π	::=	$\overline{\delta}\ e$		program	τ ::= P	type parameter name
δ	::=	$\texttt{trait}\ T\overline{[\![v\ P <: \{\overline{\tau}\}]\!]} <: \{\overline{T[\![\overline{\tau}]\!]}\} \Diamond \{\overline{T[\![\overline{\tau}]\!]}\}\ \overline{\mu}\ \texttt{end}$		trait declaration	$\mid\ T[\![\overline{\tau}]\!]$	trait type
	$\mid$	$\texttt{object}\ O\overline{[\![P <: \{\overline{\tau}\}]\!]}(\overline{z{:}\,\tau}) <: \{\overline{T[\![\overline{\tau}]\!]}\}\ \overline{\mu}\ \texttt{end}$		object declaration	$\mid\ O[\![\overline{\tau}]\!]$	object type
	$\mid$	$f\overline{[\![P <: \{\overline{\tau}\}]\!]}(\overline{x{:}\,\tau}){:}\,\tau = e$		function declaration	$\mid\ (\overline{\tau})$	tuple type
μ	::=	$m\overline{[\![\{\overline{\tau}\} <: P <: \{\overline{\tau}\}]\!]}(\overline{x{:}\,\tau}){:}\,\tau = e$		method declaration	$\mid\ \tau \to \tau$	arrow type
v	::=	$\texttt{covariant} \mid \texttt{contravariant} \mid \texttt{invariant}$		variance	$\mid\ \mathrm{Any}$	special Any type
e	::=	x	variable reference	expression	$\mid\ \mathrm{Object}$	special Object type
	$\mid$	$\texttt{self}$	self reference		P ::= identifier	type parameter name
	$\mid$	$e.z$	field reference		T ::= identifier	generic trait name
	$\mid$	$x(\overline{e})$	function value application		O ::= identifier	generic object name
	$\mid$	$O[\![\overline{\tau}]\!](\overline{e})$	object creation		x ::= identifier	variable name
	$\mid$	$f[\![\overline{\tau}]\!](\overline{e})$	function invocation		z ::= identifier	field name
	$\mid$	$e.m[\![\overline{\tau}]\!](\overline{e})$	method invocation		f ::= identifier	function name
	$\mid$	$e\ \texttt{match}\ x{:}\,T \Rightarrow e\ \texttt{else}\ e$	match expression		m ::= identifier	method name

Figure 1. Grammar for Welterweight Fortress