*// Code*

*from* csp *import* \*

neighbours = {

     "REKTORAT": ["LSI"],

     "LSI": ["REKTORAT", "FEM", "FATETA"],

     "FEM": ["LSI", "FEMA"],

     "FATETA": ["LSI", "FEMA", "FAHUTAN"],

     "FEMA": ["FEM", "FATETA", "FAHUTAN", "FAPERTA"],

     "FAHUTAN": ["FEMA", "FATETA", "FAPERTA", "FMIPA"],

     "FAPERTA": ["FEMA", "FAHUTAN", "FMIPA"],

     "FMIPA": ["FAHUTAN", "FAPERTA"]

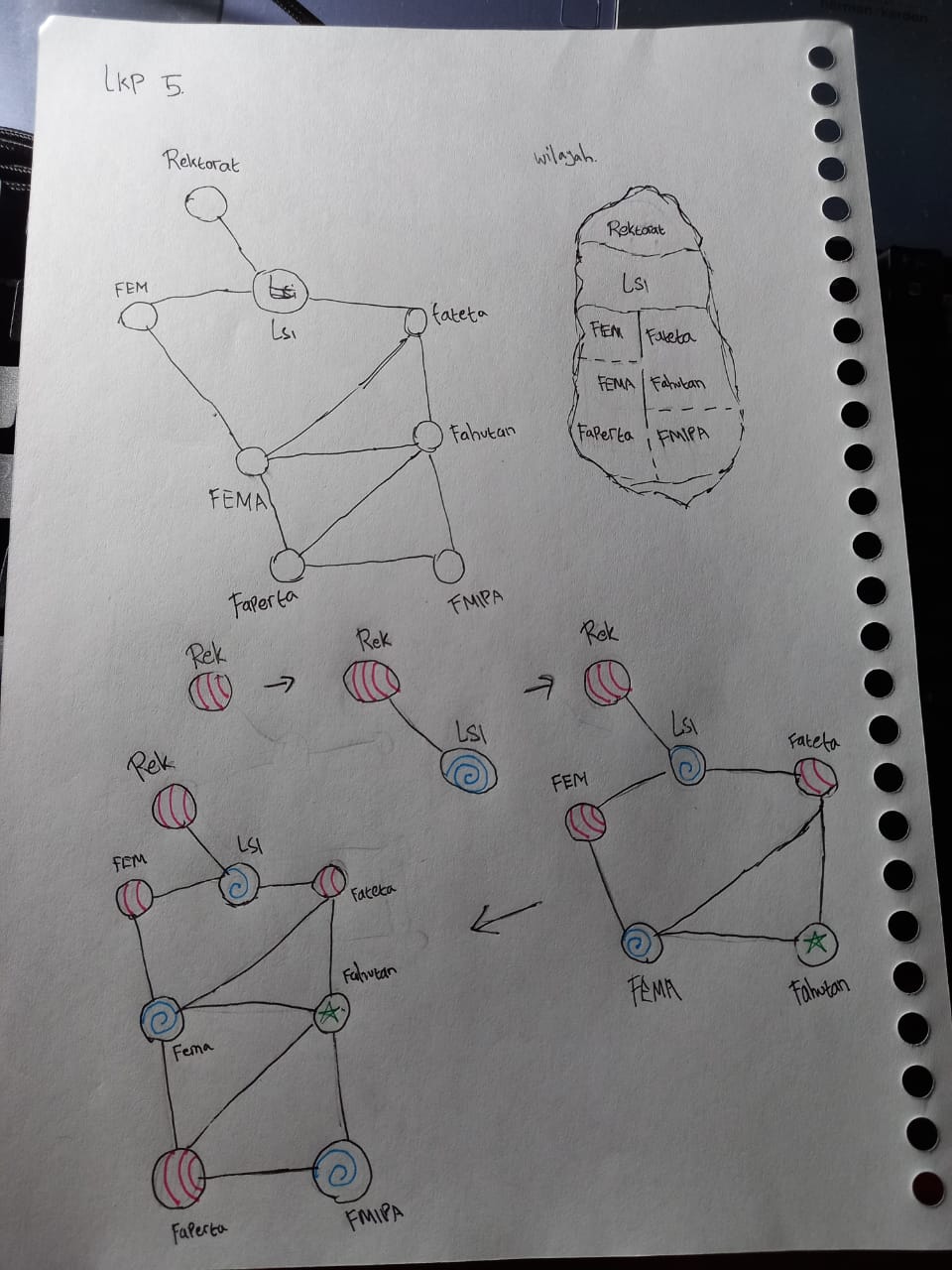
}

coloring\_problem = MapColoringCSP('RGBY', neighbours)

result1 = backtracking\_search(coloring\_problem)

print(result1)

// Manual



Dari code nya, awalnya adalah kita membuat node node neighbours, atau node node yang saling berhubungan satu sama lain, seperti Rektorat yang berhubungan ke LSI, setelah membuat neighbours, data itu akan dipakai untuk fungsi MapColoringCSP, fungsi ini berguna untuk mewarnai tiap nodes nodes nya, dengan warna yang berbeda di nodes yang berhubungan, setelah itu juga dipakai di backtracking search karena dengan backtracking search akan dicari jalan paling efektif untuk meminimalkan warna yang dipakai. Lalu hasil nya pun di print

Untuk Teknik manualnya sendiri, saya memulai dari Rektorat dengan warna merah, lalu lanjut ke LSI dengan warna biru , karena tidak boleh menggunakan warna merah lagi, lalu ke FEM , awalnya saya ingin mewarnai dengan warna hijau tetapi menyadari itu membuat tidak efektif. Maka saya back track. Kembali ke LSI, lalu membuat FEM dan FATETA menjadi merah lagi, dari FEM ke FEMA saya warnai biru lagi, lalu berikutnya ke FAHUTAN, FAHUTAN ini di hubungkan oleh FATETA ( merah) dan FEMA ( biru ) maka FAHUTAN menjadi warna lain yaitu hijau, sama seperti itu juga ke node nodes lainnya