

**Stefan Sokołowski**

# **JĘZYKI FORMALNE I METODY KOMPILACJI**

**wykład 1**

**Inst. Informatyki Stosowanej, ANS**

**Elbląg, 2024/2025**

# Języki formalne

**JĘZYK ŹRÓDŁOWY**  
(wysokiego poziomu)

**KOMPILATOR**

**JĘZYK DOCEŁOWY**  
(wewnętrzny)

# Języki formalne

**JĘZYK ŹRÓDŁOWY**  
(wysokiego poziomu)

**KOMPILATOR**

**JĘZYK DOCEŁOWY**  
(wewnętrzny)

Kompilator musi „znać” oba języki...

# Języki formalne

JĘZYK ŹRÓDŁOWY  
(wysokiego poziomu)

KOMPILATOR

JĘZYK DOCELOWY  
(wewnętrzny)

Kompilator musi „znać” oba języki...

Jak program komputerowy nauczyć języka?  
Najpierw trzeba ten język ściśle **zdefiniować**.

# Języki formalne

JĘZYK ŹRÓDŁOWY  
(wysokiego poziomu)

KOMPILATOR

JĘZYK DOCELOWY  
(wewnętrzny)

Kompilator musi „znać” oba języki...

Jak program komputerowy nauczyć języka?

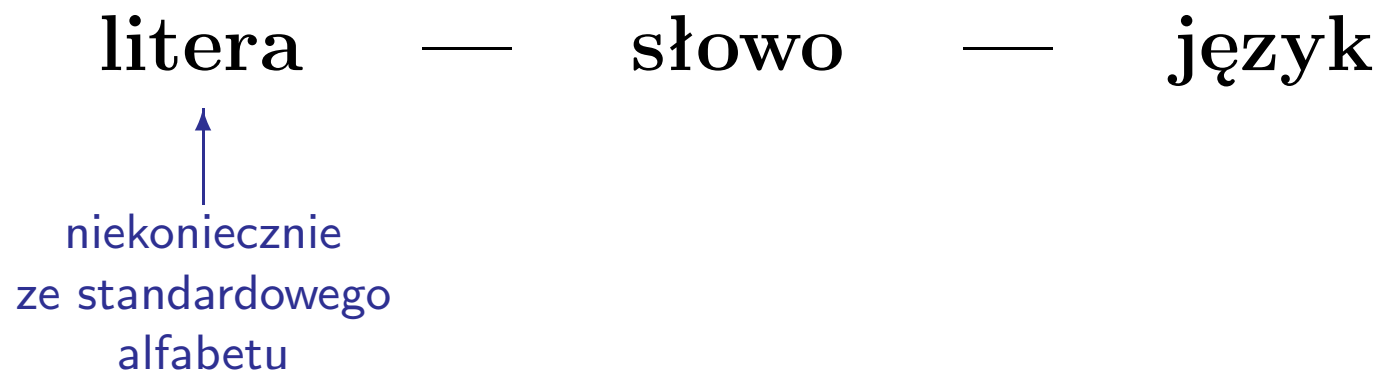
Najpierw trzeba ten język ściśle **zdefiniować**.

*Języki formalne* mają u podstaw sposoby ścisłego definiowania języków.

# Języki formalne

litera — słowo — język

# Języki formalne



# Języki formalne

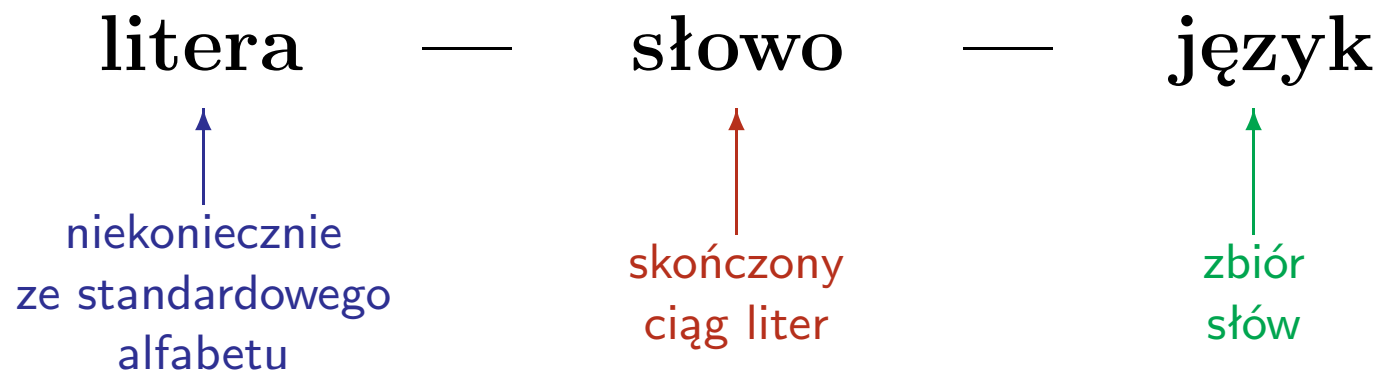




# Języki formalne



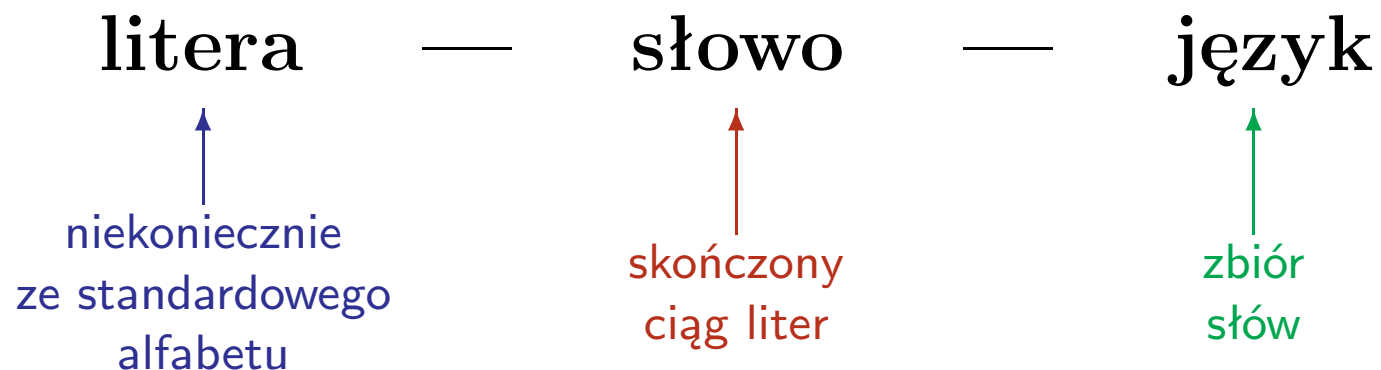
# Języki formalne



**PRZYKŁAD:**

A l a m a k o t a .

# Języki formalne

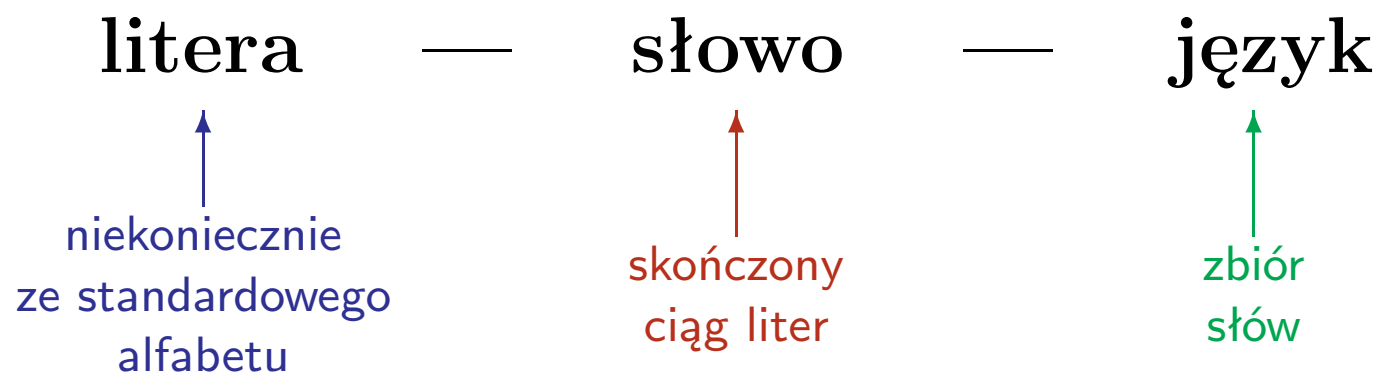


## PRZYKŁAD:

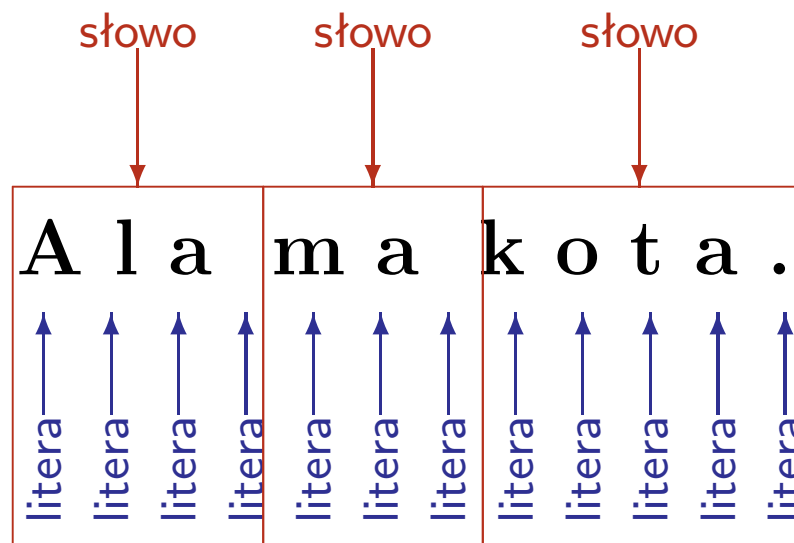
A l a m a k o t a .

Below the text, there are 12 blue arrows pointing upwards, each corresponding to a character in the string "A l a m a k o t a .". Each arrow is labeled "litera" (letter) at its base.

# Języki formalne



## PRZYKŁAD:



# Języki formalne

## PRZYKŁADY:



# Języki formalne

## PRZYKŁADY:



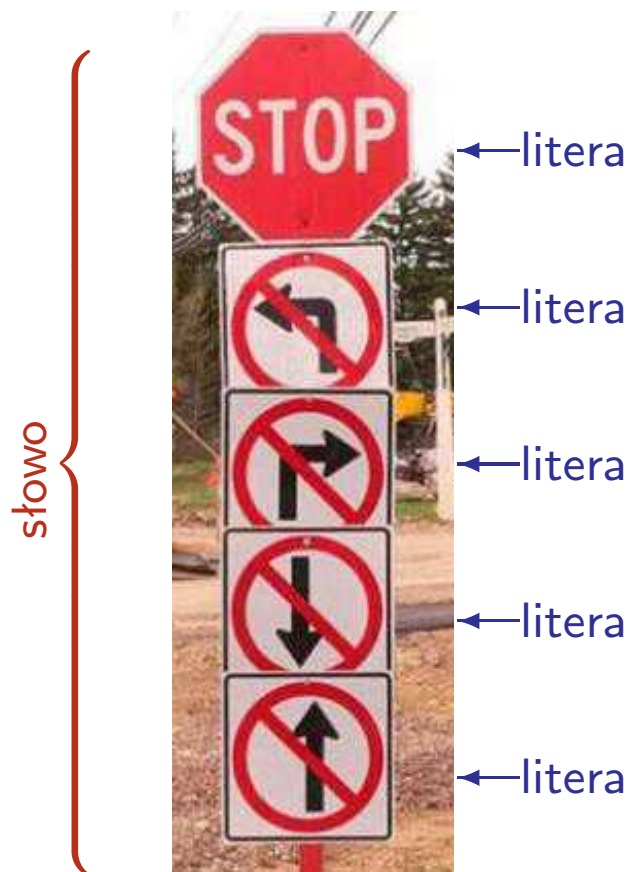
# Języki formalne

## PRZYKŁADY:



# Języki formalne

## PRZYKŁADY:



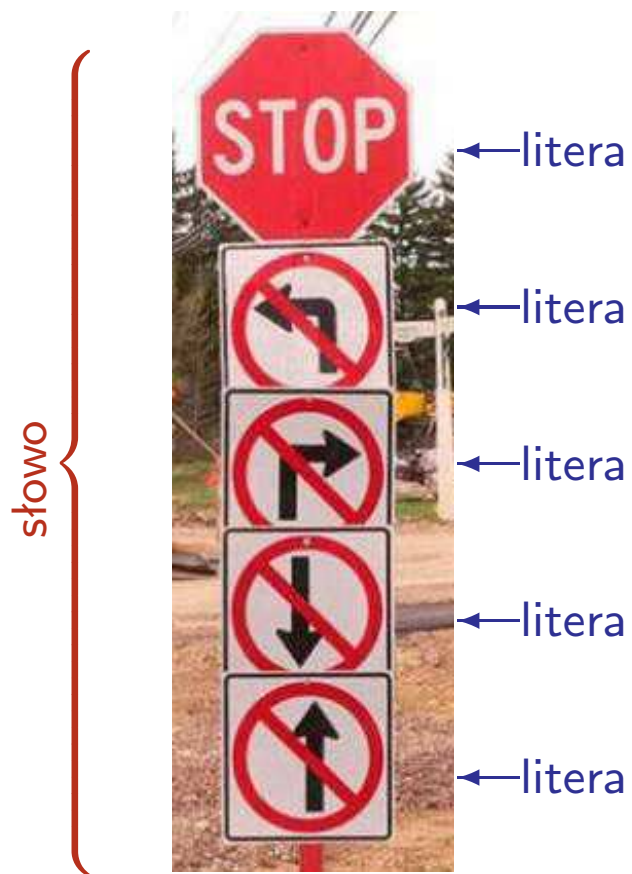
```
class Test {  
    public static void main  
        ( String [ ] args ) {  
        System . out . println  
            ( "To tylko test." ) ;  
    }  
}
```



# Języki formalne

## PRZYKŁADY:

### Przykłady:



litera:

```

class Test {
    public static void main
        ( String [ ] args ) {
        System . out . println
            ( "To tylko test." ) ;
    }
}
  
```

# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

— język (skończony) składający się ze słów języka angielskiego

# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

— język (skończony) składający się ze słów języka angielskiego

$\{0, 1, 10, 11, 100, 101, \dots\}$

— język (nieskończony) składający się z liczb binarnych

# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

— język (skończony) składający się ze słów języka angielskiego

$\{0, 1, 10, 11, 100, 101, \dots\}$

— język (nieskończony) składający się z liczb binarnych

$\{a^n b^n \mid n \geq 0\} = \{\lambda, ab, aabb, aaabbb, \dots\}$

# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

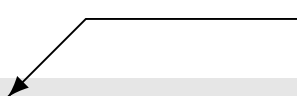
— język (skończony) składający się ze słów języka angielskiego

$\{0, 1, 10, 11, 100, 101, \dots\}$

— język (nieskończony) składający się z liczb binarnych

$\{a^n b^n \mid n \geq 0\} = \{\lambda, ab, aabb, aaabbb, \dots\}$

puste



# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

— język (skończony) składający się ze słów języka angielskiego

$\{0, 1, 10, 11, 100, 101, \dots\}$

— język (nieskończony) składający się z liczb binarnych

$\{a^n b^n \mid n \geq 0\} = \{\lambda, ab, aabb, aaabbb, \dots\}$

puste (to nie jest litera!)

# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

— język (skończony) składający się ze słów języka angielskiego

$\{0, 1, 10, 11, 100, 101, \dots\}$

— język (nieskończony) składający się z liczb binarnych

$\{a^n b^n \mid n \geq 0\} = \{\lambda, ab, aabb, aaabbb, \dots\}$

puste (to nie jest litera!)

— język (nieskończony) składający się z takich słów, w których

- występują tylko litery  $a$  i  $b$

# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

— język (**skończony**) składający się ze słów języka angielskiego

$\{0, 1, 10, 11, 100, 101, \dots\}$

— język (**nieskończony**) składający się z liczb binarnych

$\{a^n b^n \mid n \geq 0\} = \{\lambda, ab, aabb, aaabbb, \dots\}$

puste (**to nie jest litera!**)

— język (**nieskończony**) składający się z takich słów, w których

- występują tylko litery  $a$  i  $b$ ,
- liczba liter  $a$  jest równa liczbie liter  $b$



# Proste języki formalne (przykłady)

$\{a, aback, abandon, abase, \dots, zoom\}$

— język (skończony) składający się ze słów języka angielskiego

$\{0, 1, 10, 11, 100, 101, \dots\}$

— język (nieskończony) składający się z liczb binarnych

$\{a^n b^n \mid n \geq 0\} = \{\lambda, ab, aabb, aaabbb, \dots\}$

puste (to nie jest litera!)

— język (nieskończony) składający się z takich słów, w których

- występują tylko litery  $a$  i  $b$ ,
- liczba liter  $a$  jest równa liczbie liter  $b$ ,
- wszystkie litery  $a$  poprzedzają wszystkie litery  $b$ .

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

*język* — dowolny zbiór słów  $L \subseteq \Sigma^*$

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

*język* — dowolny zbiór słów  $L \subseteq \Sigma^*$

**PRZYKŁAD:** nie mylić! —

$\lambda$  — słowo *puste* (ciąg liter o długości 0)

**Uwaga:** to nie jest litera!

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

*język* — dowolny zbiór słów  $L \subseteq \Sigma^*$

## PRZYKŁAD: nie mylić! —

$\lambda$  — słowo *puste* (ciąg liter o długości 0)

$\emptyset \stackrel{\text{def}}{=} \{ \}$  — język pusty (nie zawierający żadnego słowa)

**Uwaga:** to nie jest litera!

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

*język* — dowolny zbiór słów  $L \subseteq \Sigma^*$

## PRZYKŁAD: nie mylić! —

$\lambda$  — słowo *puste* (ciąg liter o długości 0)

**Uwaga:** to nie jest litera!

$\emptyset \stackrel{\text{def}}{=} \{ \} — \text{język pusty (nie zawierający żadnego słowa)}$

$\Lambda \stackrel{\text{def}}{=} \{ \lambda \} — \text{język zawierający wyłącznie słowo puste}$



# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

*język* — dowolny zbiór słów  $L \subseteq \Sigma^*$

## PRZYKŁAD: nie mylić! —

$\lambda$  — słowo *puste* (ciąg liter o długości 0)

**Uwaga:** to nie jest litera!

$\emptyset \stackrel{\text{def}}{=} \{ \}$  — język pusty (nie zawierający żadnego słowa)

$\Lambda \stackrel{\text{def}}{=} \{ \lambda \}$  — język zawierający wyłącznie słowo puste

## PRZYKŁAD:

$1(*2 - 1)(1 - (2 - *1))()$  — słowo nad alfabetem  $\Sigma \stackrel{\text{def}}{=} \{ (, ), -, *, 1, 2 \}$

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

*język* — dowolny zbiór słów  $L \subseteq \Sigma^*$

## PRZYKŁAD: nie mylić! —

$\lambda$  — słowo *puste* (ciąg liter o długości 0)

**Uwaga:** to nie jest litera!

$\emptyset \stackrel{\text{def}}{=} \{ \}$  — język pusty (nie zawierający żadnego słowa)

$\Lambda \stackrel{\text{def}}{=} \{ \lambda \}$  — język zawierający wyłącznie słowo puste

## PRZYKŁAD:

$1(*2 - 1)(1 - (2 - *1))()$  — słowo nad alfabetem  $\Sigma \stackrel{\text{def}}{=} \{ (, ), -, *, 1, 2 \}$

$\{ \lambda, a, b, ab, ba \}$  — język nad alfabetem  $\Sigma \stackrel{\text{def}}{=} \{ a, b \}$  zawierający pięć słów

# Języki formalne

## DEFINICJE:

$\Sigma$  — skończony niepusty *alfabet*, czyli zbiór tzw. *liter*

*słowo* — dowolny ciąg skończonej długości o elementach z  $\Sigma$

$\Sigma^*$  — zbiór **wszystkich** słów

*język* — dowolny zbiór słów  $L \subseteq \Sigma^*$

## PRZYKŁAD: nie mylić! —

$\lambda$  — słowo *puste* (ciąg liter o długości 0)

**Uwaga:** to nie jest litera!

$\emptyset \stackrel{\text{def}}{=} \{ \}$  — język pusty (nie zawierający żadnego słowa)

$\Lambda \stackrel{\text{def}}{=} \{ \lambda \}$  — język zawierający wyłącznie słowo puste

## PRZYKŁAD:

$1(*2 - 1)(1 - (2 - *1))()$  — słowo nad alfabetem  $\Sigma \stackrel{\text{def}}{=} \{ (, ), -, *, 1, 2 \}$

$\{ \lambda, a, b, ab, ba \}$  — język nad alfabetem  $\Sigma \stackrel{\text{def}}{=} \{ a, b \}$  zawierający pięć słów

$\{ a^n \mid n \geq 0 \} = \{ \lambda, a, aa, aaa, \dots \}$  — jęz. nieskończ. nad alf.  $\Sigma \stackrel{\text{def}}{=} \{ a \}$

# Jak **porządnie** opisać język nieskończony?

# Jak **porządnie** opisać język nieskończony?

- $\{\lambda, ab, aabb, aaabbb, \dots\}$  — co oznaczają kropeczki?

# Jak **porządnie** opisać język nieskończony?

- $\{\lambda, ab, aabb, aaabbb, \dots\}$  — co oznaczają kropeczki?
- $\{a^n b^n \mid n \geq 0\}$  — lepiej, ale są niepożądane elementy obce (liczby, zmienne, relacje, ...)

# Jak **porządnie** opisać język nieskończony?

- $\{\lambda, ab, aabb, aaabbb, \dots\}$  — co oznaczają kropeczki?
  - $\{a^n b^n \mid n \geq 0\}$  — lepiej, ale są niepożądane elementy obce (liczby, zmienne, relacje, ...)
- 

Język definiujemy na dwa sposoby. Przez

- **generator** — cokolwiek **wyprodukuje**, uważamy za należące do języka

# Jak **porządnie** opisać język nieskończony?

- $\{\lambda, ab, aabb, aaabbb, \dots\}$  — co oznaczają kropeczki?
  - $\{a^n b^n \mid n \geq 0\}$  — lepiej, ale są niepożądane elementy obce (liczby, zmienne, relacje, ...)
- 

Język definiujemy na dwa sposoby. Przez

- **generator** — cokolwiek **wyprodukuje**, uważamy za należące do języka
- **akceptor** — cokolwiek **zatwierdzi**, uważamy za należące do języka



# Jak **porządnie** opisać język nieskończony?

- $\{\lambda, ab, aabb, aaabbb, \dots\}$  — co oznaczają kropeczki?
  - $\{a^n b^n \mid n \geq 0\}$  — lepiej, ale są niepożądane elementy obce (liczby, zmienne, relacje, ...)
- 

Język definiujemy na dwa sposoby. Przez

- **generator** — cokolwiek **wyprodukuje**, uważamy za należące do języka
- **akceptor** — cokolwiek **zatwierdzi**, uważamy za należące do języka

Opis języka programowania przeznaczony dla programisty powinien być **generatorem** — ma wyjaśniać, jak pisać poprawne programy.

# Jak **porządnie** opisać język nieskończony?

- $\{\lambda, ab, aabb, aaabbb, \dots\}$  — co oznaczają kropeczki?
  - $\{a^n b^n \mid n \geq 0\}$  — lepiej, ale są niepożądane elementy obce (liczby, zmienne, relacje, ...)
- 

Język definiujemy na dwa sposoby. Przez

- **generator** — cokolwiek **wyprodukuje**, uważamy za należące do języka
- **akceptor** — cokolwiek **zatwierdzi**, uważamy za należące do języka

Opis języka programowania przeznaczony dla programisty powinien być **generatorem** — ma wyjaśniać, jak pisać poprawne programy.

Kompilator języka programowania powinien zawierać **akceptor** — sprawdzający, czy program na jego wejściu należy do języka.

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ )

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ );
- skończony zbiór  $P$  *produkcji* postaci  $A \rightarrow w$ , gdzie  $A \in N$  (czyli jest pojedynczym nieterminalem)

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ );
- skończony zbiór  $P$  *produkcji* postaci  $A \rightarrow w$ , gdzie  
     $A \in N$  (czyli jest pojedynczym nieterminalem),  
     $w \in (N \cup \Sigma)^*$  (czyli jest słowem z terminali i nieterminali)

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ );
- skończony zbiór  $P$  *produkcji* postaci  $A \rightarrow w$ , gdzie  
     $A \in N$  (czyli jest pojedynczym nieterminalem),  
     $w \in (N \cup \Sigma)^*$  (czyli jest słowem z terminali i nieterminali);
- wybrany nieterminal  $S \in N$  nazywany *nieterminalem początkowym* albo *aksjomatem* gramatyki.



# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ );
- skończony zbiór  $P$  *produkcji* postaci  $A \rightarrow w$ , gdzie  
     $A \in N$  (czyli jest pojedynczym nieterminalem),  
     $w \in (N \cup \Sigma)^*$  (czyli jest słowem z terminali i nieterminali);
- wybrany nieterminal  $S \in N$  nazywany *nieterminalem początkowym* albo *aksjomatem* gramatyki.

## PRZYKŁAD:

Terminale:  $a, b$

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ );
- skończony zbiór  $P$  *produkcji* postaci  $A \rightarrow w$ , gdzie  
 $A \in N$  (czyli jest pojedynczym nieterminalem),  
 $w \in (N \cup \Sigma)^*$  (czyli jest słowem z terminali i nieterminali);
- wybrany nieterminal  $S \in N$  nazywany *nieterminalem początkowym* albo *aksjomatem* gramatyki.

## PRZYKŁAD:

Terminale:  $a, b$

Nieterminale:  $X$

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ );
- skończony zbiór  $P$  *produkcji* postaci  $A \rightarrow w$ , gdzie  
 $A \in N$  (czyli jest pojedynczym nieterminalem),  
 $w \in (N \cup \Sigma)^*$  (czyli jest słowem z terminali i nieterminali);
- wybrany nieterminal  $S \in N$  nazywany *nieterminalem początkowym* albo *aksjomatem* gramatyki.

## PRZYKŁAD:

Terminale:  $a, b$

Nieterminale:  $X$

Aksjomat:  $X$

# Generatory: gramatyki bezkontekstowe

**DEFINICJA:** Gramatyka bezkontekstowa  $G$  — czwórka  $\langle \Sigma, N, P, S \rangle$

- alfabet *terminalny*  $\Sigma$ ;
- alfabet *nieterminalny* lub *pomocniczy*  $N$  (rozłączny z  $\Sigma$ );
- skończony zbiór  $P$  *produkcji* postaci  $A \rightarrow w$ , gdzie  
 $A \in N$  (czyli jest pojedynczym nieterminalem),  
 $w \in (N \cup \Sigma)^*$  (czyli jest słowem z terminali i nieterminali);
- wybrany nieterminal  $S \in N$  nazywany *nieterminalem początkowym* albo *aksjomatem* gramatyki.

## PRZYKŁAD:

Terminale:  $a, b$

Nieterminale:  $X$

Aksjomat:  $X$

Produkcje:

|     |               |           |
|-----|---------------|-----------|
| $X$ | $\rightarrow$ | $\lambda$ |
| $X$ | $\rightarrow$ | $aXa$     |
| $X$ | $\rightarrow$ | $bXb$     |

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uuv$  czyli  
 $uuv$  jest *bezpośrednio wywodliwe* z  $uXv$



w  $P$  jest produkcja  
 $X \rightarrow w$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uwv$  czyli  
 $uwv$  jest *bezpośrednio wywodliwe* z  $uXv$



w  $P$  jest produkcja  
 $X \rightarrow w$

$w_1 \Rightarrow_G^* w_n$  czyli  
 $w_n$  jest *wywodliwe* z  $w_1$



$w_1 \Rightarrow_G w_2 \Rightarrow_G w_3$   
 $\vdots$   
 $\Rightarrow_G w_{n-1} \Rightarrow_G w_n$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$$uXv \Rightarrow_G uwv \quad \text{czyli} \quad uwv \text{ jest } \textit{bezpośrednio wywodliwe} \text{ z } uXv \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} \text{w } P \text{ jest produkcja} \\ X \rightarrow w \end{array}$$

$$\begin{array}{l} w_1 \Rightarrow_G^* w_n \quad \text{czyli} \\ w_n \text{ jest } \textit{wywodliwe} \text{ z } w_1 \end{array} \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} w_1 \Rightarrow_G w_2 \Rightarrow_G w_3 \\ \vdots \\ \Rightarrow_G w_{n-1} \Rightarrow_G w_n \end{array}$$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  —  
 składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .



# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$$uXv \Rightarrow_G uwv \quad \text{czyli} \quad uwv \text{ jest bezpośrednio wywodliwe z } uXv \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} \text{w } P \text{ jest produkcja} \\ X \rightarrow w \end{array}$$

$$\begin{array}{l} w_1 \Rightarrow_G^* w_n \quad \text{czyli} \\ w_n \text{ jest wywodliwe z } w_1 \end{array} \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} w_1 \Rightarrow_G w_2 \Rightarrow_G w_3 \\ \vdots \\ \Rightarrow_G w_{n-1} \Rightarrow_G w_n \end{array}$$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  —  
 składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow aXa \\ X \rightarrow bXb \end{array}$$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$$uXv \Rightarrow_G uwv \quad \text{czyli} \quad uwv \text{ jest bezpośrednio wywodliwe z } uXv \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} \text{w } P \text{ jest produkcja} \\ X \rightarrow w \end{array}$$

$$\begin{array}{l} w_1 \Rightarrow_G^* w_n \quad \text{czyli} \\ w_n \text{ jest wywodliwe z } w_1 \end{array} \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} w_1 \Rightarrow_G w_2 \Rightarrow_G w_3 \\ \vdots \\ \Rightarrow_G w_{n-1} \Rightarrow_G w_n \end{array}$$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  —  
 składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow aXa \\ X \rightarrow bXb \end{array}$$

$X$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uwv$  czyli  
 $uwv$  jest *bezpośrednio wywodliwe* z  $uXv$

def  
 $\Leftrightarrow$

w  $P$  jest produkcja  
 $X \rightarrow w$

$w_1 \Rightarrow_G^* w_n$  czyli  
 $w_n$  jest *wywodliwe* z  $w_1$

def  
 $\Leftrightarrow$

$w_1 \Rightarrow_G w_2 \Rightarrow_G w_3$   
 $\vdots$   
 $\Rightarrow_G w_{n-1} \Rightarrow_G w_n$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  —

składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$X \rightarrow \lambda$   
 $X \rightarrow aXa$   
 $X \rightarrow bXb$

$X \Rightarrow bXb$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uuv$  czyli  
 $uuv$  jest *bezpośrednio wywodliwe* z  $uXv$

def  
 $\Leftrightarrow$

w  $P$  jest produkcja  
 $X \rightarrow w$

$w_1 \Rightarrow_G^* w_n$  czyli  
 $w_n$  jest *wywodliwe* z  $w_1$

def  
 $\Leftrightarrow$

$w_1 \Rightarrow_G w_2 \Rightarrow_G w_3$   
 $\vdots$   
 $\Rightarrow_G w_{n-1} \Rightarrow_G w_n$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  —

składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$X \rightarrow \lambda$   
 $X \rightarrow aXa$   
 $X \rightarrow bXb$

$X \Rightarrow bXb \Rightarrow bbXbb$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$$uXv \Rightarrow_G uwv \quad \text{czyli} \quad uwv \text{ jest bezpośrednio wywodliwe z } uXv \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} \text{w } P \text{ jest produkcja} \\ X \rightarrow w \end{array}$$

$$\begin{array}{l} w_1 \Rightarrow_G^* w_n \quad \text{czyli} \\ w_n \text{ jest wywodliwe z } w_1 \end{array} \quad \stackrel{\text{def}}{\iff} \quad \begin{array}{l} w_1 \Rightarrow_G w_2 \Rightarrow_G w_3 \\ \vdots \\ \Rightarrow_G w_{n-1} \Rightarrow_G w_n \end{array}$$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  —  
 składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow aXa \\ X \rightarrow bXb \end{array}$$

$$X \Rightarrow bXb \Rightarrow bbXbb \Rightarrow bbaXabb$$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uuv$  czyli  $uuv$  jest *bezpośrednio wywodliwe* z  $uXv$   $\iff$  w  $P$  jest produkcja  $X \rightarrow w$

$w_1 \Rightarrow_G^* w_n$  czyli  $w_n$  jest *wywodliwe* z  $w_1$   $\iff$   $w_1 \Rightarrow_G w_2 \Rightarrow_G w_3 \Rightarrow_G \dots \Rightarrow_G w_{n-1} \Rightarrow_G w_n$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  — składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$X \rightarrow \lambda$   
 $X \rightarrow aXa$   
 $X \rightarrow bXb$

$X \Rightarrow bXb \Rightarrow bbXbb \Rightarrow bbaXabb \Rightarrow bbaabb$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uuv$  czyli  $uuv$  jest *bezpośrednio wywodliwe* z  $uXv$   $\iff$  w  $P$  jest produkcja  $X \rightarrow w$

$w_1 \Rightarrow_G^* w_n$  czyli  $w_n$  jest *wywodliwe* z  $w_1$   $\iff$   $w_1 \Rightarrow_G w_2 \Rightarrow_G w_3 \Rightarrow_G \dots \Rightarrow_G w_{n-1} \Rightarrow_G w_n$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  — składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$X \rightarrow \lambda$   
 $X \rightarrow aXa$   
 $X \rightarrow bXb$

$X \Rightarrow bXb \Rightarrow bbXbb \Rightarrow bbaXabb \Rightarrow bbaabb$

# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uuv$  czyli  $uuv$  jest *bezpośrednio wywodliwe* z  $uXv$   $\iff$  w  $P$  jest produkcja  $X \rightarrow w$

$w_1 \Rightarrow_G^* w_n$  czyli  $w_n$  jest *wywodliwe* z  $w_1$   $\iff$   $w_1 \Rightarrow_G w_2 \Rightarrow_G w_3 \Rightarrow_G \dots \Rightarrow_G w_{n-1} \Rightarrow_G w_n$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  — składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$X \rightarrow \lambda$   
 $X \rightarrow aXa$   
 $X \rightarrow bXb$

$X \Rightarrow_G^* bbaabb \in L(G)$

$X \Rightarrow bXb \Rightarrow bbXbb \Rightarrow bbaXabb \Rightarrow bbaabb$



# Jak gramatyka definiuje język?

**DEFINICJA:** Wywodliwość —

$uXv \Rightarrow_G uuv$  czyli  
 $uuv$  jest *bezpośrednio wywodliwe* z  $uXv$

def  
 $\Longleftrightarrow$

w  $P$  jest produkcja  
 $X \rightarrow w$

$w_1 \Rightarrow_G^* w_n$  czyli  
 $w_n$  jest *wywodliwe* z  $w_1$

def  
 $\Longleftrightarrow$

$w_1 \Rightarrow_G w_2 \Rightarrow_G w_3$   
 $\vdots$   
 $\Rightarrow_G w_{n-1} \Rightarrow_G w_n$

**DEFINICJA:** Język  $L(G)$  generowany przez gramatykę  $G$  —

składa się z tych słów wywodliwych z **aksjomatu**, które już nie zawierają nieterminali:  $L(G) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}$ .

**PRZYKŁAD:**

Gramatyka:

$X \rightarrow \lambda$   
 $X \rightarrow aXa$   
 $X \rightarrow bXb$

$X \Rightarrow_G^* bbaabb \in L(G)$

**LEMAT:**

$L(G) = \{ww^R \mid w \in \{a,b\}^*\}$

$X \Rightarrow bXb \Rightarrow bbXbb \Rightarrow bbaXabb \Rightarrow bbaabb$

# Jak gramatyka definiuje język?

**NIE MYLIĆ**

# Jak gramatyka definiuje język?

## NIE MYLIĆ:

→ — formalny symbol w produkcji w gramatyce

# Jak gramatyka definiuje język?

## NIE MYLIĆ:

- $\rightarrow$  — formalny symbol w produkcji w gramatyce
- $\Rightarrow$  — relacja bezpośredniej wywodliwości:  
jeśli  $X \rightarrow w$  to  $X \Rightarrow w$

# Jak gramatyka definiuje język?

## NIE MYLIĆ:

- $\rightarrow$  — formalny symbol w produkcji w gramatyce
- $\Rightarrow$  — relacja bezpośredniej wywodliwości:  
jeśli  $X \rightarrow w$  to  $X \Rightarrow w$   
ale np.  $0X1 \Rightarrow 00X11$

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow 0X1 \end{array}$$

# Jak gramatyka definiuje język?

## NIE MYLIĆ:

- — formalny symbol w produkcji w gramatyce
- ⇒ — relacja bezpośredniej wywodliwości:  
 jeśli  $X \rightarrow w$  to  $X \Rightarrow w$   
 ale np.  $0X1 \Rightarrow 00X11$   
 a **nieprawda**, że  $0X1 \rightarrow 00X11$

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow 0X1 \end{array}$$

# Jak gramatyka definiuje język?

## NIE MYLIĆ:

$\rightarrow$  — formalny symbol w produkcji w gramatyce

$\Rightarrow$  — relacja bezpośredniej wywodliwości:

jeśli  $X \rightarrow w$  to  $X \Rightarrow w$

ale np.  $0X1 \Rightarrow 00X11$

a **nieprawda, że**  $0X1 \rightarrow 00X11$

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow 0X1 \end{array}$$

$\Rightarrow^*$  — relacja wywodliwości:

jeśli  $w_1 \Rightarrow w_2$  to  $w_1 \Rightarrow^* w_2$

# Jak gramatyka definiuje język?

## NIE MYLIĆ:

$\rightarrow$  — formalny symbol w produkcji w gramatyce

$\Rightarrow$  — relacja bezpośredniej wywodliwości:

jeśli  $X \rightarrow w$  to  $X \Rightarrow w$

ale np.  $0X1 \Rightarrow 00X11$

a **nieprawda**, że  $0X1 \rightarrow 00X11$

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow 0X1 \end{array}$$

$\Rightarrow^*$  — relacja wywodliwości:

jeśli  $w_1 \Rightarrow w_2$  to  $w_1 \Rightarrow^* w_2$

ale np.  $X \Rightarrow^* 0011$



# Jak gramatyka definiuje język?

## NIE MYLIĆ:

$\rightarrow$  — formalny symbol w produkcji w gramatyce

$\Rightarrow$  — relacja bezpośredniej wywodliwości:

jeśli  $X \rightarrow w$  to  $X \Rightarrow w$

ale np.  $0X1 \Rightarrow 00X11$

a **nieprawda**, że  $0X1 \rightarrow 00X11$

$$\begin{array}{l} X \rightarrow \lambda \\ X \rightarrow 0X1 \end{array}$$

$\Rightarrow^*$  — relacja wywodliwości:

jeśli  $w_1 \Rightarrow w_2$  to  $w_1 \Rightarrow^* w_2$

ale np.  $X \Rightarrow^* 0011$

a **nieprawda**, że  $X \Rightarrow 0011$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{ll} E & \rightarrow T \\ E & \rightarrow E + T \\ T & \rightarrow F \\ T & \rightarrow T * F \\ F & \rightarrow a \\ F & \rightarrow ( E ) \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl}
 E & \rightarrow & T \\
 E & \rightarrow & E + T \\
 T & \rightarrow & F \\
 T & \rightarrow & T * F \\
 F & \rightarrow & a \\
 F & \rightarrow & ( E )
 \end{array}$$


$$E \Rightarrow E + T$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$E \Rightarrow E + T \Rightarrow T + T$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$E \Rightarrow E + T \Rightarrow T + T \Rightarrow F + T$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$E \Rightarrow E + T \Rightarrow T + T \Rightarrow \boxed{F} + T \Rightarrow \boxed{a} + T$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$\begin{array}{l} E \Rightarrow E + T \Rightarrow T + T \Rightarrow F + T \Rightarrow a + T \\ \Rightarrow a + T * F \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$\begin{array}{l} E \Rightarrow E + T \Rightarrow T + T \Rightarrow F + T \Rightarrow a + T \\ \Rightarrow a + T * F \Rightarrow a + F * F \end{array}$$



# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$\begin{array}{l} E \Rightarrow E + T \Rightarrow T + T \Rightarrow F + T \Rightarrow a + T \\ \Rightarrow a + T * F \Rightarrow a + F * F \Rightarrow a + a * F \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$\begin{array}{l} E \Rightarrow E + T \Rightarrow T + T \Rightarrow F + T \Rightarrow a + T \\ \Rightarrow a + T * F \Rightarrow a + F * F \Rightarrow a + a * F \\ \Rightarrow a + a * a \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{ll} E & \rightarrow T \\ E & \rightarrow E + T \\ T & \rightarrow F \\ T & \rightarrow T * F \\ F & \rightarrow a \\ F & \rightarrow ( E ) \end{array}$$

$$\begin{aligned} E &\Rightarrow E + T \Rightarrow T + T \Rightarrow F + T \Rightarrow a + T \\ &\Rightarrow a + T * F \Rightarrow a + F * F \Rightarrow a + a * F \\ &\Rightarrow a + a * a \in L(G) \end{aligned}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$E \Rightarrow T$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$

$$E \Rightarrow T \Rightarrow F$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{ll} E & \rightarrow T \\ E & \rightarrow E + T \\ T & \rightarrow F \\ T & \rightarrow T * F \\ F & \rightarrow a \\ F & \rightarrow ( E ) \end{array}$$


$$E \Rightarrow T \Rightarrow F \Rightarrow (E)$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl}
 E & \rightarrow & T \\
 E & \rightarrow & E + T \\
 T & \rightarrow & F \\
 T & \rightarrow & T * F \\
 F & \rightarrow & a \\
 F & \rightarrow & ( E )
 \end{array}$$


$$E \Rightarrow T \Rightarrow F \Rightarrow (E) \Rightarrow (E + T)$$



# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$


$$E \Rightarrow T \Rightarrow F \Rightarrow ( E ) \Rightarrow ( E + T ) \Rightarrow ( T + T )$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$


$$\begin{array}{l} E \Rightarrow T \Rightarrow F \Rightarrow ( E ) \Rightarrow ( E + T ) \Rightarrow ( T + T ) \\ \quad \Rightarrow ( F + T ) \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$\begin{array}{l} E \Rightarrow T \Rightarrow F \Rightarrow ( E ) \Rightarrow ( E + T ) \Rightarrow ( T + T ) \\ \quad \Rightarrow ( \textcolor{yellow}{F} + T ) \Rightarrow ( \textcolor{yellow}{a} + T ) \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$\begin{array}{l} E \Rightarrow T \Rightarrow F \Rightarrow ( E ) \Rightarrow ( E + T ) \Rightarrow ( T + T ) \\ \quad \Rightarrow ( F + T ) \Rightarrow ( a + T ) \Rightarrow ( a + F ) \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$



$$\begin{array}{l} E \Rightarrow T \Rightarrow F \Rightarrow ( E ) \Rightarrow ( E + T ) \Rightarrow ( T + T ) \\ \Rightarrow ( F + T ) \Rightarrow ( a + T ) \Rightarrow ( a + F ) \\ \Rightarrow ( a + a ) \end{array}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{lcl} E & \rightarrow & T \\ E & \rightarrow & E + T \\ T & \rightarrow & F \\ T & \rightarrow & T * F \\ F & \rightarrow & a \\ F & \rightarrow & ( E ) \end{array}$$

$$\begin{aligned} E &\Rightarrow T \Rightarrow F \Rightarrow ( E ) \Rightarrow ( E + T ) \Rightarrow ( T + T ) \\ &\Rightarrow ( F + T ) \Rightarrow ( a + T ) \Rightarrow ( a + F ) \\ &\Rightarrow ( a + a ) \in L(G) \end{aligned}$$

# Jak gramatyka definiuje język?

## PRZYKŁAD:

Gramatyka  $G$ :

$$\begin{array}{ll} E & \rightarrow T \\ E & \rightarrow E + T \\ T & \rightarrow F \\ T & \rightarrow T * F \\ F & \rightarrow a \\ F & \rightarrow ( E ) \end{array}$$

$$\begin{aligned} E &\Rightarrow T \Rightarrow F \Rightarrow ( E ) \Rightarrow ( E + T ) \Rightarrow ( T + T ) \\ &\Rightarrow ( F + T ) \Rightarrow ( a + T ) \Rightarrow ( a + F ) \\ &\Rightarrow ( a + a ) \in L(G) \end{aligned}$$

$L(G)$  = wyrażenia *arytmetyczne* zbudowane z  $a, +, *, (, )$