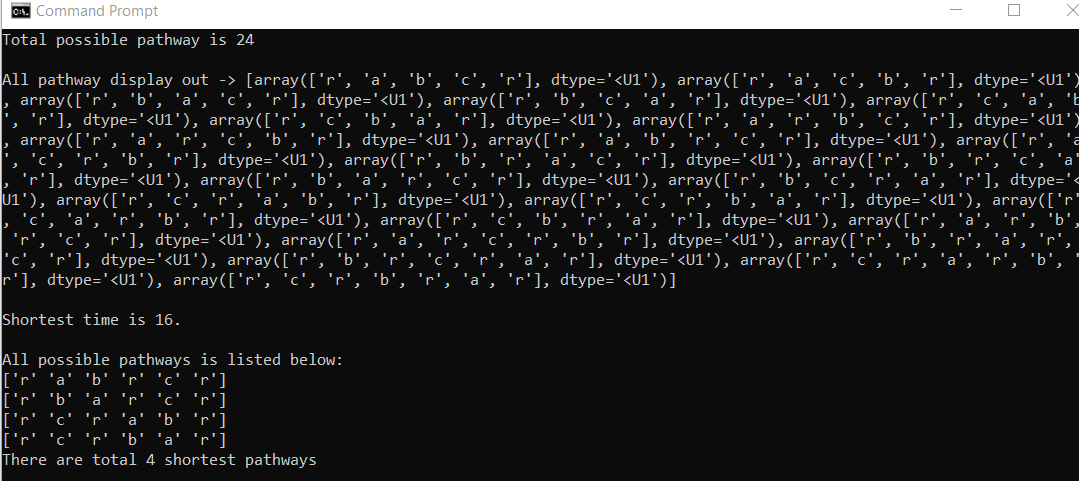
1a) There are 4 most efficient pathway: RABRCR, RBARCR, RCRABR and RCRBAR.

With time =16

I know as I made a python program for this question (wrote in pycharm IDE). See the file I uploaded. I import numpy and itertool (default have) only into the program.

Below picture is the output of the program:

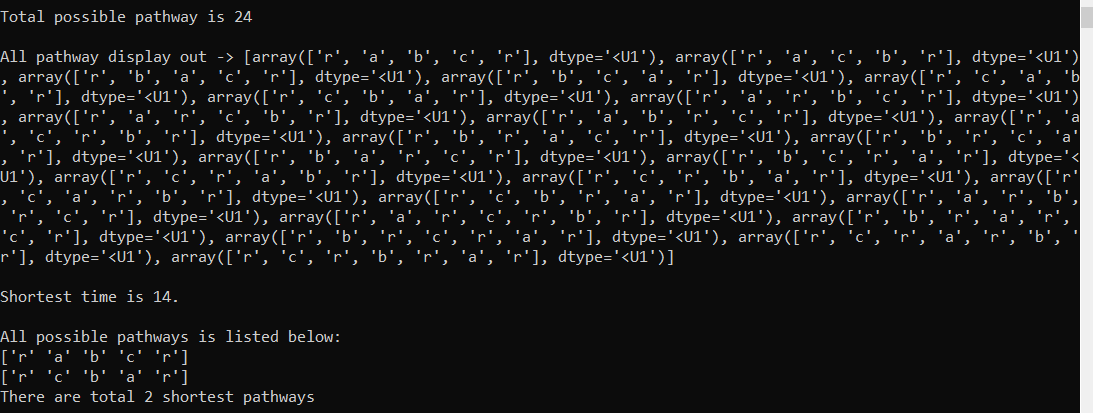


I first generate all pathways for 5(min step), 6, 7(max step) step pathways by itertool.product, then use custom loop to filter out impossible pathways. Though I think this is very inefficient way.

Then I brute force all possible pathways to find all possible shortest pathways, with shortest time.

1)b) I made a program to calculate out shortest pathways.

2)a) update the time value in the program mentioned, new output:



Thus, new shortest pathways = RABCR and RCBAR

2)b) I updated the new time in the program and let it recompute new shortest pathways.

3)a)

Rewrite my program dictionary “time\_subpath” to “time\_subpath = {'ab':x,'ba':x,'ra':w,'ar':w,'ac':y,'ca':y,'br':u,'rb':u,'cr':v,'rc':v,'bc':z,'cb':z}” at line 5

According to new time, then calculate new shortest pathway again. And remind he/she to run with python 3 with numpy imported, this programe will work if total time =< 999999 they can change the limit at “shortest\_time = 999999” at line 368.