

به نام خدا

۹۸۲۴۲۵۳

سپهر قاسمی نژاد

پروژه چهارم – گرگ و گوسفند

ورودی

ابتدا عدد n و سپس در n خط n عدد می آید که ماتریس همجواری گراف است. سپس شماره خانه شروع و مقصد می آیند.

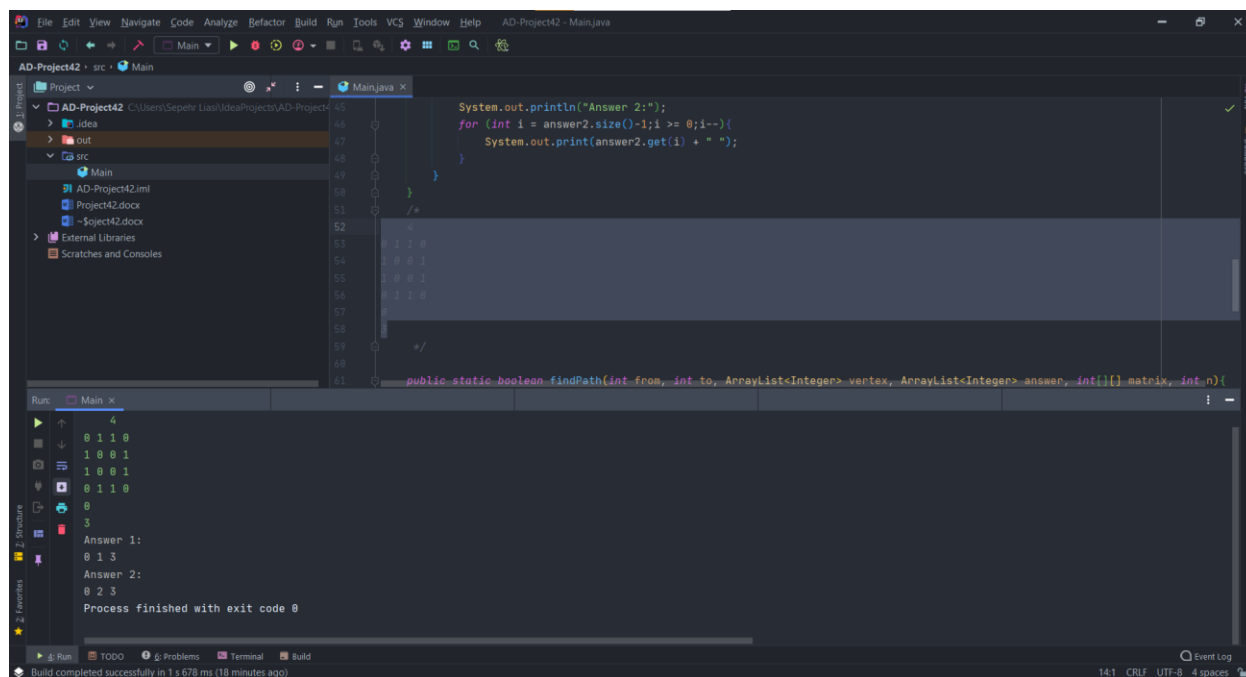
خروجی

در خروجی دو مسیر نمایش داده میشود. اگر کمتر از دو مسیر وجود داشت، چیزی نمایش داده نمیشود.

الگوریتم

الگوریتم به صورت بازگشتی است و همان رویه $\max\ flow$ پی گرفته میشود و میتوان اینطور نتیجه گرفت که حداقل به میزان $\max\ flow$ ، مسیر غیر مرتبط بین آن دو نقطه وجود دارد. پس با اجرای الگوریتم Ford Fulkerson توسط تابع بازگشتی (عقب گرد)، دو بار این الگوریتم اجرا میشود تا دو مسیر را به ما بدهد. لازم است در بین دو مرحله، ماتریس مجاورت به روزرسانی شود چرا که نمیتوان از آن مسیرهای راه اول در راه دوم استفاده کرد.

مثال از روند اجرا



The screenshot displays an IDE window for a project named "AD-Project42". The main editor shows a Java file "Main.java" with the following code:

```
45 System.out.println("Answer 2:");
46 for (int i = answer2.size()-1; i >= 0; i--){
47     System.out.print(answer2.get(i) + " ");
48 }
49
50 }
51
52 /*
53 0 1 1 0
54 1 0 0 1
55 1 0 0 1
56 0 1 1 0
57
58
59 */
60
61 public static boolean findPath(int from, int to, ArrayList<Integer> vertex, ArrayList<Integer> answer, int[][] matrix, int n){
```

The "Run" tab at the bottom shows the execution output:

```
4
0 1 1 0
1 0 0 1
1 0 0 1
0 1 1 0
0
3
Answer 1:
0 1 3
Answer 2:
0 2 3
Process finished with exit code 0
```

The status bar at the bottom indicates "Build completed successfully in 1 s 678 ms (10 minutes ago)".