

1. S obzirom na dinamiku izvođenja, jezični procesori se dijele u 3. skupine: kompilatori, interpretatori i dinamički interpretatori.

- kompilatori - prevode naredbe onim redom kojim su napisane u izvornom programu
 - izvođenje cijelog programa započinje tek nakon što završi prevodjenje
- interpretatori - prevode naredbu i odmah je izvrše
 - ne prevode redosjedom kojim su navedene u izvornom programu, već je redosjed prevodjenja naredbi određen redosjedom njihovog izvođenja
- dinamički interpretatori - paralelna izvodi procese prevodjenja i izvođenja
 - procesi su povezani FIFO spremnikom tako da se prevedene naredbe stavljaju u spremnik, a proces izvođenja uzima prevedene naredbe iz spremnika i izvodi ih

2. 1) algoritam traženja sinkronizacijskog znaka

- sintakсни analizator izbacuje redom uniformne znakove do prvog sinkronizacijskog znaka (";", "end", "}" ...)

2) algoritam lokalnih promjena

- sintakсни analizator zamjenjuje, dodaje ili izbacuje uniformne znakove u cilju postizanja prefiksa neanaliziranog djela izvornog programa koji zadovoljava sintakсна pravila

3) dodatne produkcije koje uključuju pogreške

- skup produkcija gramatike se proširuje dodatnim produkcijama koje definiraju pogreške koje nastaju u izvornom programu

4) algoritmi globalnih promjena

- minimiziraju broj promjena u izvornom programu potrebnih da se postigne sintaksno ispravan izvorni program

③ 1) uporabom različitih znakova za višeznačne operatore

- npr. za unarne operatore $+$ i $-$ moguće je koristiti znakove koji su različiti od znakova binarnih operatera $+$ i $-$

2) prenošenjem sintaksnom analizatoru svih mogućih načina grupiranja znakova u leksičke jedinice

- uzima se ovaj način grupiranja koji zadovoljava sintakсна pravila i omogućuje daljnji rad jezičnog procesora

3) ispitivanjem sive doline niza znakova izvornog programa

- uzima u obzir znakove koji su neposredno ispred ili iza leksičke jedinice

4) dinamičnom razmjenom podataka između leksičkog i sintaksnog analizatora

- zasniva se na: - prekidanju rada leksičkog analizatora u slučaju kada dobiveni niz leksičkih jedinica ne zadovoljava leksička pravila

- pozivanju sintaksnog analizatora svaki put kad se dogodi nejednoznačnost u leksičkoj analizi

4.

- 1) Desna strana bilo koje produkcije započinje završnim znakom gramatike
- 2) Desna strana ni jedne produkcije nije prazni niz ϵ
- 3) Ako više produkcija ima nezavršni znak na lijevoj strani, onda desne strane tih produkcija započinju različitim završnim znakovima

5.

a) $yyy++x$

yyy **r5** (prioritet po duljini)

$++x$ **r1** (prioritet po indeksu regularnog izraza)

b) yyx

yy **r4**

x **r6** (desni kontekst)

c) $x!++$

$x!$ ulazi u stanje S , odbacuje **r7** i vraća se na x

x **r6**

$!++$ **r8** izlazi iz S

6. $\langle S \rangle \rightarrow ffg \langle A \rangle \langle B \rangle$
 $\langle A \rangle \rightarrow f \langle C \rangle a e \langle A \rangle \langle B \rangle$
 $\langle A \rangle \rightarrow \epsilon$

$\langle B \rangle \rightarrow g a \langle A \rangle$
 $\langle C \rangle \rightarrow e e$
 $\langle C \rangle \rightarrow g \langle S \rangle$

$PA = \{Q, \Sigma, \Gamma, \delta, q_0, S, \emptyset\}$

$Q = \{q_0\}$

$\Sigma = \{a, e, f, g, \perp\}$

$\Gamma = \{S, A, B, C, a, e, f, g, \nabla\}$

Produkcije oblika: $A \rightarrow b$

1# Zamjeni $\langle B \rangle \langle A \rangle g f$;
 Pomakni;

2# Zamjeni $\langle B \rangle \langle A \rangle e a \langle C \rangle$;
 Pomakni;

3# Zamjeni $\langle A \rangle a$
 Pomakni;

4# Zamjeni e, a ;
 Pomakni;

5# Zamjeni $\langle S \rangle$;
 Pomakni;

6# Zamjeni $\langle S \rangle$;
 Pomakni;

	a	e	f	g	\perp
S	-	-	#1	-	-
A	-	-	#2	#6	-
B	-	-	-	#3	-
C	-	#4	-	#5	-
a	#7	-	-	-	-
e	-	#7	-	-	-
f	-	-	#7	-	-
g	-	-	-	#7	-
∇	-	-	-	-	#8

Produkcije oblika: $A \rightarrow \epsilon$

PRIMJENI ($A \rightarrow \epsilon$) = SLIJEDI (A) = $\{g\}$

#6 Izvuci;
 Zadrzi;

Definiranje akcije za završne
 znakove na vrhu stoga:

#7 Izvuci;
 Pomakni;

Dodavanje akcije za prihvatanje niza:

#8 Prihvati

Dodavanje akcije za odbijanje niza:

- Odbaci

(moguće još produkcije oblika $A \rightarrow b$, tad je akcija Izvuci;
 Pomakni;)

7. PREUŠE RASPISIVANJA ZA 1 BOD, NEDA MI SE :)

rišenje: www.fer2net/showpost.php?p=679460&postcount=18

8 $;$ $3+2 \rightarrow a$; $a^12 \rightarrow b$; $b+a*3 \rightarrow c$; $6 \rightarrow a$;

PUSH 3

$a=?$ $b=?$ $c=?$

3

... $1+2 \rightarrow$...

PUSH 2

ADD

$a=?$ $b=?$ $c=?$

3
2

5

... $\rightarrow a$; ... $a=5$, $b=?$, $c=?$

POP a

... $;$ a^1 ... $a=5$, $b=?$, $c=?$

PUSH a

5

... $^12 \rightarrow$...

PUSH 2

EXP

$a=5$, $b=?$, $c=?$

5
2

25

... $\rightarrow b$; ...

POP b

$a=5$, $b=25$, $c=?$

... $;$ $b+$...

PUSH b

$a=5$, $b=25$, $c=?$

25

... $+$ $a*$...

PUSH a

ADD

$a=5$, $b=25$, $c=?$

25
5

30

... $*$ $3 \rightarrow$...

PUSH 3

MUL

$a=5$, $b=25$, $c=?$

30
3

90

... $\rightarrow c$; ... $a=5$, $b=25$, $c=90$

POP c

... $;$ $6 \rightarrow$... $a=5$, $b=25$, $c=90$

PUSH 6

6

... $\rightarrow a$;

POP a

$a=6$, $b=25$, $c=90$

9.

1) $JP_a^{P \rightarrow a}$

2) $JP_b^{S \rightarrow P}$

3) $JP_s^{Q \rightarrow c}$

$JP_x^{x \rightarrow a} \quad x = ?$

$$JP_s^{Q \rightarrow c} \Rightarrow \boxed{JP_b^{S \rightarrow P}} \Rightarrow JP_p^{Q \rightarrow c} \Rightarrow \boxed{JP_a^{P \rightarrow a}} \Rightarrow JP_a^{Q \rightarrow c}$$

(P) $JP_p^{P \rightarrow a} \Rightarrow \boxed{JP_a^{P \rightarrow a}} \Rightarrow JP_a^{P \rightarrow a}$

(P_s) $\Rightarrow \boxed{JP_b^{S \rightarrow P}} \Rightarrow P_p \Rightarrow \boxed{JP_a^{P \rightarrow a}} \Rightarrow P_a \Rightarrow \boxed{JP_a^{Q \rightarrow c}} \Rightarrow P_c$

10.

redni broj	KROS
1	def
2	(
3	,
4)
5	:
6	=
7	lambda
8	.
9	for
10	in
11	return

redni broj	IDN
1	format_cond
2	string_list
3	c
4	format_func
5	s
6	join
7	split
8	rez
9	append

redni broj	KON
1	" "

izvorni program	unifarni znak	bazalika
def	KROS	1
format_cond	IDN	1
(KROS	2
string_list	IDN	2
,	KROS	3
c	IDN	3
)	KROS	4
:	KROS	5
format_func	IDN	4
=	KROS	6
lambda	KROS	7
s	IDN	5
:	KROS	5
" "	KON	1
join	KROS	8
(IDN	6
s	KROS	2
)	IDN	5
split	KROS	8
(IDN	7
)	KROS	2
)	KROS	4
for	KROS	4
s	KROS	5
in	IDN	5
string_list	KROS	10
,	IDN	2
rez	KROS	5
.	IDN	8
append	KROS	8
(IDN	9
format_func	KROS	2
(IDN	4
s	KROS	2
)	IDN	5
)	KROS	4
)	KROS	4
return	KROS	11
rez	IDN	8