

# FreeBASIC Syntax Summary

**Author:** Jan Jansen aka John Doe



# Contents

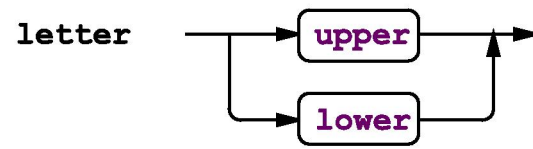
<b>1</b>	<b>Lexical Grammar</b>	<b>5</b>
1.1	Tokens	5
1.1.1	letter	5
1.1.2	upper	6
1.1.3	lower	8
1.1.4	underscore	10
1.1.5	sign	10
1.1.6	digit	10
1.1.7	identifier	11
1.1.8	constant	11
1.1.9	integer_constant	11
1.1.10	integer_suffix	12
1.1.11	unsigned_suffix	12
1.1.12	long_suffix	12
1.1.13	long_long_suffix	12
1.1.14	decimal_constant	12
1.1.15	octal_constant	12
1.1.16	octal_digit	13
1.1.17	hexadecimal_constant	13
1.1.18	hexadecimal	13
1.1.19	hexadecimal_char	14
1.1.20	binary_constant	14
1.1.21	binary_digit	14
1.1.22	floating_constant	15
1.1.23	fractional_constant	15
1.1.24	exponent_part	15
1.1.25	floating_suffix	15
1.1.26	single_suffix	15
1.1.27	double_suffix	16
1.1.28	string_constant	16
1.1.29	char	16
1.1.30	escape_sequence	16
1.1.31	simple_escape_sequence	17
1.1.32	BACKSLASH	17
1.1.33	ascii_decimal_escape_sequence	17
1.1.34	ascii_binary_escape_sequence	17

1.1.35	binary_sequence . . . . .	18
1.1.36	ascii_octal_escape_sequence . . . . .	18
1.1.37	octal_digit_sequence . . . . .	18
1.1.38	ascii_hexadecimal_escape_sequence . . . . .	18
1.1.39	unicode_escape_sequence . . . . .	18
1.1.40	hexadecimal_sequence . . . . .	18
1.2	Keywords . . . . .	19
1.3	Punctuators . . . . .	20

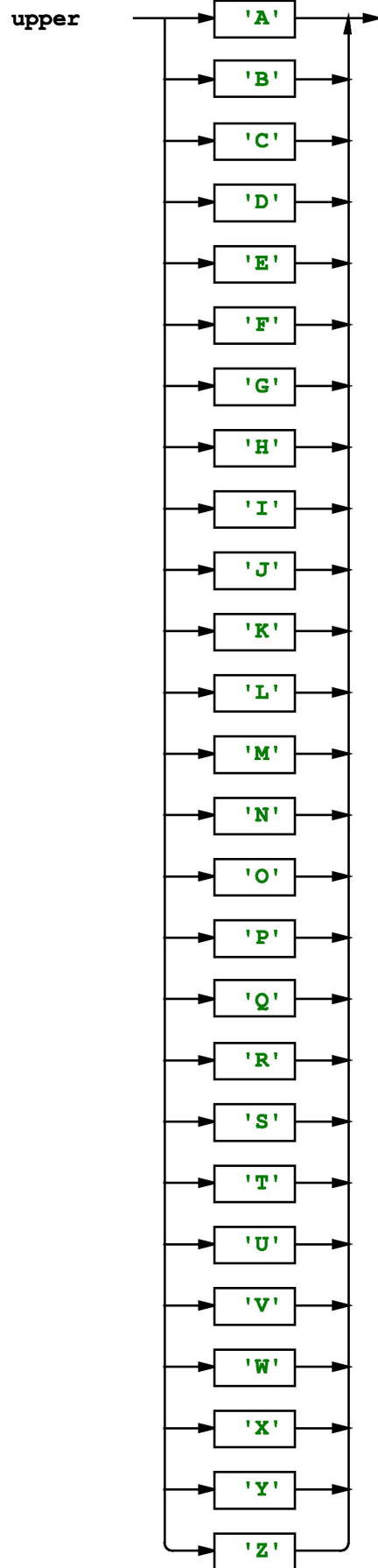
# 1 Lexical Grammar

## 1.1 Tokens

### 1.1.1 *letter*

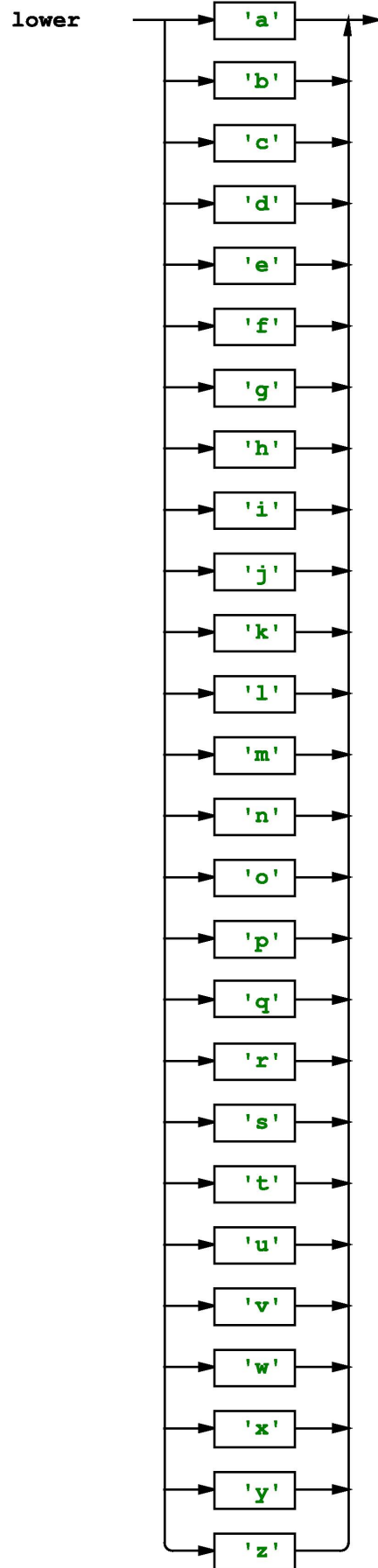


### **1.1.2 upper**

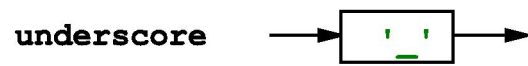


### **1.1.3 lower**

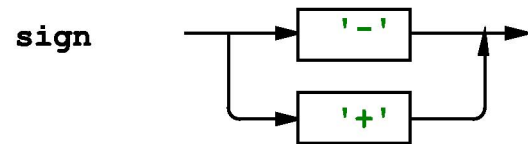




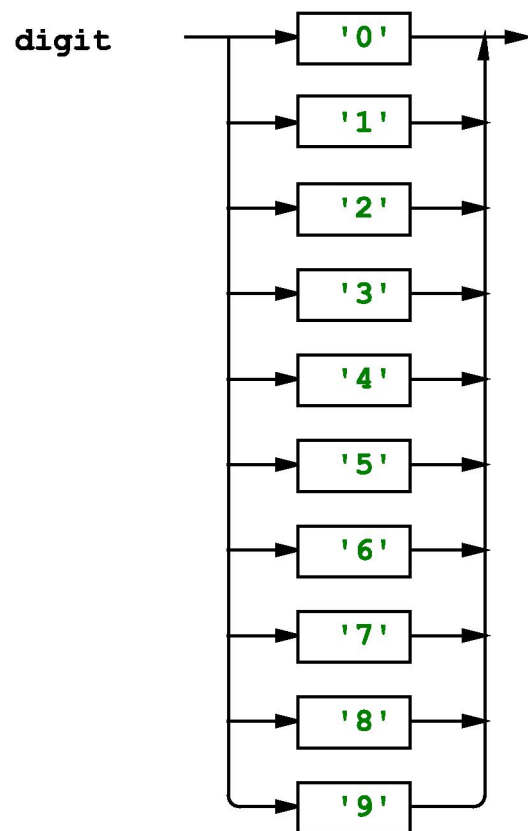
#### 1.1.4 underscore



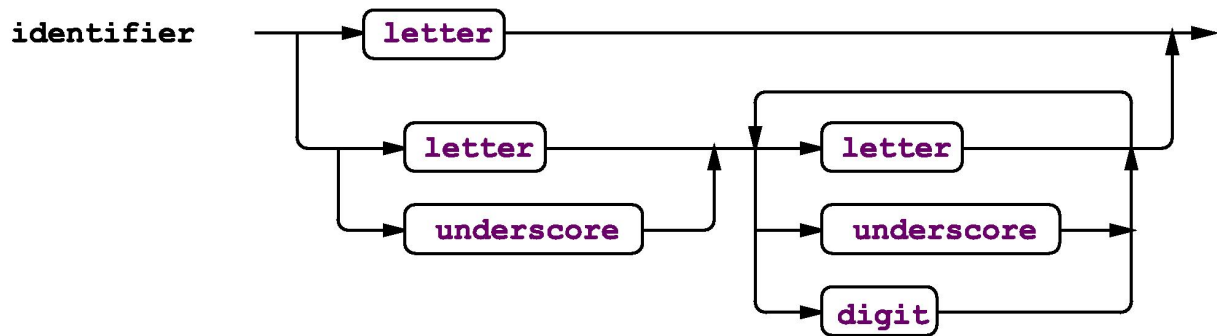
#### 1.1.5 sign



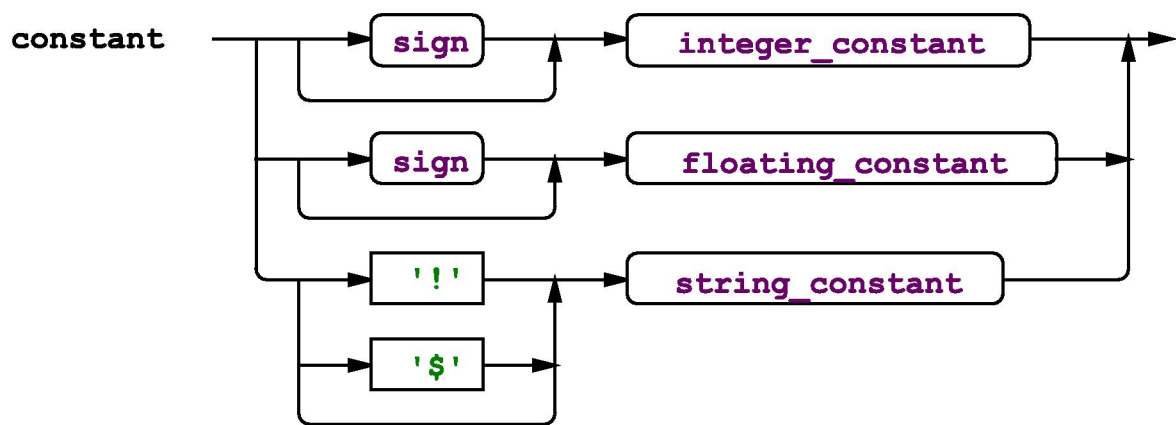
#### 1.1.6 digit



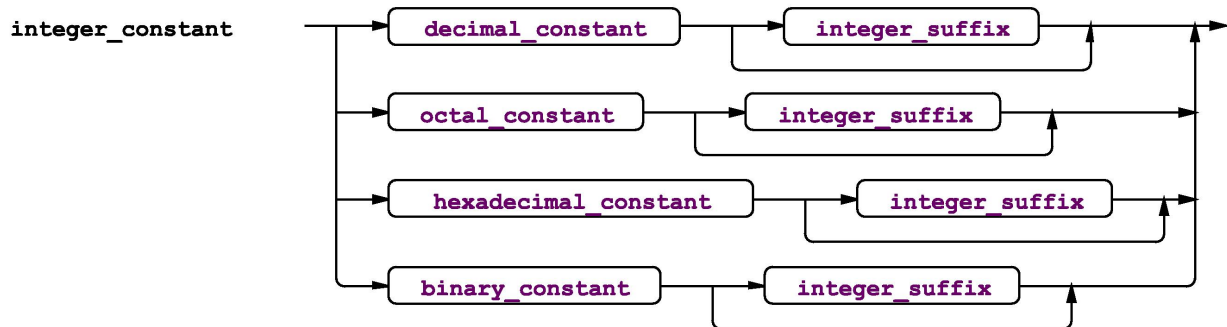
### 1.1.7 identifier



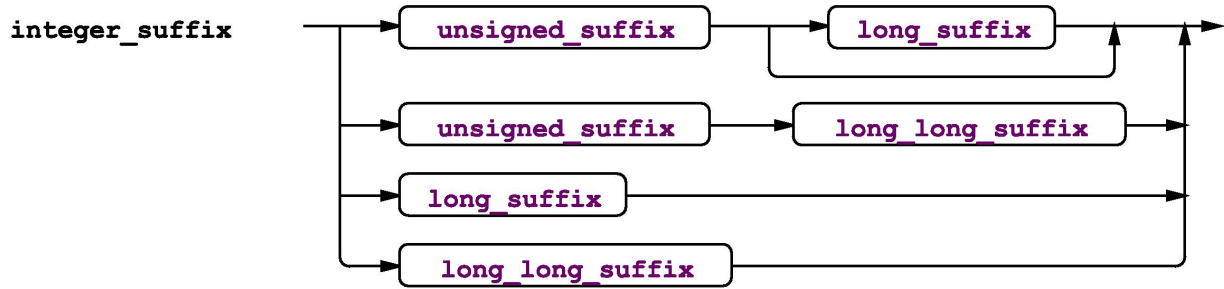
### 1.1.8 constant



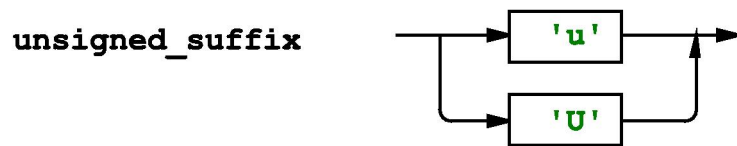
### 1.1.9 integer\_constant



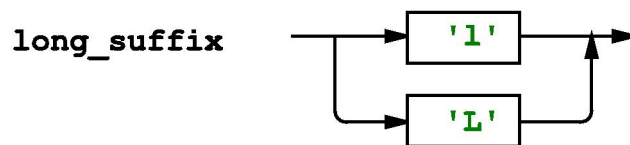
### 1.1.10 integer\_suffix



### 1.1.11 unsigned\_suffix



### 1.1.12 long\_suffix



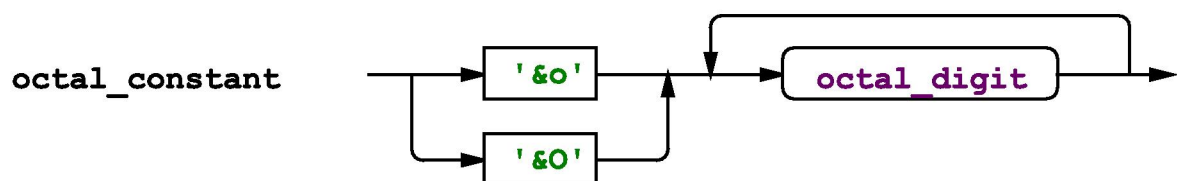
### 1.1.13 long\_long\_suffix



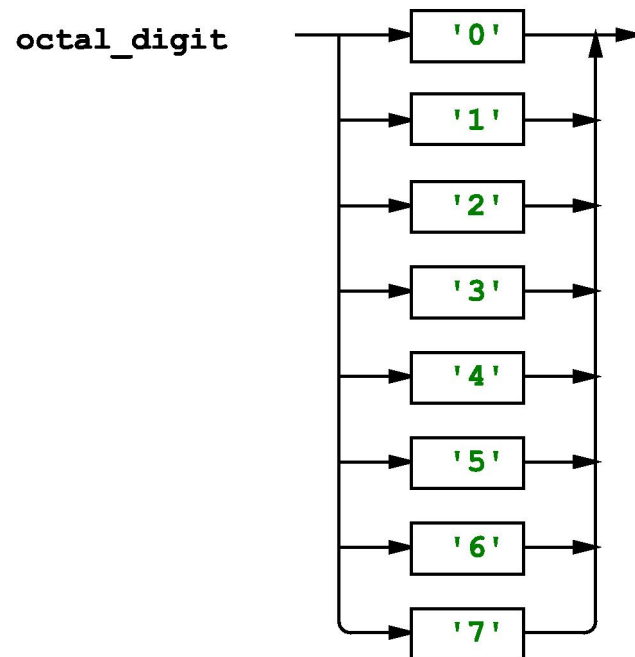
### 1.1.14 decimal\_constant



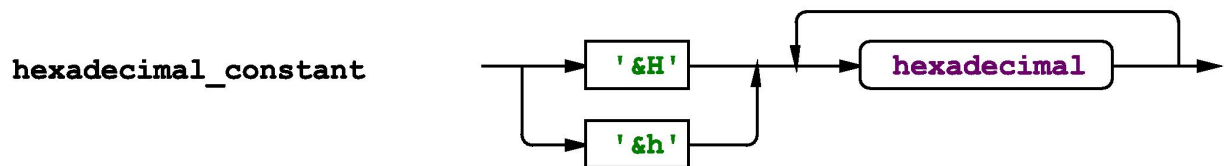
### 1.1.15 octal\_constant



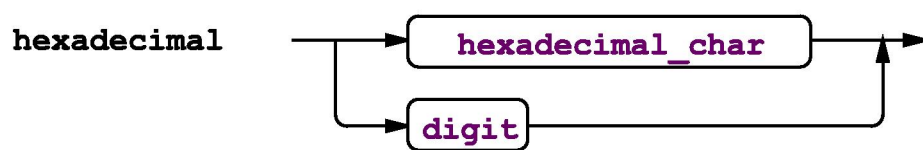
### 1.1.16 *octal\_digit*



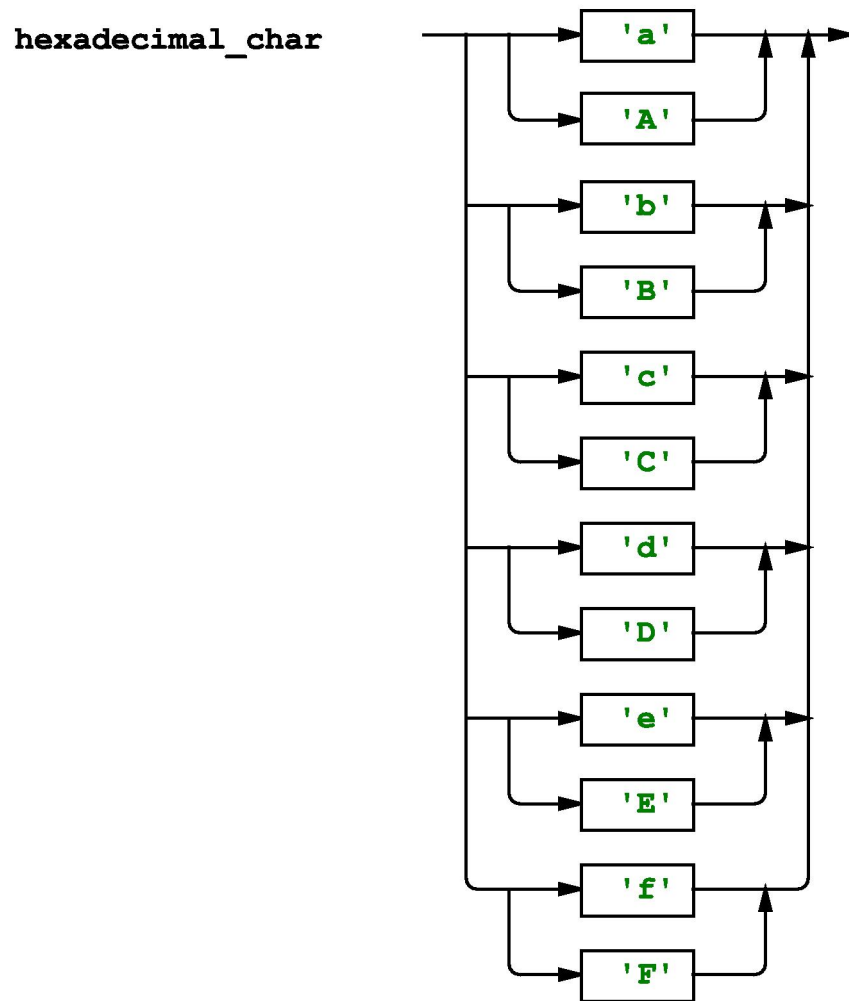
### 1.1.17 *hexadecimal\_constant*



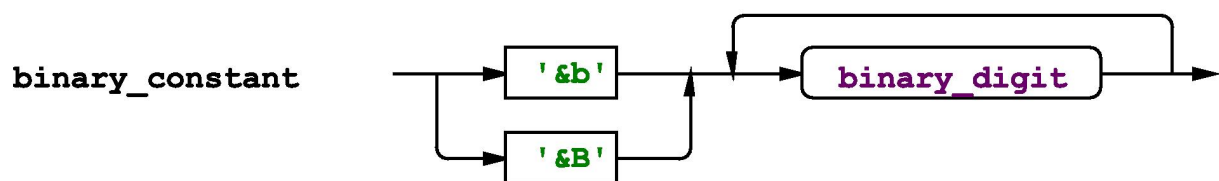
### 1.1.18 *hexadecimal*



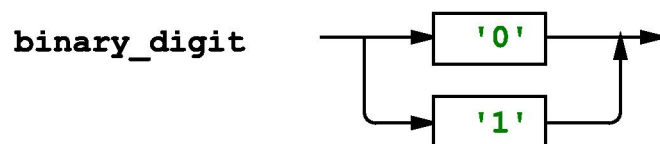
### 1.1.19 hexadecimal\_char



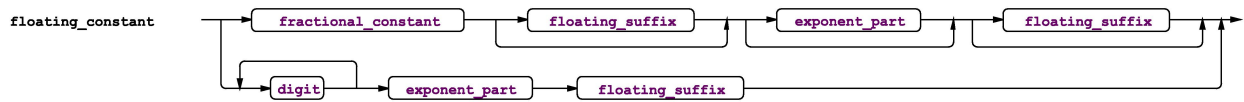
### 1.1.20 binary\_constant



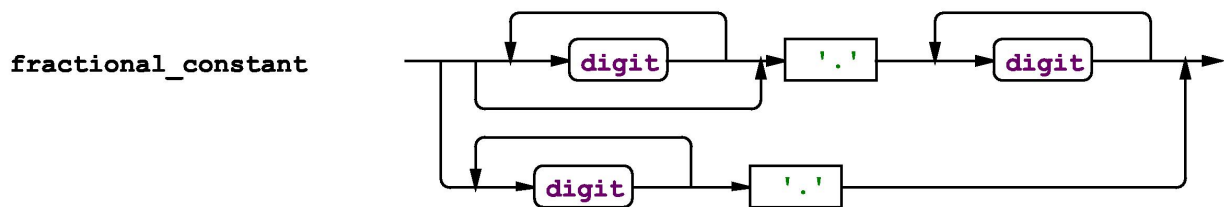
### 1.1.21 binary\_digit



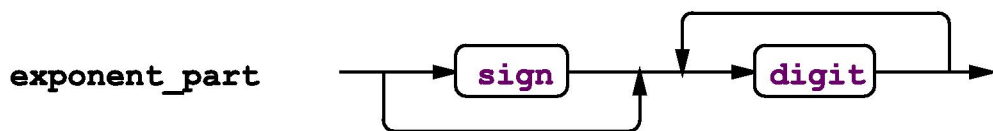
### 1.1.22 floating\_constant



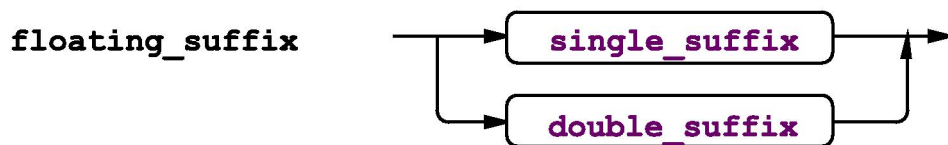
### 1.1.23 fractional\_constant



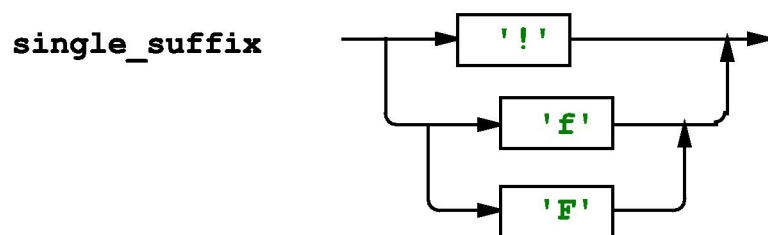
### 1.1.24 exponent\_part



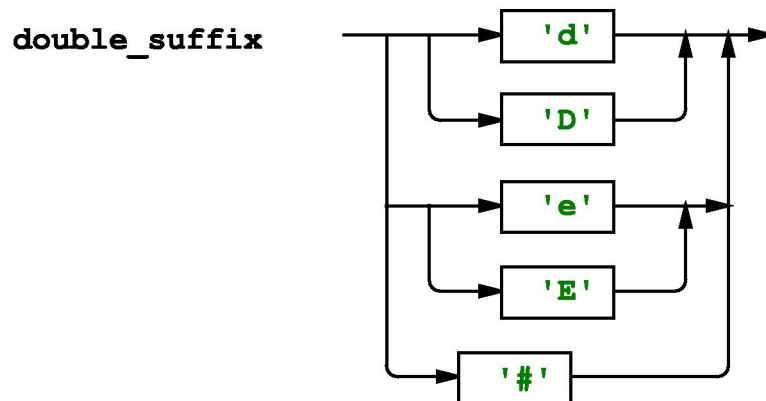
### 1.1.25 floating\_suffix



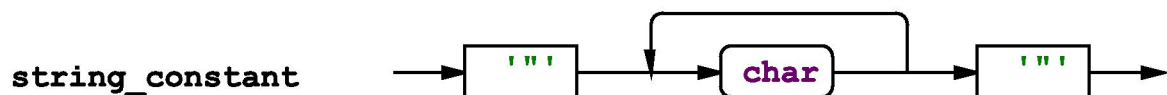
### 1.1.26 single\_suffix



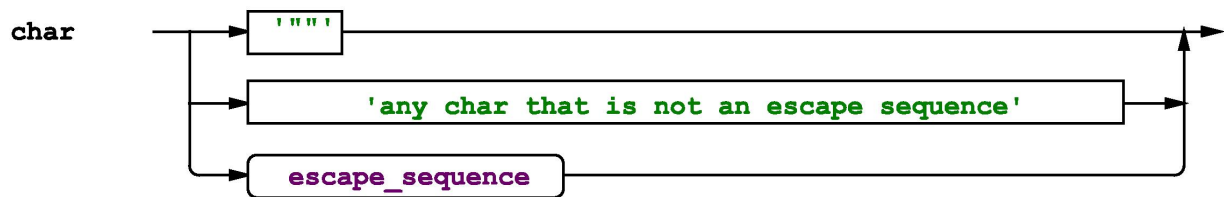
### 1.1.27 double\_suffix



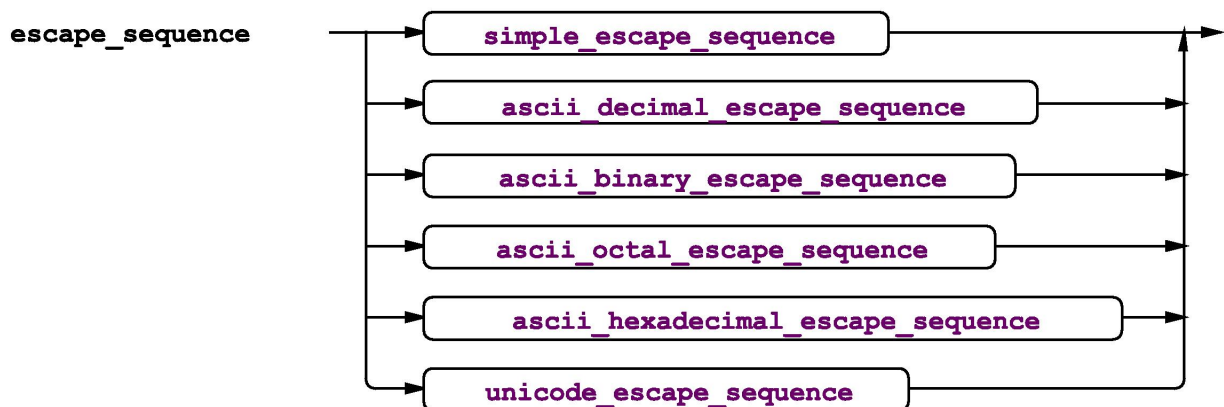
### 1.1.28 string\_constant



### 1.1.29 char



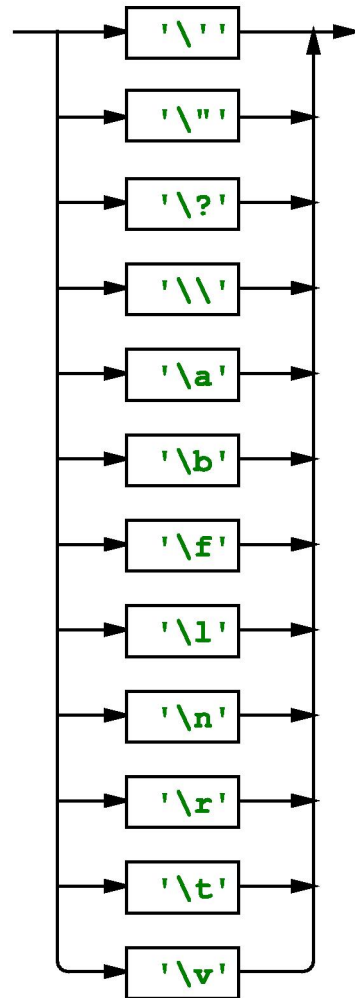
### 1.1.30 escape\_sequence





### 1.1.31 *simple\_escape\_sequence*

**simple\_escape\_sequence**

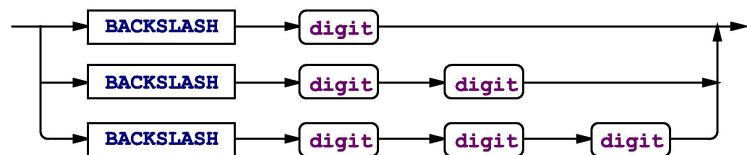


### 1.1.32 *BACKSLASH*

**BACKSLASH** →

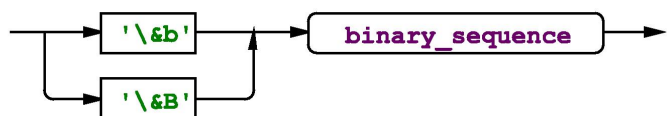
### 1.1.33 *ascii\_decimal\_escape\_sequence*

**ascii\_decimal\_escape\_sequence**

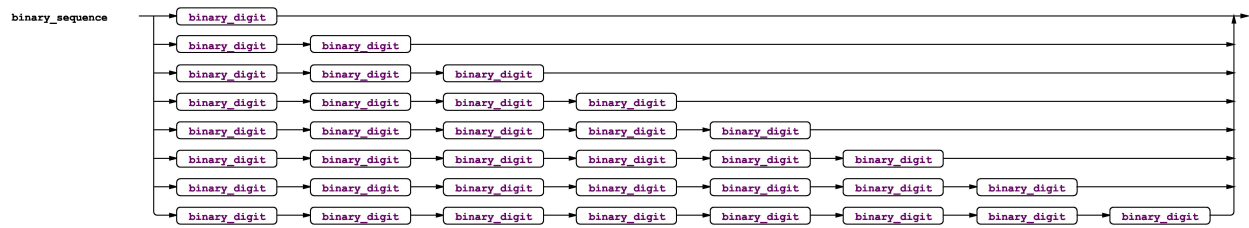


### 1.1.34 *ascii\_binary\_escape\_sequence*

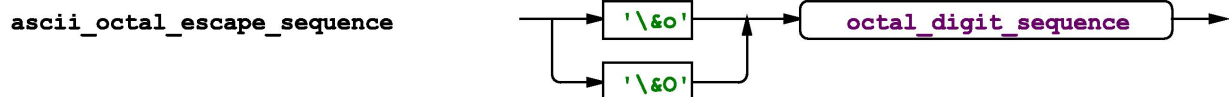
**ascii\_binary\_escape\_sequence**



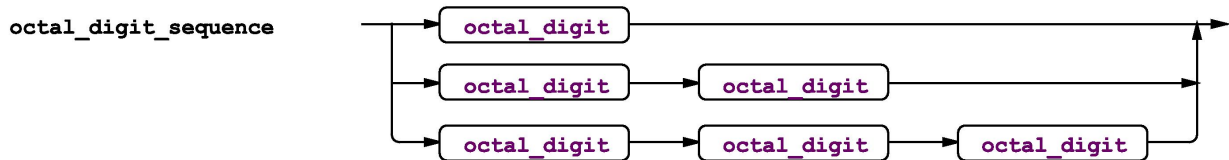
### 1.1.35 *binary\_sequence*



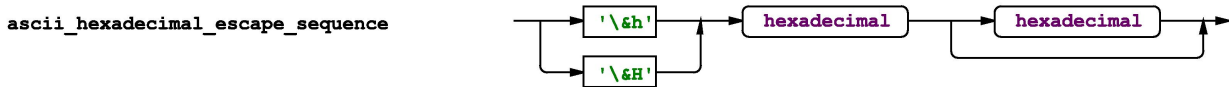
### 1.1.36 *ascii\_octal\_escape\_sequence*



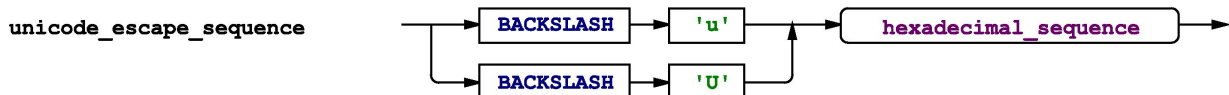
### 1.1.37 *octal\_digit\_sequence*



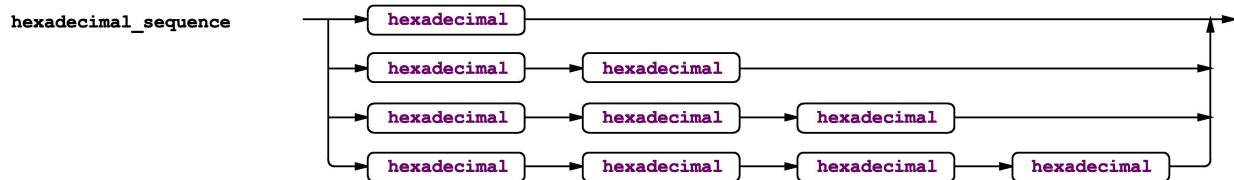
### 1.1.38 *ascii\_hexadecimal\_escape\_sequence*



### 1.1.39 *unicode\_escape\_sequence*



### 1.1.40 *hexadecimal\_sequence*



## 1.2 Keywords

abs	circle	defined	explicit	local	overload	scope	ubyte
access	class	defint	export	lock	paint	screen	uinteger
acos	clng	deflng	extern	log	palette	screenres	ulong
alias	clngint	deflongint	field	long	pascal	seek	ulongint
and	close	defshort	fix	longint	peek	select	union
andalso	color	defsng	for	loop	point	sgn	unlock
any	common	defstr	frac	loword	pointer	shared	unsigned
append	const	defubyte	function	lprint	poke	shl	until
as	constructor	defuint	get	lset	preserve	short	ushort
asc	continue	defulng	gosub	ltrim	preset	shr	using
asin	cos	defulongint	goto	mid	print	sin	va_arg
asm	cptr	defushort	hibyte	mkd	private	single	va_first
assert	cshort	delete	hiword	mki	procptr	sizeof	va_next
assertwarn	csign	destructor	if	mkl	property	spc	var
atan2	csng	dim	iif	mklongint	protected	sqr	varptr
atn	cubyte	do	imagecreate	mks	pset	static	view
binary	cuint	double	imp	mkshort	ptr	stdcall	wchr
bit	culng	draw	import	mod	public	step	wend
bitreset	culngint	dynamic	include	name	put	str	while
bitset	cunsg	else	input	namespace	random	string	width
byref	cushort	elseif	instr	new	read	strptr	window
byte	cvd	encoding	instrrev	next	redim	sub	with
byval	cvi	end	int	not	rem	swap	write
call	cvl	endif	integer	offsetof	restore	tab	wstr
case	cvlongint	enum	is	on	resume	tan	wstring
cast	cvs	eqv	lbound	open	return	then	xor
cbyte	cvshort	erase	len	operator	rgb	to	zstring
cdbl	data	err	let	option	rgba	trim	
cdecl	declare	error	lib	or	rset	type	
chr	defbyte	exit	line	orelse	rtrim	typeof	
cint	defdbl	exp	lobyte	output	sadd	ubound	

## 1.3 Punctuators

[	#	^	<	;	\=
]	##	+	>	...	+=
(	.	-	<=	=	-=
)	,	!	>=	*=	
{	->	/	<>	^=	
}	*	\	:	/=	