
CCM Programming Challenge: Dotsville Implementation, Part 3

This programming challenge pertains to the implementation of a **recognizer** for for Dotspeak, a little language that interfaces with Dotsville.

Preliminary Notes

1. Please be sure that you have completed Part 2 of the Dotsville implementation exercise prior to commencing this part, Part 3, of the Dotsville implementation exercise.
2. Please be sure to work by analogy with the Purple recognizer presented in class, recalling that a CFG was provided for Purple, and then a DCG was written to implement the CFG in Prolog.
3. This assignment should be very straightforward for you. It calls for a “mechanical” translation of a CFG to a DCG, and then for you to copy the **recognizer** program that appears in the Purple recognizer program (`purple_r.pro`).

Notes on the Dotspeak Language

1. Only six colors: RED, BLUE, YELLOW, ORANGE, PURPLE, GREEN
2. Only eight column numbers, each of which can be represented by digit or name: 1 or one, 2 or two, 3 or three, 4 or four, 5 or five, 6 or six, 7 or seven, 8 or eight.
3. Only eight sprinkling numbers, each of which can be represented by digit or name: 2 or two, 3 or three, 4 or four, 5 or five, 6 or six, 7 or seven, 8 or eight, 9 or nine.
4. Indefinite articles should match their following word.

Tasks

Simply perform the following tasks:

1. Write down 3 dozen **representative** sentences in the Dotspeak language as defined by the given CFG. By “representative” I mean that you should include sentences in the set that are **structurally** diverse. Be sure to include plenty of commands and plenty questions, and also a number of existential questions, count questions, content questions, display commands, add commands, delete commands, and movement commands. **Number each of the 36 sentences.** Place these sentences in a file called `example_dotspeak_sentences.text`
2. In a file called `dotspeak_r.pro` place code that is analogous to that found in `purple_r.pro` in order to produce a program to recognize sentences in the Dotspeak language. In particular, you should place the following items into the file:
 - (a) Appropriate comments
 - (b) An “import” statement for the IO library (`io.pro`)
 - (c) The DCG, translated “mechanically” from the CFG provided
 - (d) The recognizer program, lifted from `purple_r.pro`, that repeatedly reads a sentence and endeavors to recognize it.
3. A demo in which you feed the 3 dozen sentences to the recognizer program, intermixed with a dozen near misses. Call this file `dotspeak_r_demo.text`.

4. Please post the 3 files to your web work site.

Due Date

Wednesday, November 4, 2020

CFG for Dotspeak

1. sentence \rightarrow command .
2. sentence \rightarrow question ?
3. question \rightarrow existentialquestion
4. question \rightarrow countquestion
5. question \rightarrow contentquestion
6. command \rightarrow displaycommand
7. command \rightarrow addcommand
8. command \rightarrow deletecommand
9. command \rightarrow movecommand
10. command \rightarrow stop
11. existentialquestion \rightarrow is there nsdot
12. existentialquestion \rightarrow is there nsdot on the table
13. existentialquestion \rightarrow is there nsdot not on the table
14. existentialquestion \rightarrow is there csdot
15. existentialquestion \rightarrow is there csdot on the table
16. existentialquestion \rightarrow is there csdot not on the table
17. existentialquestion \rightarrow is there csdot to the left of csdot?
18. existentialquestion \rightarrow is there csdot to the right of csdot?
19. countquestion \rightarrow countdots in column columnid
20. countquestion \rightarrow countdots in the world
21. countquestion \rightarrow countdots on the table
22. countquestion \rightarrow countdots off the table
23. countquestion \rightarrow countcolordots in column columnid
24. countquestion \rightarrow countcolordots in the world
25. countquestion \rightarrow countspaces in column columnid
26. countquestion \rightarrow countspaces in the world
27. contentquestion \rightarrow is column columnid empty
28. contentquestion \rightarrow is column columnid full
29. contentquestion \rightarrow is the world empty
30. contentquestion \rightarrow is the world full
31. displaycommand \rightarrow list the dots

32. `displaycommand` → list the color dots
33. `displaycommand` → list the dots in column `columnid`
34. `displaycommand` → list the dots on the table
35. `displaycommand` → list the dots off the table
36. `addcommand` → add `nsdot`
37. `addcommand` → add `csdot`
38. `addcommand` → add `nsdot` to column `columnid`
39. `addcommand` → add `csdot` to column `columnid`
40. `addcommand` → fill column `columnid`
41. `addcommand` → fill the world
42. `addcommand` → sprinkle `xdigit` dots onto the world
43. `deletecommand` → remove `nsdot`
44. `deletecommand` → remove `nsdot` from column `columnid`
45. `deletecommand` → clear the world
46. `movecommand` → move `nsdot` from column `columnid` to column `columnid`
47. `countdots` → how many dots
48. `countcolordots` → how many color dots
49. `countspaces` → how many spaces available
50. `nsdot` → a dot
51. `csdot` → detcolor dot
52. `detcolor` → an orange
53. `detcolor` → a nonorange
54. `nonorange` → blue
55. `nonorange` → yellow
56. `nonorange` → red
57. `nonorange` → green
58. `nonorange` → purple
59. `color` → orange
60. `color` → blue
61. `color` → yellow
62. `color` → red
63. `color` → green
64. `color` → purple
65. `xdigit` → two
66. `xdigit` → 2
67. `xdigit` → three
68. `xdigit` → 3
69. `xdigit` → four
70. `xdigit` → 4
71. `xdigit` → five

- 72. `xdigit` \rightarrow 5
- 73. `xdigit` \rightarrow six
- 74. `xdigit` \rightarrow 6
- 75. `xdigit` \rightarrow seven
- 76. `xdigit` \rightarrow 7
- 77. `xdigit` \rightarrow eight
- 78. `xdigit` \rightarrow 8
- 79. `xdigit` \rightarrow nine
- 80. `xdigit` \rightarrow 9
- 81. `columnid` \rightarrow one
- 82. `columnid` \rightarrow 1
- 83. `columnid` \rightarrow two
- 84. `columnid` \rightarrow 2
- 85. `columnid` \rightarrow three
- 86. `columnid` \rightarrow 3
- 87. `columnid` \rightarrow four
- 88. `columnid` \rightarrow 4
- 89. `columnid` \rightarrow five
- 90. `columnid` \rightarrow 5
- 91. `columnid` \rightarrow six
- 92. `columnid` \rightarrow 6
- 93. `columnid` \rightarrow seven
- 94. `columnid` \rightarrow 7
- 95. `columnid` \rightarrow eight
- 96. `columnid` \rightarrow 8