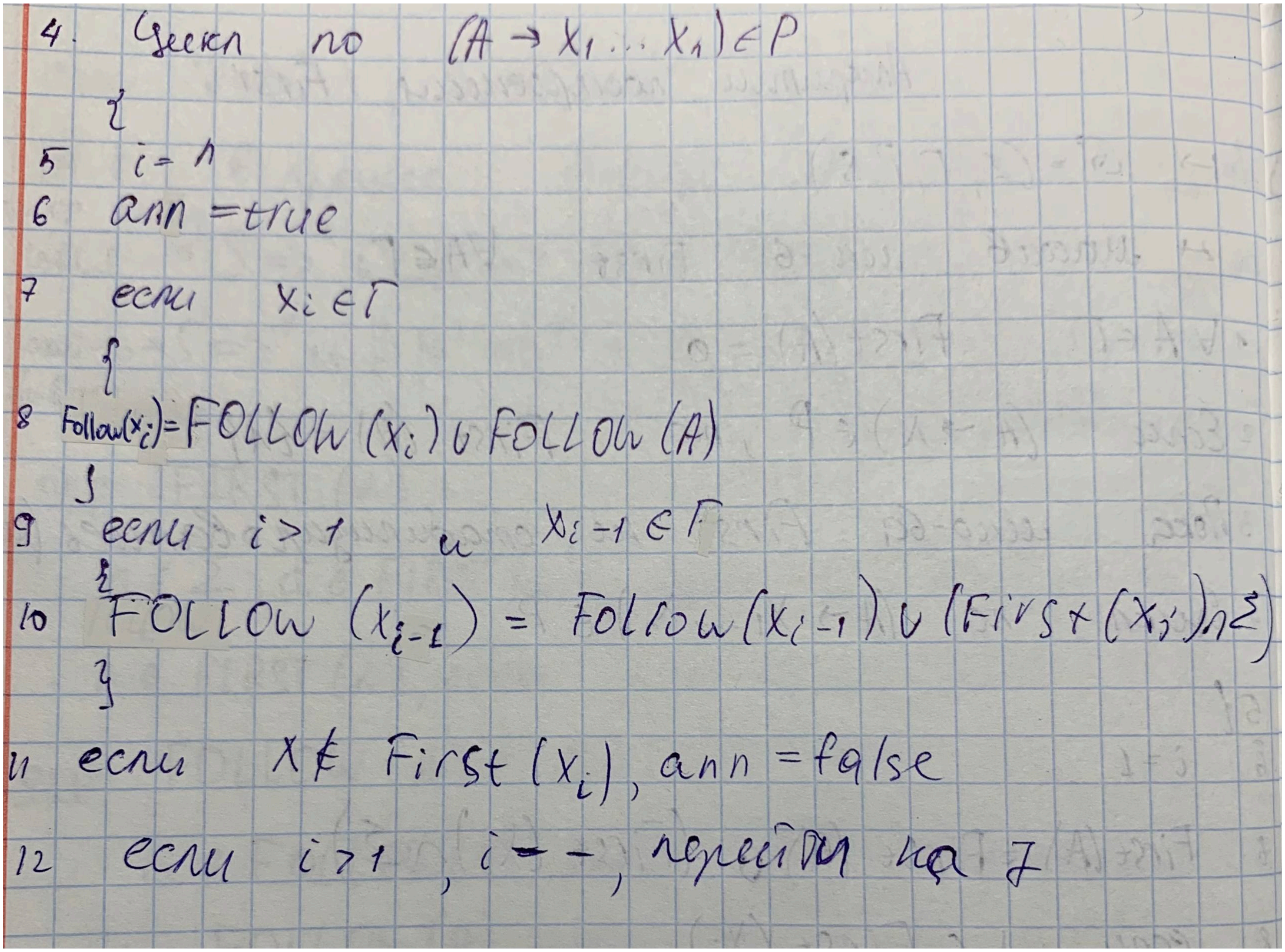
Вход: $G = (\Sigma, \Gamma, P, S)$

Выход: массив $\text{FOLLOW}(A)$ $\forall A \in \Gamma$.

1 $\forall A \in \Gamma$ $\text{FOLLOW}(A) = \varnothing$

2 $\text{FOLLOW}(S) = \{\dashv\}$

3 Пока FOLLOW не стабилизарует. (есть изменение)




**Текстом:**

1. Начинаем с аксиомы $S: FOLLOW(S) = \{\dashv\}$.
2. Итеративно просматривать все правила вывода вида $A \to \alpha B \beta$:
   - Все **терминалы** из $FIRST(\beta)$ (кроме $\lambda$) добавляются в $FOLLOW(B)$.
   - Если $\beta$ — пустая цепочка или аннулируемая ($\lambda \in FIRST(\beta)$), то всё множество $FOLLOW(A)$ добавляется в $FOLLOW(B)$.

## Множество SELECT

Множество выбора $SELECT(A \to \gamma)$ определяется для каждого конкретного правила вывода.
Оно указывает, при каких входных символах следует применять данное правило.

**Расчет SELECT:**

- Если $\lambda \notin FIRST(\gamma)$, то $SELECT(A \to \gamma) = FIRST(\gamma)$.
- Иначе $SELECT(A \to \gamma) = (FIRST(\gamma) \setminus \{\lambda\}) \cup FOLLOW(A)$.

Алгоритма на паре не давала.

$$
\begin{array}{lll}
(1) & E & \to TE' \\
(2) & E' & \to +TE' | \lambda \\
(3) & T & \to FT' \\
(4) & T' & \to *FT' | \lambda \\
(5) & F & \to (E) \\
(6) & F & \to x
\end{array}
$$

| | FIRST | FOLLOW |
|---|---|---|
| $E$ | $x, ($ | $), \dashv$ |
| $E'$ | $\lambda, +$ | $), \dashv$ |
| $T$ | $x, ($ | $), +, \dashv$ |
| $T'$ | $\lambda, *$ | $), +, \dashv$ |
| $F$ | $x, ($ | $), *, +, \dashv$ |


| | FIRST | FOLLOW |
|---|---|---|
| $E$ | $x, ($ | $), \dashv$ |
| $E'$ | $\lambda, +$ | $), \dashv$ |
| $T$ | $x, ($ | $), +, \dashv$ |
| $T'$ | $\lambda, *$ | $), +, \dashv$ |
| $F$ | $x, ($ | $), *, +, \dashv$ |


| | SELECT |
|---|---|
| $E \to TE'$ | $x, ($ |
| $E' \to +TE'$ | $+$ |
| $E' \to \lambda$ | $), \dashv$ |
| $T \to FT'$ | $x, ($ |
| $T' \to *FT'$ | $*$ |
| $T' \to \lambda$ | $), +, \dashv$ |
| $F \to (E)$ | $($ |
| $F \to x$ | $x$ |