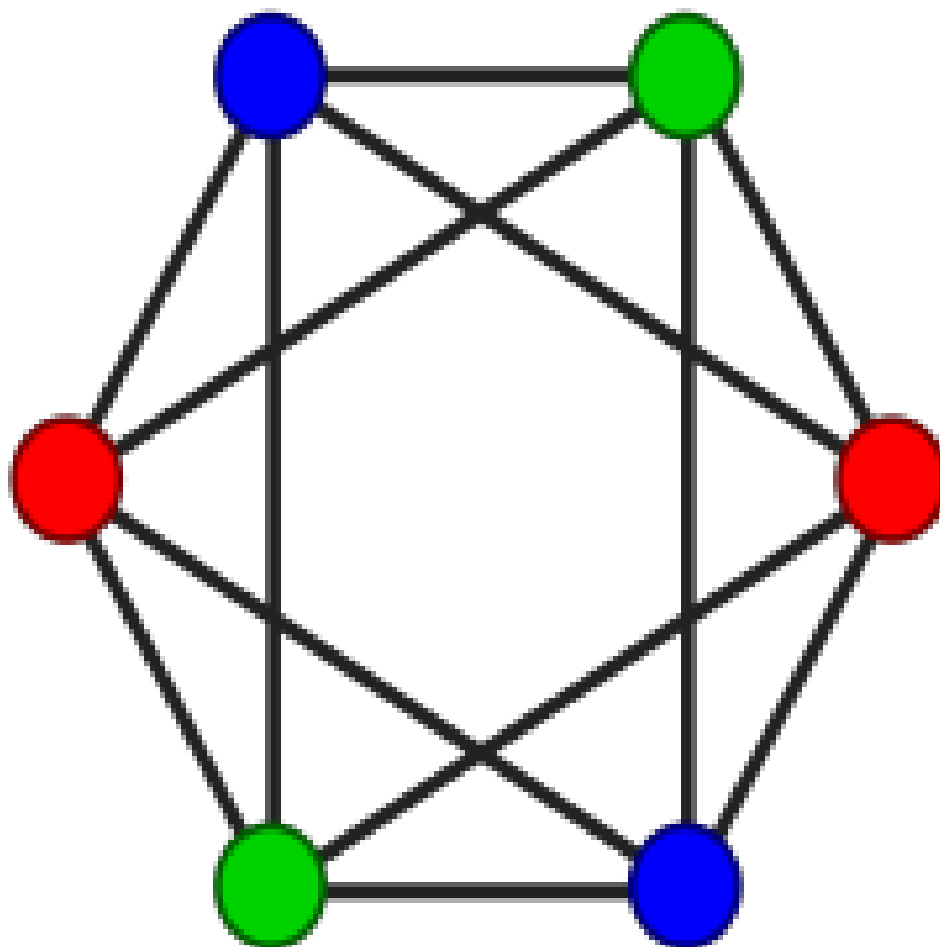


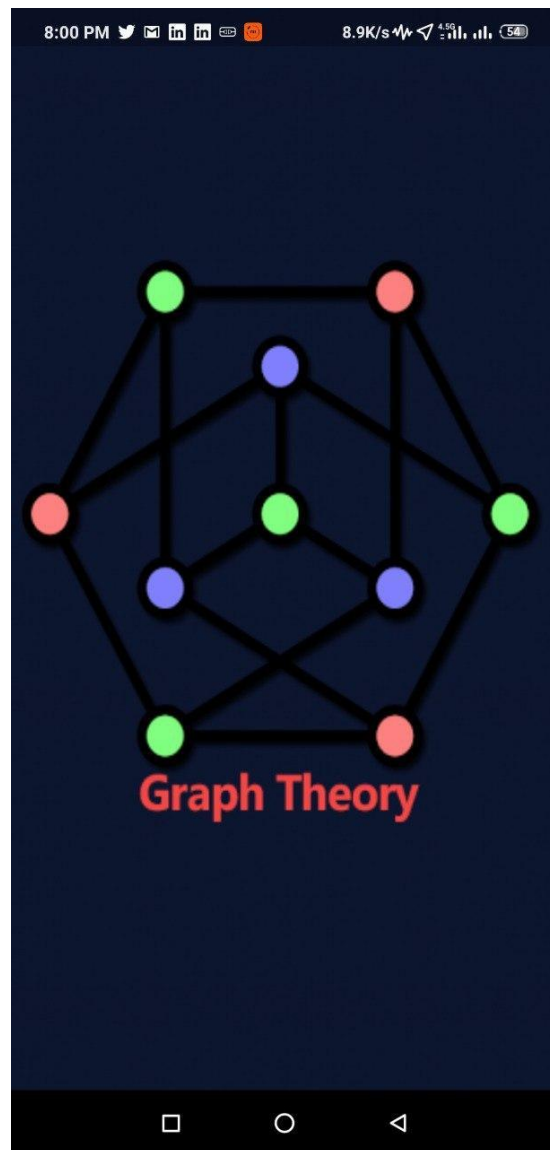
داکیومنت پروژه اپلیکیشن الگوریتم های گراف

امیرحسین ایزدجو

پرهام عابدآزاد



۱. SplashScreen



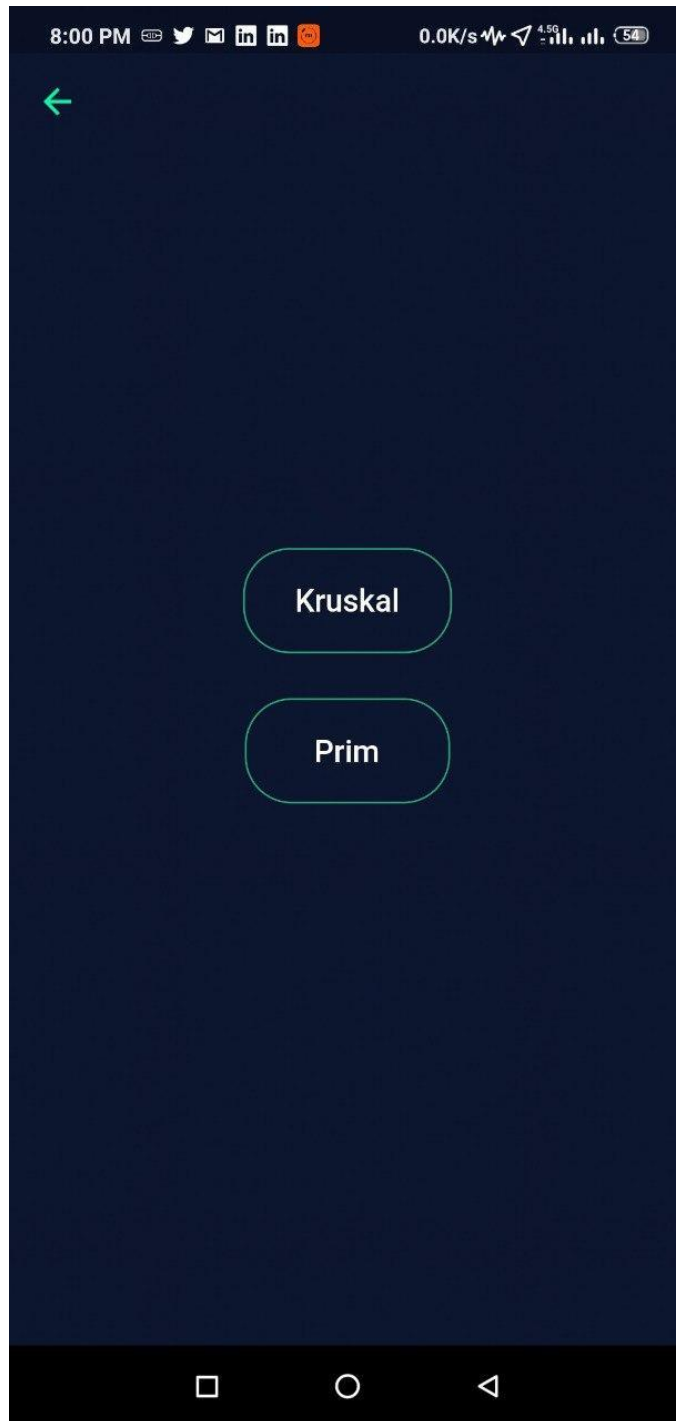
کد این صفحه در فایل `lib/SplashScreen.dart` قابل مشاهده است. پس از ۳ ثانیه کاربر وارد صفحه اصلی میشود.

۲. MainPage



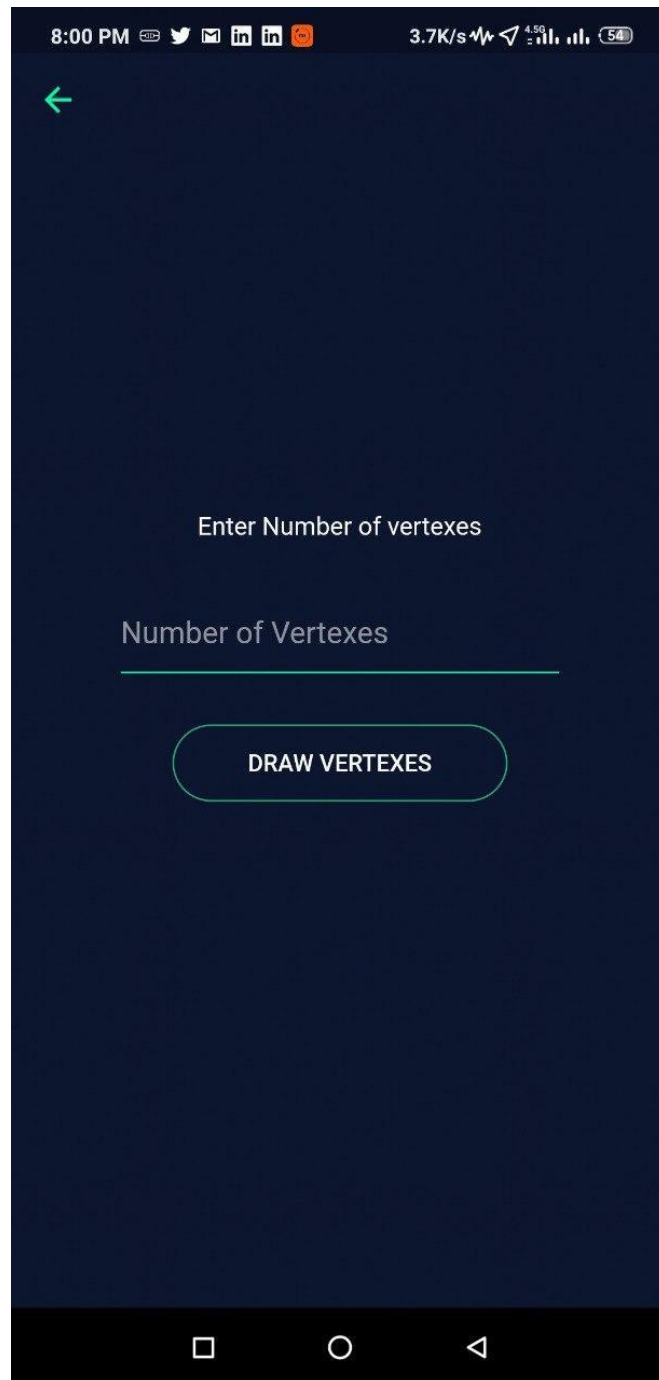
کد این صفحه در فایل `lib/main.dart` قابل مشاهده است. دو دکمه در صفحه وجود دارد که دکمه اول کاربر را به صفحه گرفتن تعداد رئوس الگوریتم ماکسیمم مچینگ میبرد و دکمه پایین ما را به صفحه انتخاب بین دو الگوریتم `prim` و `kruskal` میبرد.

MST Page ۳.



کد این صفحه در فایل `lib/views/mst.dart` قابل مشاهده است. دو دکمه در صفحه وجود دارد که هر کدام ما را به صفحه گرفتن تعداد رئوس برای الگوریتم های `kruskal` یا `prim` میبرند.

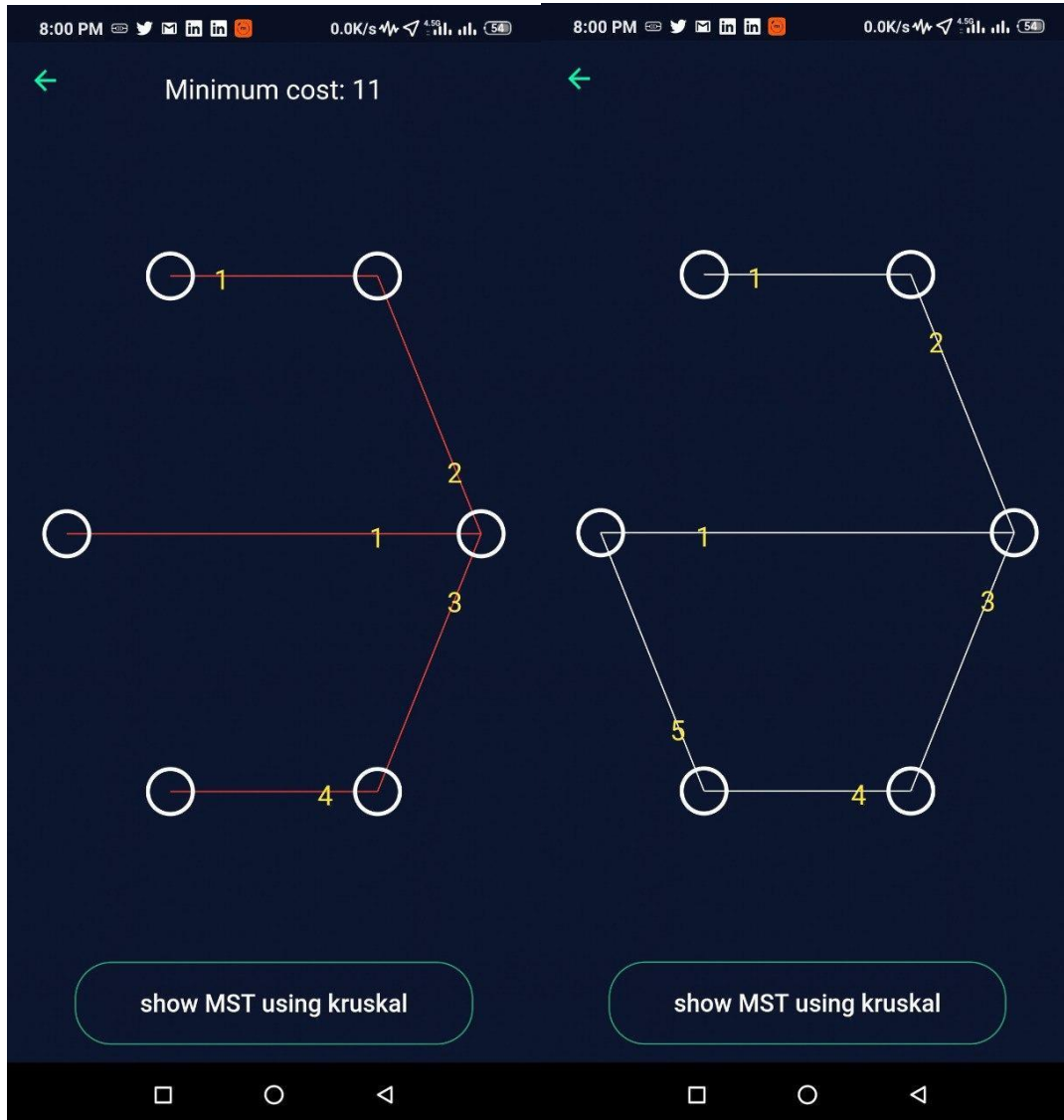
۴. Kruskal and prim Input Pages



کد این صفحه در دو فایل `lib/views/kruskalinput.dart` و `lib/views/priminput.dart` وجود دارد.

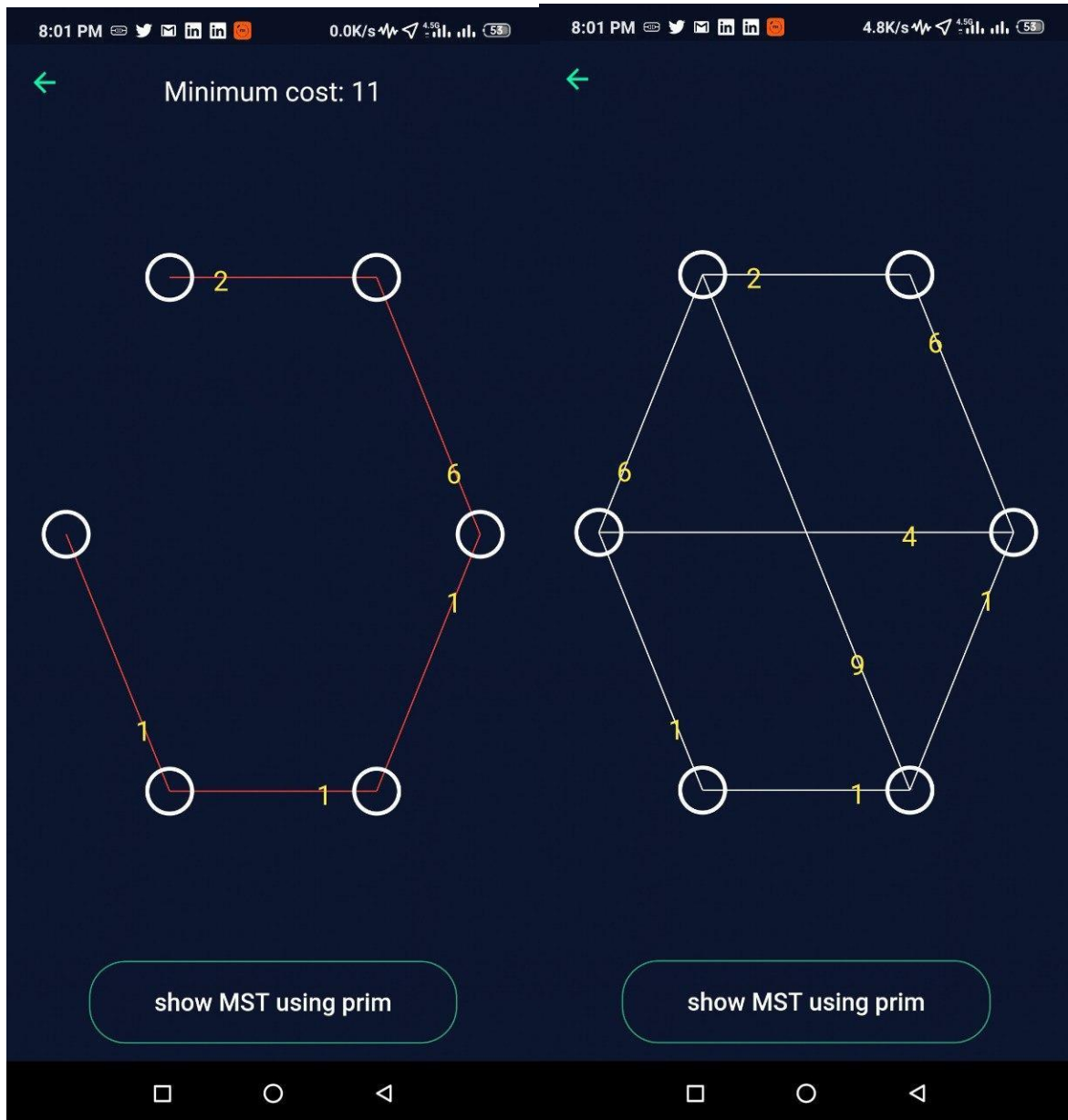
در این صفحه تعداد رئوس گرفته شده و هر کدام به صفحه اجرای الگوریتم های کروسکال یا پریم میروند

۵. Show Kruskal Page



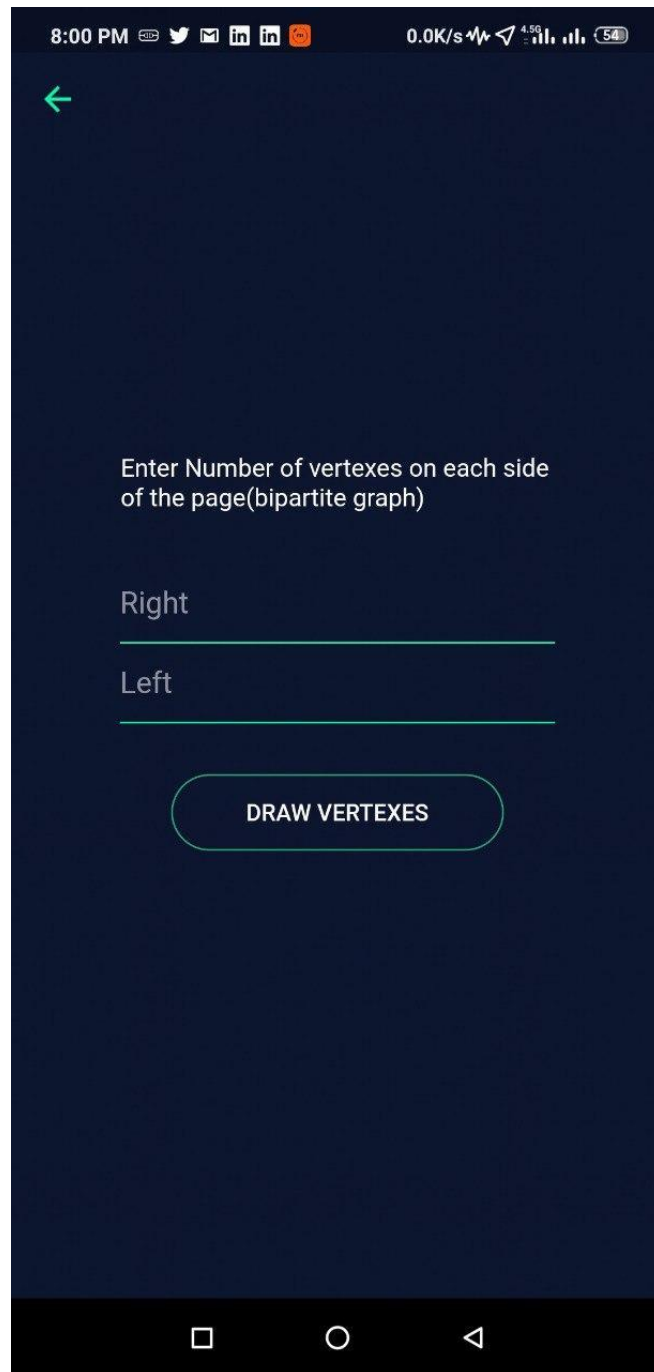
در این صفحه با فشار دادن یک دکمه دایره شکل (راس) آن دکمه به رنگ آبی در خواهد آمد. سپس میتوانید روی یک راس دیگر کلیک کنید و اگر یالی بین این دو راس وجود نداشته باشد، یک پاپ آپ باز خواهد شد که وزن یال را میپرسد. پس از وارد کردن وزن، یال در صفحه وارد میشود. پس از وارد کردن یال ها میتوانید دکمه پایین صفحه را فشار دهید تا نتیجه الگوریتم کروسکال را ببینید. کد و الگوریتم این صفحه در آدرس <lib/views/showkruskal.dart> قابل مشاهده است. در صورتی که گراف همبند نباشد، برنامه ارور همبند نبودن گراف را به شما نشان خواهد داد. پس از دیدن نتیجه الگوریتم، با کلیک بر روی رئوس میتوانید به اضافه کردن یال ها ادامه دهید.

6. Show Prim Page



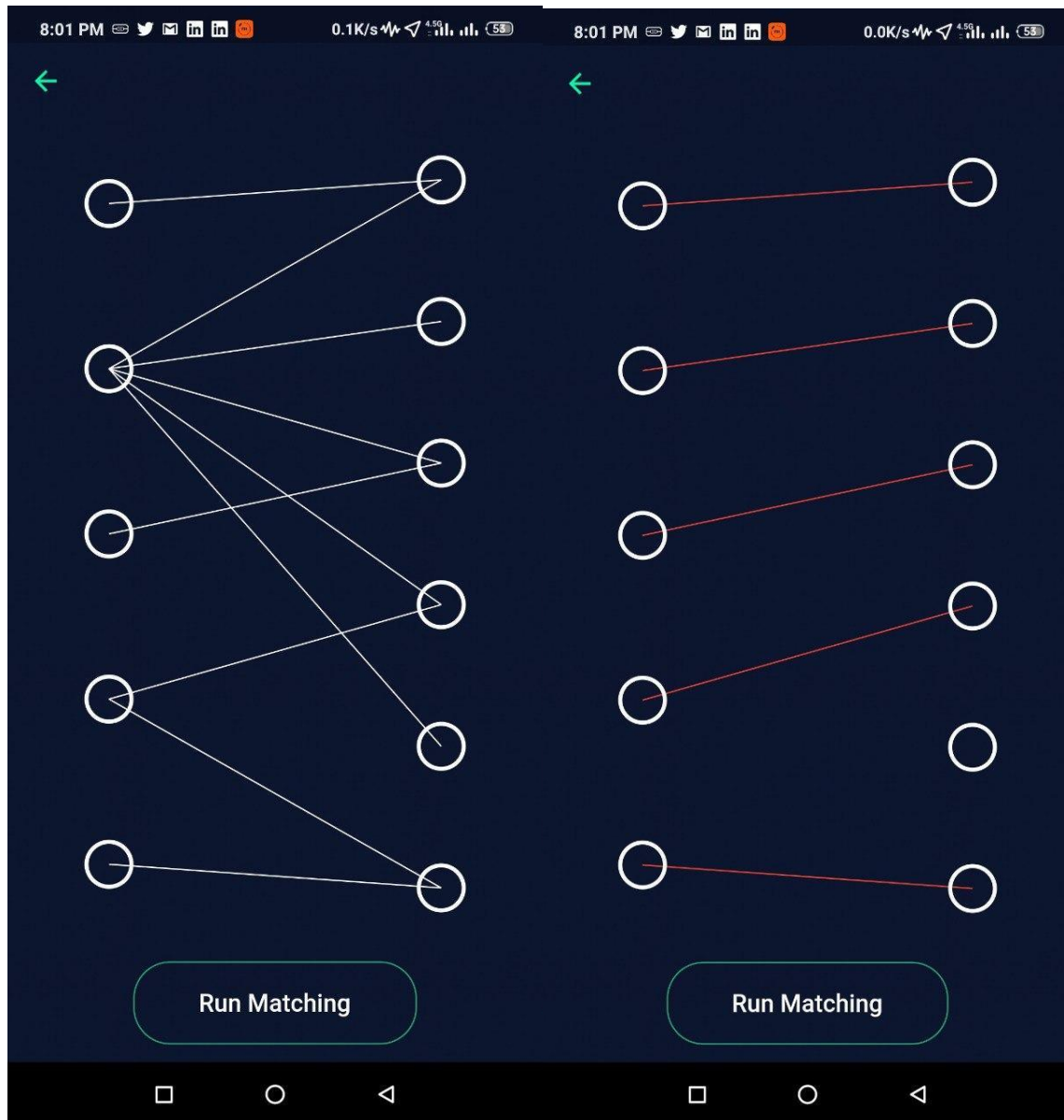
توضیحات این صفحه مانند توضیحات صفحه بالا (kruskal) میباشد. کد این صفحه در آدرس <lib/views/showprim.dart> قابل مشاهده است.

Get Matching Input .v



The screenshot shows a mobile application interface with a dark blue background. At the top, there is a status bar with the time 8:00 PM, social media icons, and network/battery indicators. Below the status bar is a green back arrow icon. The main text in the center reads "Enter Number of vertexes on each side of the page(bipartite graph)". There are two input fields: the first is labeled "Right" and the second is labeled "Left". Below these fields is a rounded rectangular button with the text "DRAW VERTEXES". At the bottom of the screen is a black navigation bar with three white icons: a square, a circle, and a triangle.

کد این صفحه در آدرس `lib/view/getmatchinginput.dart` قابل مشاهده است. کاربر باید یک گراف دوبخشی وارد کند که برای راحتی کاربر، ابتدا تعداد رئوس سمت راست کاربر و سپس تعداد رئوس سمت چپ کاربر در دو تکست باکس دریافت میشود.



در این صفحه الگوریتم مچینگ بر روی گراف ورودی کاربر اجرا میشود. الگوریتم در فایل `lib/matching.dart` قابل مشاهده است. همچنین کد مشاهده و اجرای این الگوریتم در فایل `lib/views/showmatching.dart` قابل مشاهده است. کاربر با کلیک بر روی یکی از دایره ها (رئوس) میتواند آن راس را انتخاب کند و با انتخاب یک راس دیگر، آن دو راس در صورتی که در یک بخش نباشند به هم وصل خواهند شد. با استفاده از دکمه پایین صفحه الگوریتم بر روی گراف ورودی اجرا میشود و نتیجه نمایش داده میشود. کاربر میتواند پس از آن هم با کلیک بر روی رئوس یا ل های دیگر اضافه کند و باز هم با فشردن دکمه پایین نتیجه الگوریتم را مشاهده کند.

کد این پروژه ۲۶۰۰ خط به زبان Dart و با استفاده از فریمورک flutter میباشد. کامنت های لازم در کد برای راحتی درک آن وجود دارد. خروجی ارسال شده برای اندروید است. همچنین کد زده شده توسط اعضای تیم بر روی آدرس:

<https://gitlab.com/amirhoseinizadjou/graphfinalproject>

قابل مشاهده میباشد. این پروژه در صورت لزوم قابل پیشرفت نیز میباشد.