# Grammar for Language Describing Game Structures

Brooks MacLachlan

March 21, 2019

**Pretty-printed EBNF grammar:**

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

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

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

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

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

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

⟨*attribute*⟩ ::= '`affiliation with`' ⟨*name*⟩ | ('`score`' | '`resource`' | '`counter`') ⟨*name*⟩ [⟨*number*⟩]

⟨*roundList*⟩ ::= ⟨*round*⟩ {'`,`' ⟨*round*⟩} [⟨*modifierList*⟩]

⟨*modifierList*⟩ ::= ⟨*modifier*⟩ {'`,`' ⟨*modifier*⟩}

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

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

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

⟨*round*⟩ ::= ⟨*phaseList*⟩ ['`repeated`' ⟨*number*⟩ '`times`']

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

⟨*phaseList*⟩ ::= ⟨*phase*⟩ {'`>`' ⟨*phase*⟩}

⟨*action*⟩ ::= ⟨*competition*⟩ | ⟨*decision*⟩

⟨*competition*⟩ ::= [‘`scored`’] [‘`team`’] ‘`competition between`’ ⟨*idList*⟩

⟨*decision*⟩ ::= [‘`self-included`’] ‘`vote by`’ ⟨*idList*⟩ ‘`between`’ ⟨*idList*⟩ [‘`tiebroken by`’ (⟨*competition*⟩ | ⟨*decision*⟩ | ⟨*identifier*⟩)] | [‘`self-included`’] ‘`nomination of`’ ⟨*number*⟩ ‘`by`’ ⟨*identifier*⟩ ‘`between`’ ⟨*idList*⟩ | ‘`allocate`’ ⟨*name*⟩ | [‘`self-included`’] ‘`directed vote by`’ ⟨*idList*⟩ ‘`between`’ ⟨*idList*⟩ | ‘`uses?`’ ⟨*identifier*⟩ ‘`then`’ ‘`(`’ ⟨*phaseList*⟩ ‘`)`’ [‘`otherwise`’ ‘`(`’ ⟨*phaseList*⟩ ‘`)`’]

⟨*progression*⟩ ::= (⟨*affiliationUpdate*⟩ | ⟨*counterUpdate*⟩) ‘`for`’ ⟨*idList*⟩

⟨*affiliationUpdate*⟩ ::= ‘`elimination`’ | (‘`add`’ | ‘`remove`’) ⟨*name*⟩ | [‘`number preserving`’] ‘`swap`’ ⟨*nameList*⟩ [‘`adding`’ ⟨*nameList*⟩] | ‘`change`’ ⟨*name*⟩ ‘`to`’ ⟨*name*⟩ | ‘`merge`’ ⟨*nameList*⟩ [⟨*name*⟩]

⟨*counterUpdate*⟩ ::= (‘`increase`’ | ‘`decrease`’) ⟨*name*⟩ ‘`by`’ ⟨*value*⟩ | ‘`set`’ ⟨*name*⟩ ‘`to`’ ⟨*value*⟩

⟨*value*⟩ ::= ⟨*number*⟩ | ⟨*name*⟩ | (⟨*resultReference*⟩ | ⟨*allocateReference*⟩ | ⟨*voteReference*⟩) ‘`results`’

⟨*idList*⟩ ::= ⟨*identifier*⟩ {‘`,`’ ⟨*identifier*⟩} [‘`except`’ ⟨*idList*⟩]

⟨*identifier*⟩ ::= (‘`everyone`’ | ⟨*name*⟩ | ‘`chance`’ | ‘`random`’ | ‘`nominated`’ | ‘`tied`’ | ‘`eliminated`’ | ⟨*compReference*⟩ (‘`winner`’ | ‘`loser`’) | ⟨*voteReference*⟩ (‘`majority`’ | ‘`minority`’) | (‘`highest`’ | ‘`lowest`’ | ‘`most`’ | ‘`least`’) ⟨*name*⟩ ) [‘`*`’ ⟨*value*⟩]

⟨*compReference*⟩ ::= ‘`competition`’ [⟨*number*⟩]

⟨*allocateReference*⟩ ::= ‘`allocation`’ [⟨*number*⟩]

⟨*voteReference*⟩ ::= ‘`vote`’ [⟨*number*⟩]

⟨*name*⟩ ::= (‘`a`’ | ... | ‘`z`’ | ‘`A`’ | ... | ‘`Z`’) {‘`a`’ | ... | ‘`z`’ | ‘`A`’ | ... | ‘`Z`’ | ‘`0`’ | ... | ‘`9`’}

⟨*nameList*⟩ ::= ⟨*name*⟩ {‘`,`’ ⟨*name*⟩}

⟨*number*⟩ ::= (‘`0`’ | ... | ‘`9`’)+

⟨*winCondition*⟩ ::= [‘`team`’] ‘`achieving`’ ⟨*name*⟩ ‘`of`’ ⟨*number*⟩ | [‘`team`’] ‘`competition`’ | [‘`team`’] (‘`highest`’ | ‘`lowest`’ | ‘`most`’ | ‘`least`’) ⟨*name*⟩ | ⟨*number*⟩ ‘`member jury vote`’

⟨*winConditionList*⟩ ::= ⟨*winCondition*⟩ {‘`,`’ ⟨*winCondition*⟩}

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.