

# Grammar for Language Describing Game Structures

Brooks MacLachlan

March 22, 2019

## Pretty-printed EBNF grammar:

```
 $\langle game \rangle ::= \text{'Players:'} \langle teamList \rangle \text{'Rounds:'} \langle roundList \rangle \text{'Win:'} \langle winConditionList \rangle$   
 $\langle teamList \rangle ::= (\langle team \rangle \{', ' \langle team \rangle\} \mid \langle playerList \rangle \mid \text{'randomly divide'} \langle playerList \rangle$   
     $\text{'into'} \langle nameList \rangle) [\text{'having'} \langle attributeList \rangle]$   
 $\langle team \rangle ::= \text{'team'} \langle name \rangle \text{'::'} \langle playerList \rangle$   
 $\langle playerList \rangle ::= \langle player \rangle \{', ' \langle player \rangle\}$   
 $\langle player \rangle ::= \langle name \rangle [\langle attributeList \rangle]$   
 $\langle attributeList \rangle ::= \langle attribute \rangle \{', ' \langle attribute \rangle\}$   
 $\langle attribute \rangle ::= \text{'affiliation with'} \langle name \rangle \mid (\text{'score'} \mid \text{'resource'} \mid \text{'counter'})$   
     $\langle name \rangle [\langle number \rangle] [\text{'min'} \langle number \rangle] [\text{'max'} \langle number \rangle]$   
 $\langle roundList \rangle ::= \langle round \rangle \{', ' \langle round \rangle\} [\langle modifierList \rangle]$   
 $\langle modifierList \rangle ::= \langle modifier \rangle \{', ' \langle modifier \rangle\}$   
 $\langle modifier \rangle ::= [\text{'just'} \mid \text{'from'}] (\langle roundReference \rangle (\text{'before'} \mid \text{'after'}) \langle phaseReference \rangle$   
     $\text{'insert'} \langle phase \rangle \mid \langle roundReference \rangle \text{'replace'} \langle phaseReference \rangle \text{'with'} \langle phase \rangle)$   
 $\langle roundReference \rangle ::= \text{'round'} [\langle number \rangle]$   
 $\langle phaseReference \rangle ::= \text{'phase'} [\langle number \rangle]$   
 $\langle round \rangle ::= \langle phaseList \rangle [\text{'repeated'} \langle number \rangle \text{'times'}]$   
 $\langle phase \rangle ::= \langle action \rangle \mid \langle progression \rangle$   
 $\langle phaseList \rangle ::= \langle phase \rangle \{ '>' \langle phase \rangle \}$ 
```

$\langle action \rangle ::= \langle competition \rangle \mid \langle decision \rangle$   
 $\langle competition \rangle ::= [\text{'scored'}] [\text{'team'}] \text{'competition between'} \langle idList \rangle$   
 $\langle decision \rangle ::= [\text{'self-included'}] \text{'vote by'} \langle idList \rangle \text{'between'} \langle idList \rangle [\text{'tiebroken by'} \langle \langle competition \rangle \mid \langle decision \rangle \mid \langle identifier \rangle \rangle] \mid [\text{'self-included'}] \text{'nomination of'} \langle number \rangle \text{'by'} \langle identifier \rangle \text{'between'} \langle idList \rangle \mid \text{'allocation of'} \langle name \rangle \text{'by'} \langle idList \rangle \mid [\text{'self-included'}] \text{'directed vote by'} \langle idList \rangle \text{'between'} \langle idList \rangle \mid \text{'uses?' } \langle identifier \rangle \text{'then'} \text{'(' } \langle phaseList \rangle \text{'('} \mid \text{'otherwise'} \text{'(' } \langle phaseList \rangle \text{'('}]$   
 $\langle progression \rangle ::= (\langle affiliationUpdate \rangle \mid \langle counterUpdate \rangle) \text{'for'} \langle idList \rangle$   
 $\langle affiliationUpdate \rangle ::= \text{'elimination'} \mid (\text{'add'} \mid \text{'remove'}) \langle name \rangle \mid [\text{'number preserving'}] \text{'swap'} \langle nameList \rangle [\text{'adding'} \langle nameList \rangle] \mid \text{'change'} \langle name \rangle \text{'to'} \langle name \rangle \mid \text{'merge'} \langle nameList \rangle [\langle name \rangle]$   
 $\langle counterUpdate \rangle ::= (\text{'increase'} \mid \text{'decrease'}) \langle name \rangle \text{'by'} \langle value \rangle \mid \text{'set'} \langle name \rangle \text{'to'} \langle value \rangle$   
 $\langle value \rangle ::= \langle number \rangle \mid \langle name \rangle \mid (\langle compReference \rangle \mid \langle allocateReference \rangle \mid \langle voteReference \rangle) \text{'results'} \mid \langle identifier \rangle \langle name \rangle$   
 $\langle idList \rangle ::= \langle identifier \rangle \{ \text{'}, \langle identifier \rangle \} [\text{'except'} \langle idList \rangle]$   
 $\langle identifier \rangle ::= (\text{'everyone'} \mid \langle name \rangle \mid \text{'chance'} \mid \text{'random'} \mid \text{'nominated'} \mid \text{'tied'} \mid \text{'eliminated'} \mid \langle compReference \rangle (\text{'winner'} \mid \text{'loser'}) \mid \langle voteReference \rangle (\text{'majority'} \mid \text{'minority'}) \mid (\text{'highest'} \mid \text{'lowest'} \mid \text{'most'} \mid \text{'least'}) \langle name \rangle) [\text{'*'} \langle value \rangle]$   
 $\langle compReference \rangle ::= \text{'competition'} [\langle number \rangle]$   
 $\langle allocateReference \rangle ::= \text{'allocation'} [\langle number \rangle]$   
 $\langle voteReference \rangle ::= \text{'vote'} [\langle number \rangle]$   
 $\langle name \rangle ::= (\text{'a'} \mid \dots \mid \text{'z'} \mid \text{'A'} \mid \dots \mid \text{'Z'}) \{ \text{'a'} \mid \dots \mid \text{'z'} \mid \text{'A'} \mid \dots \mid \text{'Z'} \mid \text{'0'} \mid \dots \mid \text{'9'} \}$   
 $\langle nameList \rangle ::= \langle name \rangle \{ \text{'}, \langle name \rangle \}$   
 $\langle number \rangle ::= (\text{'0'} \mid \dots \mid \text{'9'})^+$   
 $\langle winCondition \rangle ::= [\text{'team'}] \text{'achieving'} \langle name \rangle \text{'of'} \langle number \rangle \mid [\text{'team'}] \text{'competition'} \mid [\text{'team'}] (\text{'highest'} \mid \text{'lowest'} \mid \text{'most'} \mid \text{'least'}) \langle name \rangle \mid \langle number \rangle \text{'member jury vote'} \mid \text{'last one standing'}$   
 $\langle winConditionList \rangle ::= \langle winCondition \rangle \{ \text{'}, \langle winCondition \rangle \}$

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.
- ties for non-votes? Like lowest score? Maybe I need some sort of global tiebreaker setting.