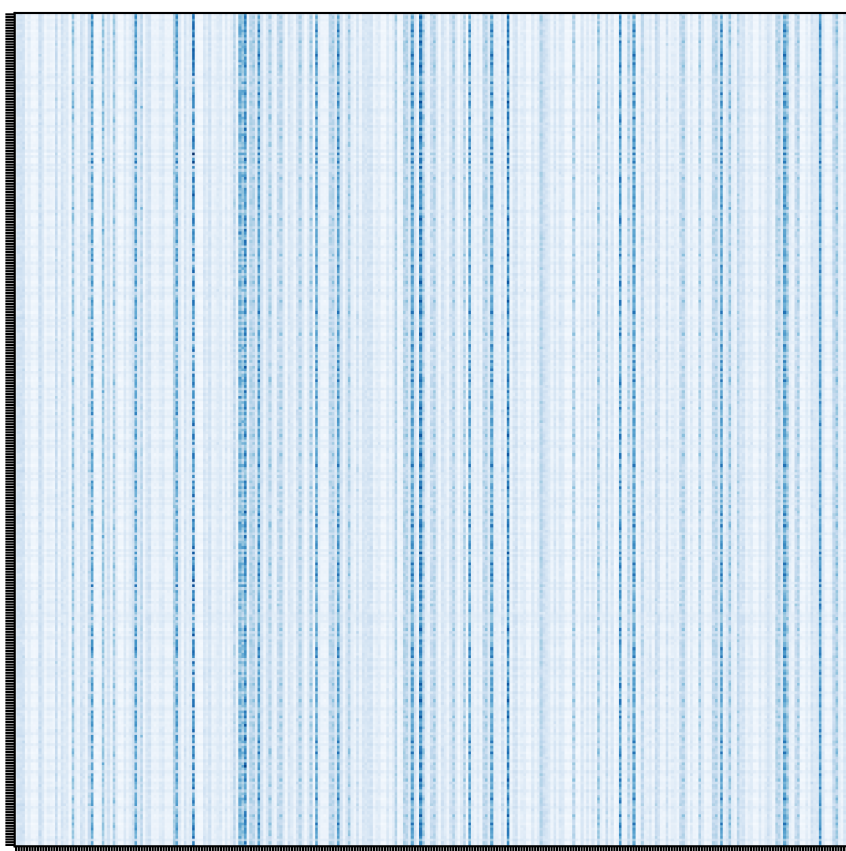


```
(!allowMixedLanguage)
!_headContext.isToken()
null; } } exposed to
ctxt.next token is
delegate.get current token (head)
_headContext.findChildOf(ctxt).exposed to
_constructError("Unexpected
delegate.next token
TokenFilter.ID.START
_headContext.createChildOf(ctxt)
{delegate.skipChildren(break
f.filterTokenMayInclude
_headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
_nextTokenWithBuffering( headContext
TokenFilter.ID.STOP
_headContext.createChildOf(ctxt)
{delegate.skipChildren(break
f.filterTokenMayInclude
_headContext.createChildOf(ctxt)
_nextTokenWithBuffering( headContext
headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
headContext.createChildOf(ctxt)
{delegate.next token(); delegate.skipChildren(break
{delegate.next token(); delegate.skipChildren(break
tokenFilter.Include
_nextTokenWithBuffering( headContext
TokenFilter.Include
_headContext.createChildOf(ctxt)
f.includeValue(delegate
e); } next token is
```



```
(!allowMixedLanguage)
!_headContext.isToken()
null; } } exposed to
ctxt.next token is
delegate.get current token (head)
_headContext.findChildOf(ctxt).exposed to
_constructError("Unexpected
delegate.next token
TokenFilter.ID.START
_headContext.createChildOf(ctxt)
{delegate.skipChildren(break
f.filterTokenMayInclude
_headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
_nextTokenWithBuffering( headContext
TokenFilter.ID.STOP
_headContext.createChildOf(ctxt)
{delegate.skipChildren(break
f.filterTokenMayInclude
_headContext.createChildOf(ctxt)
_nextTokenWithBuffering( headContext
headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
_headContext.createChildOf(ctxt)
headContext.createChildOf(ctxt)
{delegate.next token(); delegate.skipChildren(break
{delegate.next token(); delegate.skipChildren(break
tokenFilter.Include
_nextTokenWithBuffering( headContext
TokenFilter.Include
_headContext.createChildOf(ctxt)
f.includeValue(delegate
e); } next token is
```

