[Context-Free Grammar]

S ---> D

S ---> P

P ---> cout << Q;

P ---> cin >> V;

P ---> C S

Q ---> String

D ---> Y I;

Y ---> int

Y ---> float

I ---> V G

G ---> = X

G ---> Epsilon

C ---> if (X) {S} E

E ---> elseif (X) {S} E

E ---> else {S}

E ---> Epsilon

X ---> T O T

T ---> N

T ---> V

O ---> A

O ---> F

A ---> +

A ---> -

A ---> *

A ---> /

A ---> %

A ---> =

F ---> >

F ---> <

F ---> ==

F ---> !=

N ---> number

V ---> variable R

V ---> _ R

R ---> V

R ---> number R

R ---> Epsilon

[First()]

S = float, int, cout, cin, if

P = cout, cin

Q = String

D = int, float

Y = int, float

I = variable, _

G = =. Epsilon

C = if

E = else if, else, Epsilon

X = number, variable

T = number, variable

O = +, -, *, /, %, =, <, ==, !=

A = +, -, *, /, %, =

F = <, ==, !=

N = number

V = variable, _

R = variable, _, number, Epsilon

[Follow()]

S = $, {
P = $, {
Q = ;
D = $, {
Y = variable, _
I = ;
G = }, $
C = float, int, cout, cin, if
E = float, int, cout, cin
X = $,), {
T = +, -, *, /, %, =, <, ==, !=, $,), {
O = number, variable, _
A = number, variable, _
F = number, variable, _
N = +, -, *, /, %, =, <, ==, !=, $,), {
V = +, -, *, /, %, =, <, ==, !=, $,), {
R = +, -, *, /, %, =, <, ==, !=, $,), {

[Parsing Table]

	;	cout	<<	cin	>>	"	int	float	if	(	)	{	}	else if	else	epsilon	+	-	*	/	%	=	>	<	==	!=	number	variable	_	$
S		2		2			1	1	2																					
Q		3		4					5																					
D						6																								
Y							7	7																						
I							8	9																			10	10		
G													12																	12
C									13																					
E		16		16			16	16						14	15															
X																											17	17	17	
T																											18	19	19	
O																	20	20	20	20	20	21	21	21	21	21				
A																	22	23	24	25	26	27								
F																							28	29	30	31				
N																														
V																														
R	37										37	37					37	37	37	37	37	37	37	37	37	37	36	35	35	37