



**زیربرنامه:**

Flip24(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 یکی از متد­های اختصاصی کلاس MESH می باشد که کاربران به آن دسترسی ندارند و فقط توسط متدهای داخلی کلاس MESH امکان فراخوانی آن وجود دارد. وظیفه اصلی این متد اضافه کردن یک گره بر روی وسط یک ضلع رابط و تقسیم المان مثلثی آن به دو المان مثلثی است. استفاده از این متد صرفاً مربوط به متدهایی است که بر روی لایه مرزی و شبکه هیبرید کار می کنند.

1. توضیحات و تئوری­ها

از نظر تئوری نحوه عملکرد این متد کاملاً مشابه متد Flip24(EDGE) در حالتی که ضلع ورودی مرزی است می باشد. تنها تفاوت آن این است که ورودی این المان یک ضلع از لایه رابط شبکه هندسی خواهد بود.نحوه عملکرد این متد بدین صورت است که ابتدا بررسی می شود که ضلع ورودی ، ضلع لایه رابط است یا خیر. اگر ضلع لایه رابط باشد ، مختصات وسط ضلع را پیدا کرده و به عنوان گره جدید به شبکه اضافه می کند. سه ضلع بین گره جدید و سه گره قبلی المان تشکیل می شود که در واقع ضلع ورودی به دو ضلع تقسیم شده و ضلع دیگر نیز به این گره متصل می شود. درنهایت دو ضلع جدید به شبکه اضافه می شود و ضلع ورودی نیز اصلاح می شود. در آخر هم المان مثلثی که ضلع ورودی در آن قرار داشته اصلاح می شود و یک المان دیگر به شبکه اضافه می شود.

Ep

Ep

1. اعمال متد flip24 بر روی ضلع Ep در حالت ضلع لایه رابط
2. بخش­های زیربرنامه
3. تعریف متغیر­های اولیه

در این بخش سه متغیر از نوع آدرس تعریف می شود که به ترتیب آدرس اولین خانه حافظه اشغال شده توسط وکتور­های مربوط به المان­ها، اضلاع و گره­ها می باشد.

1. پیدا کردن المان­های متصل به ضلع ورودی و مرتب کردن آن ها

در این بخش به کمک دستور find المان­های متصل به این ضلع را پیدا کرده و در یک آرایه ذخیره می کنیم. نکته حائز اهمیت آن است که با توجه به اینکه ضلع ورودی یک ضلع از لایه رابط است لذا دو المان متصل به آن از نظر تعداد گره با یکدیگر متفاوت خواهند بود. به همین علت در ادامه این دو المان درون آرایه مورد نظر به گونه ای مرتب می شوند که المان مثلثی در اولین درایه آن المان قرار بگیرد.

1. پیدا کردن و مرتب کردن گره­های المان

در این بخش ابتدا به کمک متد Find گره های المان مثلثی مصل به ضلع رابط پیدا می شوند و سپس این گره ها به گونه ای مرتب می شود که اولین گره قرار گرفته در آرایه Point (گره با اندیس 0 ) گره­ای باشد که بر روی ضلع ورودی متد واقع نشده است.

1. ساخت گره جدید بر روی مرکز ضلع ورودی

در این بخش ابتدا با استفاده از متد mid مختصات مرکز ضلع ورودی محاسبه می شود. سپس یک درایه به وکتور گره­های شبکه هندسی اضافه شده و عدد گره­ها یک واحد افزایش می یابد. در نهایت مختصات بدست آمده برای مرکز ضلع ورودی به عنوان مشخصات گره جدید به آن اضافه می شود. در انتها نیز با توجه به اینکه در این حالت ضلع ورودی مرزی است لذا گره جدید نیز بر روی مرز واقع شده و این مشخصه نیز برای گره فعال می شود.

1. پیدا کردن عدد ناحیه المان متصل به ضلع ورودی

بدون توضیح

1. اصلاح ضلع ورودی و تشکیل دو ضلع جدید

در این بخش نوبت به ساخت اضلاع و اصلاح ضلع ورودی می باشد.

ابتدا به شکل زیر دقت کنید :

Ep(old)

P[0]

P[1]

Point[0]

P[0]

P[1]

Point[0]

Ep(new)

Ep(old).P[0]

1. نحوه شماره گذاری گره ها

در شکل سمت چپ و قبل از اعمال متد شماره گره ها مشخص است. گره سمت راست، گره P[0] از ضلع Ep و گره سمت چپ هم گره P[1] از ضلع Ep می باشد. گره ای هم که بر روی ضلع Ep واقع نشده همانطور که توضیح داده شد اکنون Point[0] می باشد.

پس از اعمال متد و در شکل سمت راست مشخص است که گره جدید را به عنوان گره P[0] از ضلع Ep قرار می دهیم. گره P[1] ضلع Ep اصلاح شده تغییری نمی کند. در مورد دو ضلع جدید هم اینگونه عمل می کنیم که گره با اندیس 0 هر دو ضلع جدید ، همین گره جدید می باشد. ضلع اول ساخته شده ضلع افقی سمت راست است. یعنی گره با اندیس 1 ضلع اول ساخته شده ما همان گره P[0] از ضلع Ep سابق می باشد. برای ضلع دوم نیز گره با اندیس 1 آن برابر Point[0] خواهد بود.

1. افزایش تعداد المان­های شبکه هندسی

بدون توضیح

1. اصلاح المان اولیه و ساخت المان جدید با یک ضلع

در این بخش ابتدا یک ضلع تقلبی می سازیم . این ضلع بین گره های P[0] و Point[0] در المان اولیه ساخته می شود . در شکل زیر تغییرات کاملا مشخص هستند :

P[0]

P[1]

Point[0]

Ep(new)

Ep(old).P[0]

Ep(old)

P[0]

P[1]

Point[0]

c\_a[NC-1]

C[0](new)

C[0]

e\_a[NF-2]

e\_a[NF-1]

E\_fake

1. المان بندی جدید پس از اعمال متد

برای اصلاح المان C[0] همانطور که در شکل فوق مشخص است کافیست بجای ضلعی که برابر E\_fake می باشد دومین ضلع ساخته شده یعنی e\_a[NF-1] را قرار دهیم. المان جدید هم با استفاده از ضلعی که برابر E\_fake است ساخته می شود.

1. اختصاص دو ضلع دیگر المان جدید

در این بخش دو ضلع دیگر المان جدید با توجه به شکل 4 همان دو ضلع جدید ساخته شده هستند. در انتها عدد حوزه المان جدید نیز باید مشخص شود.

1. اضافه کردن اضلاع متصل به گره جدید به لیست مربوطه

با توجه به اینکه سه ضلع (دو ضلع جدید و یک ضلع قدیمی) به گره جدید ساخته شده متصل هستند بایستی این اضلاع به کمک دستور PointAddEdg به لیست اضلاع متصل به این گره جدید وارد شوند.

1. حذف ضلع ورودی اولیه از لیست اضلاع متصل به یکی از گره های آن

در این بخش با توجه به اینکه گره اول از ضلع ورودی اولیه ( گره با اندیس 0 ) در ضلع اصلاح شده تغییر کرده و دیگر این ضلع به آن گره متصل نیست لذا این ضلع بایستی از لیست اضلاع متصل به آن گره حذف شود.

1. اضافه کردن اضلاع جدید به لیست اضلاع متصل به گره های مربوطه

در این بخش باید اولین ضلع جدید ساخته شده را به لیست اضلاع متصل به گره اول از ضلع ورودی اولیه اضافه کنیم (در شکل 4 مشخص است) . همچنین دومین ضلع ساخته شده را باید به لیست اضلاع متصل به گره­ای که بر روی ضلع ورودی واقع نشده بود، اضافه کنیم.

1. اضافه کردن المان­ها به لیست المان­های متصل به گره­های مربوطه

در این بخش در مورد المان ها ابتدا از گره جدید شروع می کنیم. المان اولیه و المان جدید ساخته شده باید به لیست المان­های متصل به این گره اضافه شوند. همچنین به گره­ای که بر روی ضلع ورودی واقع نشده بود در اینجا فقط باید المان جدید به لیست آن اضافه شود زیرا المان اصلاح شده قبلاً هم در لیست المان­های متصل به این گره قرار داشته است.

1. حذف المان اولیه از لیست المان­های متصل به گره اول از ضلع ورودی

در این بخش با توجه به اینکه المان اصلاح شده دیگر به این گره متصل نیست لذا بایستی از لیست المان­های متصل به آن گره حذف شود. این کار به کمک دستور PointRemCell اجرا می شود.

1. اضافه کردن المان جدید به لیست المان­های متصل به گره اول ضلع ورودی

بدون توضیح