																								Ар	куш1																			
2000	<app><app><appname></appname></app></app>	<pre><li></li></pre> <pre></pre> <p< th=""><th><pre><definition></definition></pre></th><th><definition2></definition2></th><th><pre><li><li><li>dist of operators2&gt;</li></li></li></pre></th><th>List of operators of coperators of copera</th><th><pre><operator2> <setter></setter></operator2></pre></th><th><input/></th><th><output></output></th><th><condition></condition></th><th><lostdoord <="" li=""></lostdoord></th><th><losteron <="" li=""></losteron></th><th><relation></relation></th><th><expression3></expression3></th><th><expression></expression></th><th><term2></term2></th><th><pre><multiplier> <multiplier2></multiplier2></multiplier></pre></th><th><expr.response></expr.response></th><th><expr.response2></expr.response2></th><th>CONST</th><th>@implementation @interface</th><th>@end</th><th>input</th><th>output</th><th>for</th><th>to</th><th>next</th><th><u></u></th><th>else endif</th><th>endset }</th><th>} (</th><th>( )</th><th>[ ]</th><th>= eq</th><th>u !=</th><th>&gt; -</th><th>&lt; &gt;=</th><th>: &lt;= _</th><th>. + -</th><th>/</th><th>* ^ ,</th><th>and</th><th>or ;</th><th># &gt;</th></p<>	<pre><definition></definition></pre>	<definition2></definition2>	<pre><li><li><li>dist of operators2&gt;</li></li></li></pre>	List of operators of coperators of copera	<pre><operator2> <setter></setter></operator2></pre>	<input/>	<output></output>	<condition></condition>	<lostdoord <="" li=""></lostdoord>	<losteron <="" li=""></losteron>	<relation></relation>	<expression3></expression3>	<expression></expression>	<term2></term2>	<pre><multiplier> <multiplier2></multiplier2></multiplier></pre>	<expr.response></expr.response>	<expr.response2></expr.response2>	CONST	@implementation @interface	@end	input	output	for	to	next	<u></u>	else endif	endset }	} (	( )	[ ]	= eq	u !=	> -	< >=	: <= _	. + -	/	* ^ ,	and	or ;	# >
app> appName>																																												>
list of definitions>																						+		>																				>
list of definitions2>																								=																				>
definition>																						4		>																				>
definition2> list of var>																						+		>								=												>
list of operators2>																						+++		=																				>
list of operators>																								>																				>
coperator>																								>																				>
coperator2>																								>																				>
setter> input>																						+		>																				
output>																						+		>																				>
cycle>																								>																				>
condition>											H											$\bot$		>				L.T.							$\perp$								$\bot$	>
clogical expression>				-										-	+-							++		>					-				>		-								=	- 2
<pre>clogical expression2&gt; clog.exp.lev1&gt;</pre>					+	-				_		-		-	++		-			-		++		=				++	+				= >		-			++		-			>	>
log.exp.lev2>																						+++		>									>		+							>		
relation>																								>									>									_	>	>
expression3>																								=																				
expression2>																		_				+		>		=				=		=	>	>	-		> >						>	>
expression>																						+		>		> >				>		>	>	>	>	+ +	> >		= :	_	= >		>	;
sterm2>																						++		>		>	_			>		>	>	>	+-	_	_	>	> :	_	=	_	>	
multiplier>																								>		>	>			>		>	>	>	>	>	> >	>	> :	> >	> =	>	>	:
multiplier2>																								>		>	>			>		>	>	>	>	>	> >	>	> :	> >	> >	>	>	
expr.response>																								>		>	_			>		>	>	>	>	>	> >	>	> :	> >	> >	>	>	
expr.response2>																								>		>	_			>		>	>	>	+-	_	> >			_	> >	>	>	>
D CONST				=														-				+++		>		= >			-	>		>	>	= >	_	_	> >		> :	_	> >	< >		= >
Dimplementation																		+				+++		=											Ť									>
Dinterface Dend	=																			<																								>
nt nput				=																												=										<		>
output																																=												>
ENTER		< :	= <	=	=   •	< <	= <	< <	< <	<										< =	=	= <	< <	< <	<			<	=   =															>
or rom															< <	: <	< <	: <		_		+++										<												>
0															= <			_														<												>
step														= <	< <	: <	< <	: <	< .	< <												<												>
next																								>																				>
										<	=	< <	<	<	< <	<	< <	<	< •	< <												<	<						<					>
else					=   •	<																+		>																				;
endif endset												_			++					+		++		>					+						-									
																																												:
				=	+									=	< <	<	< <	<	< -	< <		+-						+	_	1.		<			-							<	1	?
						+				-				-			_   .			_   _		++		>		>	>	+++	+	>		>	>	>	>	>	> >		> :	>   >	>   >	>	>	- 2
						-				<	=	` <	<	<	< <	. <	`   <	\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	< ·	`		++		>					+			<	< >					+ + •	<	-		>	>	>
<u> </u>					+									=	< <	: <	< <	: <	< .	< <		+										<										-	+	
qu															< <			_		_												<												>
=														_	< <			_	_			$\bot$										<												>
				_		_								_	< <	-		_		_		+							_			<			-									
=						-				-				_	< <			_		_		++							+			< <								-				;
<del>=</del>  =					+										< <			_				++							+			<						+						
												=	<	_	< <			_		_												<	<						<					>
															<	=	< <	: <	< .	< <												<												:
											H				<	=		_		_		$\bot$										<				$\Box$								
										-				-				_	< .	_		++							+			<								-				
																			= -	_												<												:
and .																				=		1							T															:
and or					+							= <	<		< <			_		_		++						+++	+	+++		< <	<		-				<		++		++	>
																								>																				:
	e e		< <	< <	< .	< <	< <	< <	< <	< <	<	< <	<	< <	< <	: <	< <	: <	< .	< <	1 1	e .	/ /			< <	< <	<	< <	<	< <	< <	< <	< <	<	<	1 1	< .	< < .	< <	< <	< <	<	<