"Grammar"

S → AS ①| BS ②| CS ③| DS ④| ES ⑤| ε ⑥

A → int identifier F ⑦

F → ⑧= 0 ; | ; ⑨

B → float identifier F ⑩

C → cout << String ; ⑪

D → cin >> identifier ; ⑫

E → if (G) M N ⑬

P → ⑭if (G) Q | Q ⑮

M → ⑯R | Q ⑰

N → ⑱else P N | ε ⑲

G → ⑳Constant T O | identifier T O ㉑

R → { S } ㉒

Q → ㉓A | ㉔B | ㉕C | ㉖D

T → ㉗== | ㉘< | ㉙>℗ | ㉚<= | ㉛>= | ㉜!=

O → ㉝identifier | Constant ㉞

	First	Follow	
S	{ int, float, cout, cin, if, ε }	{ $, } }	
A	{ int }	{ int, float, cout, cin, if, $, }, else }	✓ P
F	{ =, ; }	{ int, float, cout, cin, if, $, }, else }	✓ B
B	{ float }	{ int, float, cout, cin, if, $, }, else }	✓ Q
C	{ cout }	{ int, float, cout, cin, if, $, }, else }	✓ Q
D	{ cin }	{ int, float, cout, cin, if, $, }, else }	&
E	{ if }	{ int, float, cout, cin, if, $, } }	
P	{ if, int, float, cout, cin }	{ else, int, float, cout, cin, if, $, } }	
M	{ {, int, float, cout, cin }	{ else, int, float, cout, cin, if, $, } }	
N	{ else, ε }	{ int, float, cout, cin, if, $, } }	
G	{ Constant, identifier }	{) }	
R	{ { }	{ else, int, float, cout, cin, if, $, } }	
Q	{ int, float, cout, cin }	{ else, int, float, cout, cin, if, $, } }	
T	{ ==, <, >, <=, >=, != }	{ identifier, Constant	
O	{ identifier, Constant }	{ ;) }	

Parsing Table

	$	int	identifier	=	;	float	cout	<<	String	cin	>>	if	(	)	else	{	}	==	<	>	<=	>=	!=	Constant
S	6	1			2		3			4		5					6							
A		7																						
F				8	9																			
B					10																			
C								11																
D										12														
E												13												
P		15				15	15			15		14												
M		17				17	17			17							16							
N	19	19				19	19			19		19				18		19						
G			21																					20
R														22										
Q		23				24	25			26														
T																		27	28	29	30	31	32	
O			33																					34