



**زیربرنامه:**

Do\_Delauney(EDGE)

|  |  |  |
| --- | --- | --- |
| **توسعه دهندگان** | مرتضی نامور |  |
| **تهیه کنندگان مستند** | مرتضی نامور | |
| **تاییدکنندگان** | مرتضی نامور | |
| **تاریخ تنظیم سند** | 22/02/1394 | |
| **شناسه سند** | **MC2F003F1** | |
| **زبان برنامه‌نویسی** | **Fortran 90** | |

1. وظایف

این متد به منظور دلانی کردن دو المان غیر دلانی بکار می رود. ورودی متد از نوع EDGE\* می باشد. این متد از هرنظر مشابه متد Flip22 می باشد. تنها تفاوت آن ها در ابتدای این متد است که ابتدا چک می کند اگر دو المان متصل به ضلع ورودی نسبت به یکدیگر دلانی نباشند، آن­گاه عملیات دلانی انجام شود. در حالیکه در Flip22 این شرط چک نمی شود و حتی می توان دو المان دلانی را نسبت به هم غیر دلانی کرد.

1. توضیحات و تئوری­ها

برای دلانی کردن دو مثلث غیر دلانی فقط کافیست ضلع مشترک المان را بین دو گره غیر مشترک قرار دهیم. به شکل زیر دقت کنید :

1. نحوه دلانی کردن دو مثلث غیر دلانی
2. بخش­های زیربرنامه

در این قسمت تمام بخش های زیربرنامه مطابق با شماره گذاری موجود در برنامه کامپیوتری ارائه شده است.

1. پیدا کردن مثلث­های همسایه

در این بخش ابتدا دو المان همسایه ضلع ورودی به متد پیدا شده و سپس در دو متغیر جداگانه ذخیره می شود. لازم بذکر است که اگر ضلع مرزی باشد (یعنی دارای یک همسایه باشد) و یا نوع المان­های همسایه آن بایکدیگر متفاوت باشد، امکان انجام این عملیات وجود ندارد.

1. ذخیره کردن اضلاع المان اول

در این بخش ابتدا آرایه ای تعریف می شود که دارای 5 درایه به منظور ذخیره سازی پنج ضلع دو مثلث می باشد. سپس سه ضلع از مثلث اول درون این آرایه کپی می شود.

1. ذخیره کردن اضلاع مثلث دوم

در این بخش بر روی اضلاع مثلث دوم جستجو می شود و آن اضلاعی که با ضلع ورودی برابر نیستند نیز درون آرایه مذکور قرار می گیرند. لازم بذکر است ضلع ورودی به متد، ضلع مشترک بین دو مثلث بوده و قبلا در آرا ذخیره شده است.

1. مرتب کردن موقعیت ضلع ورودی به متد در درون آرایه

در این بخش با استفاده از یک حلقه بر روی اضلاع حرکت کرده و با استفاده از دستور swap جای ضلع ورودی و ضلع موجود در اولین درایه را جابجا می کنیم. در انتهای این بخش ضلع ورودی به متد (ضلع مترک بین دو مثلث) در درایه اول (با اندیس 0) از آرایه نگهداری اضلاع ذخیره می شود.

1. مرتب سازی مابقی اضلاع موجود در درون آرایه

در این بخش قصد داریم چهار ضلع دیگر را به گونه ای مرتب کنیم که اضلاع قرار گرفته در درایه های دوم و سوم آرایه مذکور (درایه های با اندیس 1 و 2) اضلاعی باشند که به گره اول از ضلع مشترک متصل باشند. طبیعتا پس از مرتب سازی این دو ضلع ، دو ضلع دیگر باقیمانده در درایه های چهارم و پنجم ، اضلاعی هستند که به گره دوم از ضلع ورودی متصل هستند. کل 2 این اضلاع را نشان می دهد:

E

E->P[0]

E->P[1]

1. اضلاع دو مثلث غیر دلانی

با توجه به شکل 2 و توضیحات ارائه شده، مشخص است که ضلع قرمز رنگ (ضلع مشترک) در اولین درایه از آرایه ذخیره شده است. سپس اضلاع سبز رنگ در درایه های دوم و سوم ذخیره شده اند. در انتها نیز اضلاع آبی رنگ در درایه های چهارم و پنجم ذخیره می شوند.

1. تغییر گره­های دو سر ضلع ورودی (چرخاندن ضلع مشترک)

در این بخش قصد داریم ضلع ورودی را جابجا کرده و گره های غیر مشترک موجود در دو مثلث کنونی را به عنوان گره­های دو سر ضلع مشترک معرفی کنیم. بدین منظور بررسی می کنیم اگر گره اول از ضلع موجود در درایه دوم آرایه (درایه با اندیس 1) با گره اول ضلع ورودی یکی نباشد (یعنی گره اول از این ضلع جزء گره های مشترک کنونی دو مثلث نباشد) آن­گاه گره اول از ضلع مشترک را تغییر داده و برابر با گره اول ضلع موجود در درایه دوم قرار می دهیم. در غیر اینصورت (یعنی در حالتی که گره اول ضلع موجود در درایه دوم با گره اول ضلع مشترک برابر باشد) آن­گاه گره اول ضلع مشترک را برابر با گره دوم ضلع موجود در درایه دوم (یکی از دو ضلع سبز رنگ) قرار می دهیم. مشابه این کار را برای گره دیگر ضلع مشترک نیز انجام می دهیم ولی این بار این مقایسه را با ضلع سبز رنگ دیگر یعنی ضلع موجود در درایه سوم انجام می دهیم. این مطلب در شکل 3 نشان داده شده است.

E

E->P[1]

edge[1]

edge[2]

E->P[0]

E->P[1]

Edge[3]

Edge[4]

E

E->P[0]

E->P[1]

edge[1]

edge[2]

Edge[3]

Edge[4]

C[0]

C[0]

C[1]

C[1]

1. تغییر گره های متصل به ضلع مشترک و جابجایی آن
2. تغییراضلاع المان­های متصل به ضلع مشترک

در این قسمت با توجه به شکل 3 و ذکر این نکته که در حال حاضر گره های ضلع مشترک عوض شده­اند، بایستی اضلاع مرتبط با دو المان سابق نیز عوض شوند. به طور مثال در حالت قبل (قبل از انجام عملیات) المان اول دارای اضلاع موجود در درایه های اول ، دوم و چهارم (درایه های با اندیس 0 ، 1و3) بود (یعنی دارای ضلع قرمز رنگ ، ضلع سبز رنگ بالا و ضلع آبی رنگ بالا بود) اما در حال حاضر باید این اضلاع تغییر کنند و ضلع قرمز جدید (ضلع مشترک کنونی) ، ضلع سبز رنگ بالا و ضلع سبز رنگ پایین جایگزین اضلاع قدیمی شوند. مشابه همین کار برای المان دیگر انجام می شود و اضلاع آن به ضلع قرمز جدید و دو ضلع آبی تغییر می کند.

1. حذف آدرس ضلع مشترک سابق از لیست اضلاع متصل به گره اول آن

با توجه به اینکه گره های دو سر ضلع مشترک قدیمی عوض شده اند اما آدرس این ضلع در ساختار داده­ای عوض نشده است ، لذا هنوز آدرس این ضلع در لیست اضلاع متصل به اولین گره قدیمی آن وجود دارد. در این بخش قصد داریم این آدرس را از لیست مذکور حذف کنیم که هیچگونه ارتباطی بین ضلع مشترک جدید و گره های قدیمی آن وجود نداشته باشد.

1. حذف آدرس ضلع مشترک سابق از لیست اضلاع متصل به گره دوم آن

مشابه عملیات انجام گرفته در بخش 8 را در این بخش برای گره دوم آن ضلع انجام می دهیم.

1. اضافه کردن آدرس ضلع مشترک جدید به لیست اضلاع متصل به گره­های آن

در این بخش قصد داریم آدرس ضلع مشترک جدید که همان آدرس قبلی است را به لیست اضلاع متصل به دو گره جدید آن اضافه کنیم.

1. حذف آدرس المان دوم از لیست المان­های متصل به گره اول ضلع مشترک سابق

مشابه بخش 8 در این بخش المانی که در حالت قبل به گره اول ضلع مشترک متصل بود و در حالت جدید دیگر به آن گره متصل نیست را باید از لیست المان­های متصل به آن گره حذف کنیم. با توجه به شکل 3 المان دوم یعنی C[1] باید از لیست المان های متصل به گره اول ضلع مشترک سابق حذف شود.

1. حذف آدرس المان اول از لیست المان­های متصل به گره دوم ضلع مشترک سابق

مشابه بخش 11 در این بخش آدرس المان C[0] را از لیست المان ­های متصل به گره دوم ضلع مشترک سابق حذف می کنیم.

1. اضافه کردن آدرس دو المان به گره­های ضلع مشترک جدید

در این بخش نیز آدرس دو المان C[0] و C[1] به لیست المان­های متصل به گره­های ضلع مشترک جدید اضافه می شود.