

□字句解析(Lexical Analysis)

□文 (プログラム) を単語に分ける。

→文の中の単語を一つ一つ「辞書」を引くことに似ている (パターンマッチング)

→"Ihaveapen."

□"abc=123+ cde"

□「辞書」の内容

○0,1,2,3, ...→数字

○a, aa, aaa, ... ab, ... a1.. → 識別子 (名前)

○= → 演算子=

○+ → 演算子+

○タブ文字、空白文字、改行文字 (列) など → 空白空(White Space)

□単語 (これを字句、(あるいは符、トークンなどとも) という) の列に変換

[("abc", 識別子), ("=", 演算子=), ("123", 数字), ("+", 演算子+), ("efg", 識別子)]

→White Space は通常省略される。

□留意点

□処理を行うときは、先頭から一文字ずつ取り出しながら行う。コンピュータは全体を同時に見渡せない。

□最長一致、a も、ab も識別子の辞書に含まれているけれど、abc で一つの識別子として認識する。

→数字や識別子のようなものは、どこまでが単語かは少なくとも一つ先まで調べて見ないとわからない。

□p5 補足 :

○拡張BNF記法 (正規右辺文法) →第2回配布スライド p

<名前> → <英字> {<英字> | <数字> }

これは、拡張BNF記法 (正規右辺文法とも) による文法の表し方。

(1) X → <英字> | <数字>

(2) <名前> → <英字> {X}

(2)は、 <名前> → <英字> | <英字>X | <英字>XX | <英字>XXX | ...の意味。

すなわち、L(<名前>)= {a,b,... , aa, ... a1, a2, b1, b2 aa1, ...}

○拡張BNF記法は、普通のBNFに書き換えが可能。

(2)で、さらに非終端記号 A を導入して

<名前> → <英字>A

A → ε | XA

とすればよい。

○状態遷移図による字句解析器の考え方

◎=最終状態：この場合<名前>の読み取りが終わった状態

(A)→(B) = Aの状態にいるときに a を読み取ったら B の状態へ移る (遷移)

a

□字句解析プログラム (字句解析器) を作る公式

(1)正規表現→(2) 非決定性有限オートマトン

→(3) 決定性有限オートマトン

(→(4) 状態を最小化した決定性有限オートマトン)

→字句解析プログラム

○正規表現とは? →p15,p16

「辞書」に入れる単語を決める方法の一つ。

□練習 1:  $R=(ab|c)^*c(bc|a)^*$ として、 $L(R)$  (すなわち $[R]$ ) に含まれる語を短いものから 10 個程度 列挙せよ。

○有限オートマトンとは? →p20

記号列  $x$  がオートマトンによって「受理される」とは、その初期状態から  $x$  を読み始め、 $x$  を読み終わったときにちょうど最終状態になるような遷移が一つはある、ということである。

オートマトンの例→p23

□練習 2: 練習 1 の  $R$  に対するオートマトンは、p23 の一番下の図であるが、練習 1 で列挙した単語がきちんと受理されるかどうか確認せよ。

例: 入力="c"の場合:  $i \rightarrow (\epsilon) \rightarrow 4 \rightarrow (c) \rightarrow 5 \rightarrow (\epsilon) \rightarrow f$

○正規表現→非決定性有限オートマトンへの変換→p21,p22

○決定性有限オートマトンとは? →p24

決定性有限オートマトンの例→p27

□練習 3: 練習 2 のオートマトンを DFA に変換したものが、資料図 4.13 であるが、練習 1 で列挙した単語がきちんと受理されるかどうか確認せよ。

○非決定性有限オートマトンから決定性有限オートマトンへの変換

→p25 (考え方)、本日配布の資料参照

→p26 (アルゴリズム)

○状態数を最小化した例→p28

→p29 (アルゴリズム)

□練習 4: 練習 3 の DFA の状態数を最小にしたものが、資料図 4.15 であるが、練習 1 で列挙した単語がきちんと受理されるかどうか確認せよ。