Grammar for Language Describing Game Structures

Brooks MacLachlan

March 27, 2019

Pretty-printed EBNF grammar:

⟨*game*⟩ ::= 'Players:' ⟨*teamList*⟩ 'Rounds:' ⟨*roundList*⟩ 'Win:' ⟨*winConditionList*⟩

⟨*teamList*⟩ ::= (⟨*team*⟩ {',' ⟨*team*⟩} | ⟨*playerList*⟩ | 'randomly divide' ⟨*playerList*⟩ 'into' ⟨*nameList*⟩) ['all with' ⟨*attributeList*⟩]

⟨*team*⟩ ::= 'team' ⟨*name*⟩ ':' ⟨*playerList*⟩

⟨*playerList*⟩ ::= ⟨*player*⟩ {',' ⟨*player*⟩}

⟨*player*⟩ ::= ⟨*name*⟩ [⟨*attributeList*⟩]

⟨*attributeList*⟩ ::= ⟨*attribute*⟩ {',' ⟨*attribute*⟩}

⟨*attribute*⟩ ::= 'affiliation called' ⟨*name*⟩ | ('score' | 'resource' | 'counter') 'called' ⟨*name*⟩ ['starting at' ⟨*number*⟩] ['with minimum of' ⟨*number*⟩] ['with maximum of' ⟨*number*⟩]

⟨*roundList*⟩ ::= ⟨*round*⟩ ';' {⟨*round*⟩}

⟨*round*⟩ ::= ⟨*phaseList*⟩ ['repeated' ⟨*number*⟩ 'times' 'with modifications' ':' ⟨*modifierList*⟩]

⟨*modifierList*⟩ ::= ⟨*modifier*⟩ ';' {⟨*modifier*⟩}

⟨*modifier*⟩ ::= ['just' | 'from'] (⟨*roundReference*⟩ ('before' | 'after' | 'instead of') ⟨*phaseReference*⟩ 'insert' ⟨*phase*⟩)

⟨*roundReference*⟩ ::= 'round' [⟨*number*⟩]

⟨*phaseReference*⟩ ::= 'phase' [⟨*number*⟩]

⟨*phase*⟩ ::= ⟨*action*⟩ | ⟨*progression*⟩

$\langle phaseList \rangle ::= \langle phase \rangle$ {'\n' $\langle phase \rangle$}

$\langle action \rangle ::= \langle competition \rangle \mid \langle decision \rangle$

$\langle competition \rangle ::=$ ['scored'] ['team'] 'competition between' $\langle idList \rangle$

$\langle decision \rangle ::=$ ['self-included'] 'vote by' $\langle idList \rangle$ 'between' $\langle idList \rangle$ | ['self-included'] 'nomination of' $\langle number \rangle$ 'by' $\langle idList \rangle$ 'between' $\langle idList \rangle$ | 'allocation of' $\langle name \rangle$ 'by' $\langle idList \rangle$ | ['self-included'] 'directed vote by' $\langle idList \rangle$ 'between' $\langle idList \rangle$ | 'uses?' $\langle identifier \rangle$ 'then' '(' $\langle phaseList \rangle$ ')' ['otherwise' '(' $\langle phaseList \rangle$ ')']

$\langle progression \rangle ::= (\langle affiliationUpdate \rangle \mid \langle counterUpdate \rangle)$ ('for' | 'of') $\langle idList \rangle$

$\langle affiliationUpdate \rangle ::=$ 'elimination' | ('add' | 'remove') $\langle name \rangle$ | ['number preserving'] 'swap' $\langle nameList \rangle$ ['adding' $\langle nameList \rangle$] | 'change' $\langle name \rangle$ 'to' $\langle name \rangle$ | 'merge' $\langle nameList \rangle$ [$\langle name \rangle$]

$\langle counterUpdate \rangle ::=$ ('increase' | 'decrease') $\langle name \rangle$ 'by' $\langle value \rangle$ | 'set' $\langle name \rangle$ 'to' $\langle value \rangle$

$\langle value \rangle ::= \langle number \rangle \mid \langle name \rangle \mid$ 'results' 'of' ($\langle compReference \rangle \mid \langle allocateReference \rangle \mid \langle voteReference \rangle$)

$\langle idList \rangle ::= \langle identifier \rangle$ {',' $\langle identifier \rangle$} ['except' $\langle idList \rangle$]

$\langle identifier \rangle ::=$ ('everyone' | $\langle name \rangle$ | 'chance' '(' $\langle idList \rangle$ ')' | 'nominated' | 'tied' | 'eliminated' | ('winner' | 'loser') 'of' $\langle compReference \rangle$ | ('majority' | 'minority') 'of' $\langle voteReference \rangle$ [$\langle tiebreaker \rangle$] | ('highest' | 'lowest' | 'most' | 'least') $\langle name \rangle$ '(' $\langle idList \rangle$ ')' [$\langle tiebreaker \rangle$]) ['*' $\langle value \rangle$]

$\langle compReference \rangle ::=$ 'competition' [$\langle number \rangle$]

$\langle allocateReference \rangle ::=$ 'allocation' [$\langle number \rangle$]

$\langle voteReference \rangle ::=$ 'vote' [$\langle number \rangle$]

$\langle tiebreaker \rangle ::=$ 'tiebroken by' ([$\langle action \rangle$] $\langle identifier \rangle$)

$\langle name \rangle ::=$ ('a' | ... | 'z' | 'A' | ... | 'Z') {'a' | ... | 'z' | 'A' | ... | 'Z' | '0' | ... | '9'}

$\langle nameList \rangle ::= \langle name \rangle$ {',' $\langle name \rangle$}

$\langle number \rangle ::=$ ['-'] ('0' | ... | '9')+

$\langle winCondition \rangle ::=$ ['team'] 'reach' $\langle name \rangle$ 'of' $\langle number \rangle$ | ['team'] 'competition' | ['team'] ('highest' | 'lowest' | 'most' | 'least') $\langle name \rangle$ | $\langle number \rangle$ 'member jury vote' | 'survive'

$\langle winConditionList \rangle ::= \langle winCondition \rangle \; \{`,' \; \langle winCondition \rangle\}$

Notes:

- make grammar unambiguous

- should allocate be a decision and progression or just progression? Both, but decision will automatically also do the progression. The progression exists for non-decision related resource updates, such as awarding a player more of a resource for winning a competition.