

```

}
warna {
  nikalo "Still a kid bro!";
}

```

2. Keywords:

Keyword	Meaning	Example
nikalo	output	nikalo "Yo bro!";
dalo	input	dalo
agar	if	agar (x > 0)
warna	else	warna { nikalo "Nope"; }
jabtak	while	jabtak (x > 0) { }
karlo	do	karlo { nikalo "x"; } jabtak (a > 0)
forbro	for	forbro (i = 0; i < 5; i++)
wapas	return	wapas result;
textbro	string	textbro name = "Lavaib";
boolbro	bool datatype	boolbro done = sach;
sach	true	//
jhoot	false	boolbro done = jhoot;
ginti	int	ginti x = 10;

pointbro
rukja

float
brak

pointbro pi = 3.14;
rukja;

3. Operators and Punctuations

Symbol	Operator	Example
>+	Addition	a>+b
<-	Subtract	a<-b
**	Multiplication	a**b
//	Divison	a//b
:/bro	Modulus	a:/bro b
+bro	Increment	a+bro
-bro	Decrement	a-bro
=	Assignment	a=10
>>	Greater than	x>>10
<<	Less than	x<<10
?=	Equal to	agav(x?=10)
&bro	AND	agav(x?=10 &bro y?=20)
!	NOT	agav(x!=10)

Punctuation:

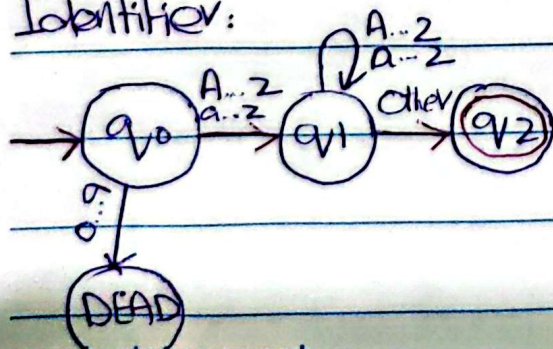
Symbol	Descp	Example
;	Statement Termin	nikalo "Yo bro";
{ }	Code Block	{ nikalo "Done"; }
()	Condition or Parameter	agar (x? = 10)
" "	String Literal	nikalo "Bro!";

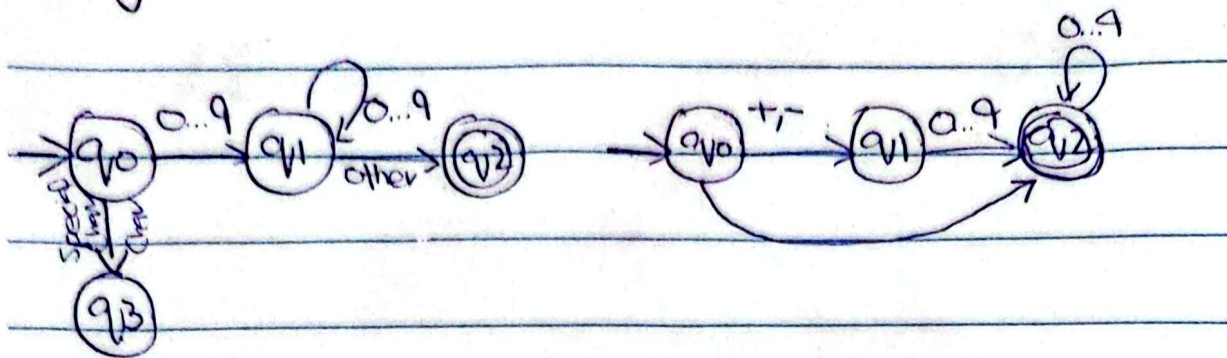
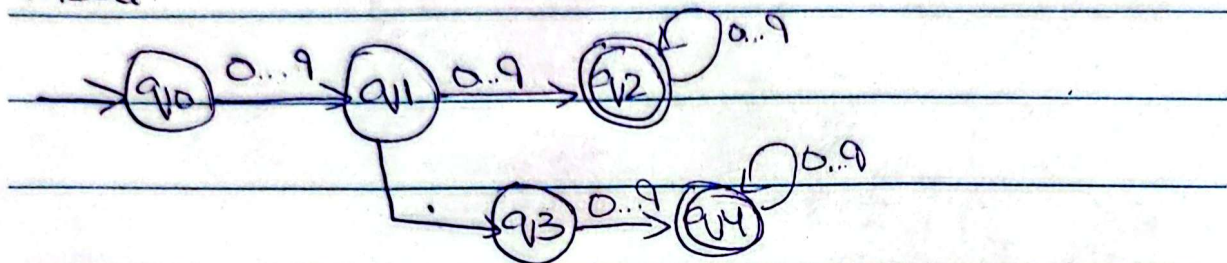
4. Regular Expression:

Token Type	Re	Example
Identifier	$[a \dots z, A \dots Z]^+ [a \dots z, A \dots Z]^*$	ginti score = 10;
Integer	$[+/-]? [0 \dots 9]^+$	123
Float	$[0 \dots 9]^+ \cdot [0 \dots 9]^+$	3.14
Operator	$[>, <, =, //, **, \%, \dots]$	<-

5. Finite Automata

Identifier:



Integer:Float:Explanation:

Bro++ was designed to make programming more fun, relaxed for new learners. It replaces strict keywords with strings.

- Instead of if, it uses agar
- Instead of cin, cout, it uses nikalo, dalo
- Operators are <-, >+

This unique approach makes language creative and fun.