



**زیربرنامه:**

Add(Surface,int)

|  |  |  |
| --- | --- | --- |
| **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 اضافه کردن یک لایه جدید از اضلاع و المان ها به یک شبکه هندسی که دارای حوزه هایی از نوع hybrid می باشد ، تعریف شده است. این متد دو ورودی دارد . یک شئ (Object) از نوع surface که حاوی اضلاع و المان های یک لایه است ، و یک عدد از نوع int است. از آنجایی که یک لایه از اضلاع ، دارای دو ردیف المان در طرفین خود می باشد ، ورودی دوم مشخص می کند که قصد اضافه کردن لایه در کدام طرف لایه ورودی را داریم.

1. توضیحات و تئوری­ها

با توجه به توضیحاتی که قبلاً در ارتباط با بحث لایه ها ارائه شده است ، می دانیم که یک لایه ممکن است از دو انتها منتهی به اضلاع روی لایه رابط باشد ، یا از یک طرف منتهی به لایه رابط و از طرف دیگر منتهی به مرز داخلی یا خارجی باشد و حالت دیگر هم نیز زمانی است که ابتدا و انتهای لایه به یکدیگر متصل است مانند یک حلقه. این متد بایستی بتواند تمامی حالت ها را در بر بگیرد و با توجه به نوع لایه ورودی عملیات مناسب اضافه کردن لایه جدید را انجام دهد.

در این بخش بر خلاف رویه مرسوم در ارتباط با توضیح متد­ها ، مباحث مربوط به چگونگی اضافه کردن لایه ها را در ذیل بخش­های برنامه توضیح خواهیم داد.

1. بخش­های زیربرنامه
2. تعریف متغیر­های اولیه

بدون توضیح

1. ذخیره اولین و آخرین المان لایه

در این بخش اولین و آخرین المان لایه در دو متغیر جداگانه ذخیره می شود. همچنین آرایه ای برای ذخیره اضلاع روی لایه رابط که درون این دو المان قرار دارند، تعریف شده است.

شکل زیر یک لایه از اضلاع و المان ها را نشان می دهد :

1. یک لایه از اضلاع و المان ها

با توجه به اینکه در این متد ما باید یک ردیف از المان ها را انتخاب کنیم شکل 1 تغییر می کند و به شکل زیر در می آید :

1. یک لایه از اضلاع بهمراه یک ردیف از المان ها

در شکل فوق اضلاع قرمز رنگ ، اضلاع لایه می باشند. ضلع آبی رنگ ، ضلع مربوط به لایه رابط است. ضلع سبز رنگ ضلع مرزی است. المان زرد رنگ اولین المان لایه و المان آبی رنگ آخرین المان لایه می باشد(در حالتی که لایه از دو طرف به لایه رابط منتهی می شود مهم نیست که کدام یک المان اول و کدام یک المان آخر است).

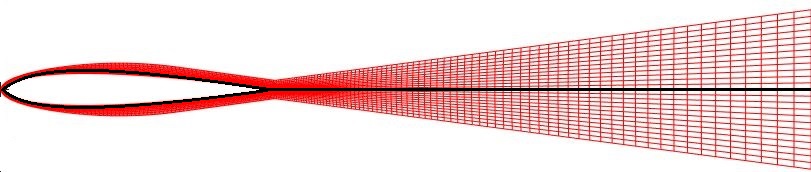
در مورد آرایه ی ذخیره سازی اضلاع لایه رابط ، در ابتدا هر چهار درایه آن را برابر با nulltpr قرار گرفته است. به طور معمول اگر لایه از دو طرف به لایه رابط منتهی شود ، المان اول و آخر که با لایه رابط در ارتباطند هر کدام فقط دارای یک ضلع روی لایه رابط هستند پس در این حالت فقط دو درایه از این آرایه پر می شود. در حالتی که لایه ورودی از یک طرف منتهی به لایه رابط و از طرف دیگر منتهی به مرز باشد ، فقط یک المان اول و یا آخر آن یک ضلع بر روی لایه رابط دارد و در این صورت فقط یک درایه از این آرایه پر می شود. اما یک حالت استثنا هم وجود دارد که در آن حالت هر چهار درایه آرایه مذکور پر می شوند. این حالت استثنا مربوط به زمانی است که کاربر دقیقاً اولین لایه ی زیر لایه رابط را به عنوان ورودی به متد بدهد و از طرفی شماره ردیف المانی که انتخاب می کند آن ردیفی باشد که المان های آن بین لایه رابط و لایه ورودی باشد. در این صورت المان اول و المان آخر هر کدام دارای دو ضلع بر روی لایه رابط خواهند بود که در این حالت هر چهار درایه پر می شود.

در انتهای این بخش سه پارامتر از نوع int تعریف شده است. پارامتر اول تعداد اضلاع مرزی المان اول و آخر را می شمارد. پارامتر دوم تعداد اضلاع المان اول و آخر را که بر روی لایه رابط قرار دارند، می شمارد. در نهایت پارامتر سوم مشخص می کند که آیا لایه یک لایه حلقوی است یا خیر . اگر 0 باشد به معنی لایه معمولی و اگر 1 باشد به معنی لایه حلقوی است. مقدار اولیه این پارامتر 0 در نظر گرفته شده است.

1. تشخیص حلقوی بودن لایه

در این بخش ابتدا المان های همسایه اولین المان لایه را به کمک دستور find می یابیم. سپس با نوشتن یک حلقه بر روی المان های همسایه حرکت می کنیم . اگر در المان های همسایه، المانی وجود داشت که با المان آخر برابر بود (یعنی المان اول و آخر همسایه بودند) آن گاه یک حلقه for دیگر با شمارنده j تعریف می کنیم که از 0 تا کمتر از 4 حرکت کند (در واقع می خواهیم بر روی اضلاع المان های مربعی حرکت کنیم ) . درون این حلقه این شرط را بررسی می کنیم که آیا ضلع j ام از المان دوم لایه (C[1]🡪E[j]) درون المان یکی مانده به آخر لایه (C[C.size()-2] ) قرار دارد یا خیر. اگر قرار ندارد آن گاه مقدار متغیر o را برابر با 1 قرار می دهیم. و این بدان معنیست که لایه یک لایه حلقوی می باشد.

در واقع در این مجموعه دستورات این مفهوم را بررسی کردیم که آیا المان اول و آخر لایه به هم متصل هستند یا خیر . اگر هستند آیا المان دوم و المان یکی مانده به آخر لایه نیز به هم متصل هستند یا خیر. اگر المان اول و آخر به هم متصل باشند و المان دوم و المان ماقبل آخر به هم متصل نباشند این بدان معنی است که لایه حلقوی است. اما یک حالت وجود دارد که در آن هم المان های اول و آخر به هم متصل باشند و هم تعداد دیگری از المان ها . شکل زیر این نمونه لایه را نشان می دهد. البته این لایه یک لایه معمولی است :



1. یک لایه خاص که المان های دو طرف اضلاع لایه متعلق به یک ردیف از المان های لایه می باشند.

در شکل فوق خط سیاه رنگ اضلاع یک لایه را نشان می دهد که المان اول لایه در بالای ضلع و المان آخر لایه در پایین ضلع واقع شده است. همچنین المان دوم لایه در بالای ضلع و المان یکی مانده به آخر در پایین ضلع. در این شکل فقط همین لایه است که دارای این خاصیت است . این لایه دقیقا لایه وسط می باشد.

1. شمارش تعداد اضلاع مرزی دو المان اول و آخر

در این بخش وضعیت اضلاع هر دو المان اول و آخر را بررسی می کنیم. در صورتی که خاصیت State هر یک از اضلاع برابر با 0 باشد (آن ضلع مرزی باشد) آن گاه یک واحد به مقدار پارامتر مربوطه اضافه می شود.

1. شمارش تعداد اضلاع متصل به لایه رابط

مشابه فرآیند بخش قبل را این بار برای بررسی اضلاع روی لایه رابط انجام می دهیم. در این بخش اگر خاصیت Is\_Interface هر ضلع برابر با 1 باشد (ضلع روی لایه رابط باشد) آن گاه هم به مقدار پارامتر مربوط به شمارش این اضلاع یک واحد اضافه می شود و هم این ضلع درون آرایه تعریف شده در بخش دوم ذخیره می شود.

1. شروع فرآیند اضافه کردن یک لایه جدید برای حالتی که لایه از دو طرف به لایه رابط متصل باشد

در این بخش ابتدا برسی می شود که اگر مقدار پارامتر شمارنده اضلاع لایه رابط برابر با 2 و مقدار پارامتر تعیین کننده حلقوی بودن لایه برابر 0 باشد آن گاه وارد مابقی دستورات این بخش بشویم. در این حالت لایه ورودی به شکل زیر است :

1. لایه ای که از دو انتها به لایه رابط متصل است

در ادامه این بخش دو متغیر برای ذخیره سازی گره­های اول اضلاع موجود در آرایه ذخیره سازی اضلاع تعریف می شود. به عنوان قرارداد می خواهیم که گره های دوم (گره با اندیس 1 ) از اضلاع موجود در آرایه ، گره هایی باشند که به اضلاع روی لایه متصل هستند . این موضوع در شکل زیر مشخص است :

IE[0]

IE[1]

IE[0].P[1]

IE[1].P[1]

IE[0].P[0]

IE[1].P[0]

S.E[0]

S.E[C.size()-1]

1. شماره گذاری گره ها و اضلاع

در انتها گره های اول دو ضلع روی لایه رابط درون متغیر­های تعریف شده در ابتدای این بخش ذخیره می شود. دراینجا مطمئن خواهیم بود که این دو گره بر روی لایه ورودی واقع نشده­اند.

1. اصلاح المان بندی بیرون لایه ( المان های درون حوزه simple)

برای اضافه کردن یک لایه جدید به شکل شماره 5 بایستی اضلاع عمودی لایه را از وسط نصف کرده و اضلاعی موازی با اضلاع ورودی لایه بین آن ها تشکیل دهیم. اگر به دو ضلع آبی رنگ در شکل 5 توجه شود مشخص می شود که با نصف کردن آن­ها بایستی عملیاتی نیز در خارج لایه و بر روی المان­های مثلثی متصل به این اضلاع صورت پذیرد. این عملیات مطابق عملیات انجام شده در متد Flip24 برای حالتی که ضلع مرزی باشد خواهد بود با این تفاوت که اینجا ضلع مرزی نیست و روی لایه رابط است. متد I\_add یک متد اختصاصی از کلاس Mesh می باشد که بدین منظور تعریف شده است. در این بخش این متد را بر روی دو ضلع ذخیره شده در آرایه (اضلاع آبی رنگ لایه رابط) اعمال می کنیم.

1. ایجاد فضای لازم در حافظه برای ساخت گره بر روی اضلاع عمودی

اکنون می خواهیم المان های موجود در این لایه را از وسط نصف کنیم به طوریکه یک لایه کاملا موازی با لایه اولیه تشکیل دهیم. این بدان معناست که باید اضلاع عمودی در شکل 5 را نصف کرده و در مرکز هر کدام یک گره قرار دهیم . سپس هر گره را به گره مقابل بر روی ضلع مقابل خود متصل کنیم. با این حساب تعداد گره های ساخته شده یکی بیشتر از تعداد المان های موجود می باشد. البته ذکر این نکته لازم است که در اینجا دو گره قبلاً توسط متد I\_add ساخته شده است که در حال حاضر آخرین گره های اضافه شده به شبکه هندسی است. از طرفی در این بخش ابتدا یک آرایه از نوع bPOINT که صرفا مختصات گره­ها را در خود ذخیره می کند به تعداد یک واحد بیشتر از المان های لایه می سازیم. سپس درایه اول و درایه آخر این آرایه را برابر با مختصات گره های ساخته شده توسط متد I\_add که هم اکنون دو گره آخر شبکه هندسی می باشد قرار می دهیم.

1. پیدا کردن مختصات وسط هر ضلع عمودی لایه و اضافه کردن آن مختصات به آرایه تشکیل شده در بخش 8

پس از تشکیل آرایه مختصات مرکز اضلاع عمودی در بخش قبل، اکنون بایستی مختصات مرکز اضلاع عمودی را پیدا کرده و در این آرایه قرار دهیم. بدین منظور ابتدا یک متغیر واسط تعریف کرده و آن را برابر با اولین ضلع ذخیره شده در آرایه مربوط به اضلاع روی لایه رابط قرار می دهیم. سپس با حرکت بر روی المان­های لایه ورودی به کمک دستور across ضلع مقابل (موازی) با ضلع لایه رابط را در المان کناری پیدا می کنیم و آن را در متغیر واسط ذخیره می کنیم. سپس مطابق بخش 6 گره های این ضلع را به گونه ای مرتب می کنیم که گره دوم آن بر روی لایه ورودی باشد و گره اول آن بر روی لایه ورودی قرار نگیرد. در انتها نیز مختصات مرکز این ضلع را به آرایه تشکیل شده در بخش 8 اضافه می کنیم. این کار به طور پیوسته انجام می گیرد و به ترتیب تمام اضلاع عمودی لایه را نصف کرده و مختصات آن ها را به آرایه اضافه می کند.

1. افزایش تعداد گره های شبکه هندسی

بدون توضیح

1. افزایش تعداد اضلاع شبکه هندسی

بدون توضیح

1. ذخیره تعداد اضلاع و تعداد گره­های شبکه هندسی تا این بخش برنامه

در این بخش دو متغیر تعریف می کنیم و تعداد کنونی اضلاع و گره­های شبکه را در آن دو ذخیره می کنیم. در ادامه از این دو متغیر به منظور فراخوانی آخرین گره­ها و اضلاع اضافه شده به شبکه استفاده می شود. در واقع این دو متغیر به صورت Check Point عمل می کنند.

1. ساخت اضلاع افقی و اضافه کردن آن ها به شبکه هندسی به همراه گره های مرکزی

در این بخش با توجه به اینکه در بخش های قبل آرایه از نوع bPOINT ساخته ایم و مختصات مرکز اضلاع عمودی لایه را در آن قرار داده ایم . حال بایستی مختصات درون این آرایه را به آرایه مربوط به گره های شبکه هندسی که از نوع POINT می باشد اضافه کنیم. همچنین بین هر دو گره مجاور یک ضلع می سازیم که این ضلع افقی خواهد بود و آن را نیز به آرایه مربوط به اضلاع شبکه هندسی اضافه می کنیم. در هر بار اجرای حلقه نیز یک واحد به تعداد گره­ها و یک واحد به تعداد اضلاع شبکه هندسی اضافه می کنیم. توجه شود چون دو گره قبلا به شبکه تویط متد I\_add اضافه شده اند لذا شمارنده حلقه از درایه دوم تا درایه یکی مانده با آخر آرایه تعریف شده از نوع bPOINT حرکت می کند. با این حساب آخرین ضلع افقی که باید بین گره روی یکی مانده به آخرین ضلع عمودی و ضلع عمودی روی لایه رابط تشکیل شود، تشکیل نخواهد شد. این کار در بخش بعدی انجام می شود.

1. ساخت آخرین ضلع عمودی

همانطور که گفته شد در بخش قبل آخرین ضلع افقی ساخته نشد. این ضلع در این بخش بین آخرین گره ساخته شده در شبکه هندسی و یکی مانده به آخرین گره­ی که پس اجرای مرحله 7 ساخته شده و به شبکه اضافه شده است تشکیل می شود. در واقع این گره بر روی ضلع دوم از آرایه ذخیره اضلاع لایه رابط (اضلاع آبی رنگ) قرار دارد. در انتها یک واحد به تعداد اضلاع اضافه می شود.

1. ذخیره تعداد اضلاع شبکه هندسی تا این مرحله و افزایش فضای وکتور اضلاع شبکه هندسی

بدون توضیح

1. ساخت اضلاع عمودی جدید و اصلاح اضلاع عمودی سابق

با توجه به اینکه با ساخت گره های جدید در وسط اضلاع عمودی سابق ، دو ضلع در آنجا تشکیل می شود می توان اینگونه حساب کرد که یک ضلع جدید ساخته می شود و ضلع قدمی فقط باید اصلاح شود و مختصات یکی از گره ­های آن تغییر کند. در این بخش مطابق بخش 9 ابتدا یک متغیر واسط تعریف می کنیم و آن را برابر با اولین ضلع لایه رابط قرار می دهیم. سپس با حرکت بر روی المان های لایه ، ضلع مقابل به ضلع عمودی لایه رابط را پیدا می کنیم. آن گاه برای ساخت ضلع عمودی جدید، گره اول آن را برابر با گره جدید ساخته شده قرار می دهیم. گره دوم آن را برابر با گرهی از ضلع عمودی سابق که بر روی لایه ورودی واقع نشده بود (یعنی گره با اندیس 0 ضلع عمودی سابق) قرار می دهیم. در اینجا ضلع عمودی جدید ساخته شد. حال برای اصلاح ضلع عمودی سابق فقط کافیست گره با اندیس 0 آن ضلع را برابر با گره جدید ساخته شده قرار دهیم. بدین صورت این ضلع قدیمی نیز اصلاح می شود.

شکل 6 این نکات را نشان می دهد :

IE[0].P[0]

e\_a[NF+1]

e\_a[NF]

P\_a[check\_P]

P\_a[check\_P-1]

lE=across(IE[0])

lE=across(IE[0])

IE[0]

lE.P[1]

lE.P[0]

IE[0].P[1]

IE[0]

S.E[0]

1. نام گذاری گره ها و اضلاع
2. ذخیره تعداد المان­های شبکه هندسی تا این مرحله و افزایش فضای وکتور المان­های شبکه هندسی

بدون توضیح

1. ساخت و اصلاح اولین المان لایه

در این بخش اولین المان لایه را اصلاح کرده و اولین المان جدید را نیز با اضلاع جدید آن می سازیم.

across(S.E[0])

across(S.E[0])

e\_a[check\_E1]

e\_a[check\_E2]

e\_a[check\_E1-2]

lE=across(IE[0])

IE[0]

S.E[0]

C[NC]

C[0]

lE=across(IE[0])

IE[0]

S.E[0]

C[0]

1. المان های جدید و اضلاع ان ها

با توجه به شکل 7 مشخص است که برای اصلاح المان سابق کافیست به جای آن ضلعی که مقابل ضلع لایه ورودی است، اولین ضلع افقی جدید ساخته شده را قرار دهیم. مابقی اضلاع این المان تغییر نمی کنند.

همچنین با توجه به شکل 7 ، اولین المان جدید بین این چهار ضلع تشکیل می شود:

1. ضلع مقابل به ضلع لایه ورودی
2. اولین ضلع جدید افقی ساخته شده
3. اولین ضلع جدید عمودی ساخته شده
4. دومین ضلع جدید عمودی ساخته شده

در این بخش به کمک دستور CellAddEdg عملیات اضافه کردن اضلاع به المان جدید صورت می پذیرد. در انتها یک واحد به تعداد المان ها اضافه می شود.

1. ساخت و اصلاح دیگر المان­های لایه

این بخش مشابه بخش قبل است با این تفاوت که این عملیات درون یک حلقه صورت گرفته و مابقی المان­ها در این بخش ساخته و اصلاح می شوند. البته آخرین المان لایه نیز باید به صورت جداگانه اصلاح شود.

1. ساخت و اصلاح آخرین المان لایه

این بخش مشابه بخش 18 است با این تفاوت که شماره اضلاعی که باید در المان­ها قرار بگیرد متفاوت است.

1. اضافه کردن المان­های ساخته شده به دو گره روی لایه رابط

پس از ساخت گره ها ، اضلاع و المان ها و اضافه کردن آن ها به شبکه اکنون باید اطلاعات لیست اضلاع و المان های متصل به گره های ساخته شده و گره های درگیر در لایه را تکمیل کنیم . این کار را از گره های P1 و P2 که در ابتدای متد آن ها را مشخص کردیم شروع می کنیم. این نکته یادآوری می شود که این دو گره در سمت لایه ورودی نبوده اند لذا اولین المان ساخته شده باید به P1 و آخرین المان ساخته شده باید به P2 اضافه شوند. همچنین اولین المان اصلاح شده باید از P1 و آخرین المان اصلاح شده باید از P2 کم شوند

1. حذف اولین المان قدیمی از لیست المان­های متصل به گره P1

بدون توضیح

1. حذف آخرین المان قدیمی از لیست المان­های متصل به گره P2

بدون توضیح

1. اضافه کردن اولین و آخرین ضلع عمودی جدید به لیست اضلاع متصل به گره های جدید در مرکز آن ها

اکنون به سراغ گره های جدید ساخته شده در مرکز اضلاع عمودی می رویم و اضلاع و المان های متصل به آن ها را مشخص می کنیم . ابتدا از دو گره اولی که توسط متد I\_add بر روی اضلاع لایه رابط (اضلاع آبی رنگ) ساخته شده است شروع می کنیم. به این نکته توجه شود که شماره درایه گره روی اولین ضلع برابر با Check\_P-1 و شماره درایه گره روی آخرین ضلع برابر با Check\_P-2می باشد.

1. اضافه کردن اولین و آخرین المان ساخته شده و اصلاح شده به لیست المان­های متصل به اولین و آخرین گره جدید ساخته شده

بدون توضیح

1. اضافه کردن اضلاع جدید به لیست اضلاع متصل به گره­های جدید ساخته شده

در ابتدا ذکر این نکته لازم است که مطالب این بخش تا بخش 30 کاملاً به هم پیوسته است لذا ابتدا توضیحاتی در این رابطه بیان می شود و سپس به توضیح این چند بخش خواهیم پرداخت:

ابتدا باید اضلاع و المان های متصل به گره های روی اضلاع عمودی ساخته شده را اصلاح کنیم . به این نکته توجه کنید که گره های ساخته شده به عنوان گره با اندیس 0 از اضلاع عمودی جدید قرار می گیرند. اما گره هایی که با اندیس 1 این اضلاع جدید که از قبل در شبکه موجود بوده اند نیز باید اطلاعات ان ها اصلاح شود. به شکل زیر توجه کنید:

e\_a[check\_E2+i].P[1]

e\_a[check\_E2+i]

S.E[i]

p\_a[check\_P+i]

S.E[i+1]

e\_a[check\_E1+i+1]

e\_a[check\_E1+i]

**c\_a[check\_C+i+1]**

**c\_a[check\_C+i]**

C[i+1]

C[i]

یس

lE

1. نحوه شماره گذاری گره ها ، اضلاع و المان ها

با توجه به شکل فوق به گره قرمز رنگ 4 ضلع باید متصل شود. این چهار ضلع ، دو ضلع سبز رنگ ، ضلع ابی رنگ و ضلع مشکی رنگ می باشد. همچنین به این گره باید 4 المان متصل شود که شامل دو المان زرد رنگ و دو المان بی رنگ است. در مورد گره مشکی رنگ باید به این نکته توجه شود که قبل از اضافه کردن لایه جدید ، این گره متصل به دو المان بی رنگ بود. لذا اکنون آن دو المان باید از لیست المان های متصل به این گره حذف شوند. از طرفی دو المان جدید ساخته شده زرد رنگ باید به این لیست اضافه شوند. همچنین این گره ، قبل از اضافه کردن لایه جدید ، یکی از گره های ضلع مشکی عمودی پایین یعنی (lE) بود اما اکنون یکی از گره های ضلع عمودی جدید ساخته شده است .بنابراین باید ضلع آبی رنگ به لیست اضلاع متصل به این گره اضافه شود و ضلع مشکی رنگ عمودی نیز از این لیست حذف شود.

با توجه به این توضیحات در بخش 26 سه ضلع جدید ساخته شده (دو ضلع سبز و یک ضلع آبی) متصل به گره قرمز رنگ (گره جدید ساخته شده) را به لیست اضلاع متصل به آن به کمک متد PointAddEdg اضافه می کنیم.

1. حذف المان­های قدیمی متصل به گره دوم (گره با اندیس 1) اضلاع عمودی جدید

بدین منظور در لیست المان­های متصل به گره مذکور جستجو می کنیم و آن المانی را که مطابق با المان­های قدیمی باشد را از آن لیست حذف می کنیم.

1. حذف اضلاع عمودی سابق متصل به گره دوم (گره با اندیس 1) اضلاع عمودی جدید

بدون توضیح

1. اضافه کردن دو المان جدید و ضلع عمودی جدید ساخته شده به لیست المان­ها و اضلاع متصل به گره دوم ( گره با اندیس 1) ضلع عمودی جدید

بدون توضیح

1. اضافه کردن اضلاع عمودی اصلاح شده و چهار المان متصل به گره جدید به لیست اجزاء متصل به آن گره

در آخرین بخش این قسمت کافیست ضلع عمودی که قبلاً اصلاح شده است (ضلع مشکی رنگ) را به لیست اضلاع متصل به گره جدید ساخته شده (گره قرمز رنگ) اضافه کنیم. سپس دو المان جدید ساخته شده (المان های زرد رنگ ) و المان های اصلاح شده (المان های سفید رنگ) را نیز به لیست المان­های متصل به گره جدید اضافه می کنیم.

فرآیند تشکیل لایه جدید را می توانیم در مجموعه اشکال زیر ملاحظه کرد:

÷

1. روند تشکیل یک لایه جدید
2. شروع فرآیند اضافه کردن یک لایه جدید برای حالتی که لایه از دو طرف به لایه رابط متصل باشد و دقیقا در زیر لایه رابط نیز قرار گرفته باشد

در این بخش دستورات مربوط به لایه ای را بررسی خواهیم کرد که دقیقاً زیر لایه رابط قرار گرفته و از دو طرف نیز منتهی به لایه رابط است . شکل زیر را در نظر بگیرید :



1. لایه ورودی دقیقا زیر لایه رابط قرار گرفته است

در تصویر فوق همانطور که مشاهده می شود لایه ورودی همان لایه ای است که با خط مشکی کشیده شده است. خطوط آبی رنگ اضلاع لایه رابط می باشند. اگر المان های بالای لایه مشکی رنگ به عنوان المان های لایه ورودی برای تقسیم شدن ، انتخاب شوند در این صورت با لایه ای مواجه می شویم که دقیقا زیر لایه رابط قرار گرفته و اولین المان آن دارای دو ضلع بر روی لایه رابط می باشد (ضلع آبی رنگ افقی بالا و ضلع آبی رنگ عمودی سمت راست) . مشابه همین حالت برای آخرین المان لایه نیز اتفاق می افتد و آن المان نیز دارای دو ضلع بر روی لایه رابط خواهد بود. با این تفاسیر ، آرایه ذخیره اضلاع روی لایه رابط که در ابتدای متد add پر شده بود و اضلاع دو المان اول و آخر لایه را که بر روی لایه رابط قرار داشتند در خود ذخیره کرده بود ، اکنون دارای 4 درایه پر شده می باشد. از این چهار درایه ما فقط نیاز به دو درایه داریم. این دو درایه مربوط به اضلاع آبی رنگ عمودی می باشند که باید در المان اول و المان آخر، آن ها را پیدا کنیم. در همینجا مشخص می شود که تنها تفاوت مهم این حالت با حالتی که لایه در زیر لایه رابط قرار نداشت در این است که ما باید این دو ضلع عمودی را بیابیم. مابقی دستورات کاملاً مشابه حالت قبل است.

در ابتدای این بخش بررسی می شود که اگر تعداد اضلاع روی لایه رابط دو المان اول و آخر برابر با 4 باشد و همچنین لایه حلقوی نباشد آنگاه دستورات بعدی مرتبط با این حالت اجرا شود. در اینجا یک آرایه با دو درایه تعریف می شود تا دو ضلع متصل به لایه رابط که با لایه ورودی متقاطع هستند در آن ذخیره شود.

1. پیدا کردن اضلاع روی لایه رابط که با لایه ورودی متقاطع باشند

ابتدا بررسی می شود که اگر تعداد اضلاع روی لایه رابط برابر 4 باشد (n=4) و لایه حلقوی نباشد آن گاه وارد مجموعه دستورات شویم. در ابتدا یک آرایه nE از نوع EDGE\* تعریف می کنیم. این آرایه قرار است آن دو ضلع عمودی توضیح داده شده را در خود نگهداری کنند. متغیر q نیز که یک شمارنده می باشد را هم تعریف می کنیم. برای پیدا کردن ضلع عمودی روی لایه رابط ابتدا به شکل زیر توجه کنید :

C[0]

C[1]

S.E[0]

S.E[1]

IE[1]

IE[0]

C[0]

C[1]

S.E[0]

S.E[1]

IE[0]

IE[1]

(b)

(a)

1. دو حالت ممکن برای نحوه قرارگیری اضلاع IE[0] و IE[1]

همانطور که در ابتدای متد توضیح داده شده است زمانی که در آرایه IE هر چهار درایه پر شوند آن گاه دو آرایه اول مربوط به اولین المان لایه و دو آرایه آخر مربوط به آخرین المان لایه می باشند. شکل 11 دو حالت ممکن برای قرارگیری اضلاع IE[0] و IE[1] را در المان اول نمایش می دهد. در حالت (a) ضلع IE[0] ضلع عمودی روی لایه رابط است و ضلعIE[1] ضلع افقی روی لایه رابط است. در شکل (b) برعکس این حالت اتفاق افتاده است. در اینجا می خواهیم راهی پیدا کنیم که ضلع عمودی را پیدا کرده و آن را حتما د IE[0] قرار دهیم. الگوریتم پیشنهادی به این صورت است که ما در هر حالت ضلع مقابل به ضلع IE[0] را در المان C[0] پیدا کنیم. اگر IE[0] یک ضلع عمودی باشد (مانند شکل (a) ) آن گاه ضلع مشکی رنگ ضلع مقابل به آن در المان C[0] می باشد. از طرفی ضلع مشکی رنگ در المان C[0] ، یکی از اضلاع المان C[1] نیز می باشد.

اما اگرIE[0] ضلع افقی باشد (مانند شکل (b) ) ، آن گاه ضلع مقابل به آن در المان C[0] ، ضلع قرمز رنگ (S.E[0]) می باشد. اما این ضلع یکی از اضلاع المان C[1] نخواهد بود. از این خاصیت استفاده می کنیم و ضلع عمودی را پیدا می کنیم. مطابق این کار را برای المان آخر نیز انجام می دهیم.

1. تغییر درایه­های آرایه IE

در این قسمت اولین و دمین درایه آرایه IE را به ترتیب مطابق با اولین و دومین درایه آرایه nE ، قرار می دهیم. با این کار از بخش بعدی همه چیز مطابق حالت قبلی اضافه کردن لایه خواهد بود.

1. مطابق بخش 6
2. مطابق بخش 7
3. مطابق بخش 8
4. مطابق بخش 9
5. مطابق بخش 10
6. مطابق بخش 11
7. مطابق بخش 12
8. مطابق بخش 13
9. مطابق بخش 14
10. مطابق بخش 15
11. مطابق بخش 16
12. مطابق بخش 17
13. مطابق بخش 18
14. مطابق بخش 19
15. مطابق بخش 20
16. مطابق بخش 21
17. مطابق بخش 22
18. مطابق بخش 23
19. مطابق بخش 24 و 25
20. مطابق بخش 26
21. مطابق بخش 27
22. مطابق بخش 28
23. مطابق بخش 29
24. مطابق بخش 30

در این جا دستورات مربوط به این حالت لایه تمام می شود.

1. شروع فرآیند اضافه کردن یک لایه جدید برای حالتی که لایه از یک طرف به لایه رابط و از طرف دیگر به مرز منتهی شود

در این بخش فرآیند اضافه کردن لایه به لایه ای را اجرا می کنیم که از یک طرف متصل به لایه رابط و از طرف دیگر متصل به مرز باشد. شرط ورود به این بخش آن است که در دو المان اول و آخر لایه مجموعا فقط یک ضلع روی لایه رابط و یک ضلع مرزی وجود داشته باشد و همچنین لایه حلقوی نباشد.ابتدا به شکل زیر توجه کنید:

1. لایه ای که از یک طرف منتعی به لایه رابط و از طرف دیگر منتهی به مرز می باشد

در شکل فوق لایه ای را مشاهده می کنیم که از سمت راست منتهی به مرز و از سمت چپ منتهی به لایه رابط است. می خواهیم اینگونه قرارداد کنیم که اولین المان لایه حتماً سمت لایه رابط باشد. در شکل فوق یعنی المان زرد رنگ ، اولین المان لایه و المان آبی رنگ آخرین المان لایه باشد. این قضیه کاملاً به نحوه استخراج یک لایه که توسط متد extend انجام می شود ، مربوط است. در ابتدای این بخش این موضوع را بررسی می کنیم که آیا المانی که به لایه رابط متصل است در ابتدای لیست قرار دارد یا نه. اگر قرارداشت که هیچ ولی اگر این المان در ابتدای لیست نباشد آن گاه به کمک متد flip از کلاس surface می توانیم ترتیب اضلاع و المان های لایه را برعکس کنیم به صورتی که المان زرد رنگ در ابتدای لیست قرار بگیرد.

از این بخش به بعد برنامه بسیار شبیه حالت­های قبلی است با این تفاوت که ما این بار تمامی عملیات را فقط در یک سمت( سمت لایه رابط) انجام می دهیم و کاری به سمت دیگر یعنی مرز نداریم.

1. اصلاح جهت قرار گیری ضلع روی لایه رابط

در این بخش مطابق بخش 6 قرارداد می کنیم که گره­های ضلع لایه رابط به گونه ای قرار بگیرند که گره با اندیس 1 آن منطبق بر ضلع ورودی لایه باشد. ولی در اینجا این کار را فقط برای ضلع موجود در آرایه IE[0] که تنها ضلع لایه رابط است انجام می دهیم.

1. ا اصلاح المان بندی بیرون لایه ( المان های درون حوزه simple)

مطابق بخش 7 باید المان بندی بیرون لایه را تکمیل کنیم. ولی در اینجا فقط کافیست که این کار را در سمت لایه رابط انجام دهیم و در سمت ضلع مرزی نازی به این کار نیست. این کار به کمک متد I\_add انجام می شود.

1. ایجاد فضای لازم در حافظه برای ساخت گره بر روی اضلاع عمودی

مطابق بخش 8 با این تفاوت که فقط گره ابتدایی بر روی لایه رابط را در اینجا تشکیل می دهیم. گره بر روی ضلع مرزی در ادامه در سایر بخش­ها تشکیل خواهد شد.

1. پیدا کردن مختصات وسط هر ضلع عمودی لایه و اضافه کردن آن مختصات به آرایه تشکیل شده در بخش 61

مطابق بخش 9 مختصات وسط اضلاع عمودی پیدا می شود و همچنین جهت گیری اضلاع نیز درس می شود.

1. مطابق بخش 10
2. مطابق بخش 11
3. مطابق بخش 12
4. مطابق بخش 13
5. مطابق بخش15
6. مطابق بخش 16
7. مطابق بخش 17
8. مطابق بخش 18
9. مطابق بخش 19
10. مطابق بخش 21 و 22

تنها تفاوت این بخش با دو بخش مذکور این است که در این بخش عملیات اضافه و حذف فقط مربوط به گره P1 می باشد و گره P2 وجود ندارد.

1. مطابق بخش 24 و 25

تنها تفاوت این بخش مانند بخش قبل آن است که عملیات صرفاً برای گره جدیدی که روی اولین ضلع عمودی یعنی ضلع روی لایه رابط، ساخته شده است صورت می پذیرد. این گره اولین گرهی است که به شبکه اضافه شده است.

1. اضافه کردن سه ضلع جدید ساخته شده به لیست اضلاع متصل به گره­های جدید

این بخش مطابق بخش 26 می باشد با این تفاوت که در بخش 26 آخرین گره که بر روی ضلع لایه رابط المان آخر قرار داشت، اطلاعات آن در خارج حلقه تکمیل می شد اما در اینجا که آخرین گره بر روی ضلع مرزی المان آخر واقع شده است اطلاعات آن در داخل همین حلقه تکمیل می شود. برای تکمیل اطلاعات این گره باید به این نکته اشاره شود که به لیست اضلاع آن فقط بایستی یک ضلع افقی جدید و یک ضلع عمودی جدید اضافه شود درحالیکه برای مابقی گره ها دو ضلع افقی و یک ضلع عمودی اضافه می شود. به همین علت در این بخش با بررسی مقدار شمارنده حلقه و در نظر گرفتن شرط i!=C.size()-1 عملیات را برای مابقی گره ها انجام می دهیم. در صورتیکه شمارنده حلقه به آخر برسد فرآیند اضافه کردن یکی از اضلاع افقی برای گره آخر اتفاق نمی افتد.

1. حذف آخرین المان اصلاح شده از لیست المان­های متصل به گره دوم آخرین ضلع عمودی جدید

این بخش فقط زمانی اجرا می شود که شمارنده حلقه در بخش 74 به انتها رسیده باشد (یعنی فرآیند اصلاح برای آخرین گره یا ضلع صورت پذیرد). در اینجا فقط بایستی آخرین المان اصلاح شده از لیست المان های متصل به گره مذکور حذف شود.

1. حذف دو المان اصلاح شده از لیست المان های متصل به گره­های دوم اضلاع عمودی ساخته شده

این بخش تا زمانی که شمارنده حلقه بخش 74 به آخر نرسیده باشد اجرا می شود و دو المان اصلاح شده چپ و راست گره دوم اضلاع عمودی ساخته شده را از لیست المان­های متصل به آن گره­ها حذف می کند. رجوع شود به شکل 8.

1. حذف ضلع عمودی اصلاح شده از لیست اضلاع متصل به گره دوم ضلع عمودی جدید ساخته شده

بدون توضیح

1. اضافه کردن المان­های جدید ساخته شده بهمراه ضلع عمودی جدید ساخته شده به لیست اجزاء گره دوم ضلع عمودی جدید ساخته شده

تنها توضیح مورد نیاز این بخش ان است که تا زمانیکه شمارنده حلقه به انتها نرسیده است به هر گره باید دو المان و یک ضلع اضافه شود اما برای گره دوم آخرین ضلع عمودی باید یک المان و یک ضلع بهلیست اجزاء آن اضافه شود.

1. اضافه کردن المان­های جدید ساخته شده و المان­های اصلاح شده به لیست المان­های متصل به گره­های جدید ساخته شده

در این بخش باید دو المان جدید ساخته شده و دو المان قدیمی اصلاح شده در سمت چپ و راست گره جدید به لیست المان­های متصل به گره­های جدید اضافه شود­ . اما برای آخرین گره فقط بایستی یک المان جدید و یک المان اصلاح شده به آن لیست اضافه شوند.

1. شروع فرآیند اضافه کردن یک لایه جدید برای حالتی که لایه حلقوی باشد

اگر لایه ورودی حلقوی باشد یعنی مقدار o برابر با 1 باشد آن گاه دستورات این بخش اغاز می شود. ابتدا باید به این نکته اشاره کنیم که این لایه معمولا با لایه رابط ارتباطی ندارد. به همین علت معمولا آرایه IE در این حالت کاملا خالی می باشد. البته یک حالت خاص نیز وجود دارد آن هم این است که لایه حلقوی دقیقاً زیر لایه رابط باشد که در آن صورت تمامی المان های لایه ، یک ضلع مشترک با لایه رابط خواهند داشت. بدین ترتیب دو درایه از آرایه IE نیز پر می شود (ضلع مربوط به المان اول و آخر) اما این دو ضلع موجود در IE هیچ اهمیتی برای ما ندارند زیرا در حالت لایه حلقوی ما نیاز به پیدا کردن ضلع عمودی بین دو المان اول و آخر داریم که باعث شده است این دو المان به یکدیگر متصل شوند. برای نزدیک شدن ذهن به موضوع به شکل زیر توجه کنید:

C[0]

C[1]

C[2]

C[3]

1. یک لایه حلقوی فرضی با 4 المان

شکل فوق یک شکل فرضی است . در عمل یک لایه حلقوی تعداد بسیار زیادی المان دارد اما در اینجا برای سادگی نمایش فقط از چهار المان استفاده شده است. در اینجا المان C[0] اولین المان لایه و المان C[3] آخرین المان لایه است. چهار ضلع روی دایره داخلی (اضلاع قرمز رنگ) ، اضلاع لایه ورودی می باشند. همانطور که مشخص است در این حالت المان اول با المان آخر همسایه هستند. لذا در این حالت حتما یک ضلع وجود دارد که این دو المان را به یکدیگر متصل کرده است. این ضلع در شکل همان ضلع آبی رنگ می باشد. حال اگر فرض کنید که دایره مشکی رنگ لایه رابط باشد و لایه ورودی دقیقا زیر لایه رابط قرار گرفته است باز هم در این حالت برای ما فرقی نمی کند زیرا ما بدنبال همان ضلع آبی رنگ ذکر شده هستیم تا بتوانیم نقطه وسط آن را بیابیم و به نقطه وسط ضلع مقابلش در همان المان C[0] متصل کنیم. این کار را آنقدر تکرا می کنیم تا مجدداً به خود ضلع آبی رنگ برسیم.

در این بخش ابتدا یک متغیر واسط به نام lE تعریف می کنیم. سپس بر روی اضلاع المان اول جستجو می کنیم و هرکدام از آن اضلاع را که در المان آخر نیز وجود دارند را در متغیر lE ذخیره می کنیم.

پس از اینکه ضلع lE (ضلع آبی رنگ در شکل 13) را پیدا کردیم ، از این قسمت برنامه به طورکلی مشابه حالت های قبل می باشد. فقط اندک تفاوت هایی در تعداد گره ها و نحوه حرکت روی تعدا المان ها وجود دارد. مثلا در این حالت تعداد گره های اضافه شده به مجموعه با تعداد المان ها برابر است زیرا ضلع اول که گره اول روی آن قرار دارد در دو المان تکرار شده است. اما در حالت های قبل تعداد گره های اضافه شده همیشه یکی بیشتر از تعداد المان ها بود.

1. مطابق بخش 8 و 9

تنها تفاوت آن در اندازه آرایه ساخته شده است که یک واحد کمتر از حالت­های قبل می باشد.

1. مطابق بخش 10
2. مطابق بخش 11
3. مطابق بخش 12
4. اضافه کردن مختصات اولین گره ساخته شده به لیست گره­های موجود در شبکه هندسی

بدون توضیح

1. مطابق بخش 13

بدون توضیح

1. مطابق بخش 14
2. مطابق بخش 15
3. پیدا کردن مجدد ضلع مشترک المان اول و آخر

بدون توضیح

1. ساخت اولین ضلع عمودی جدید و اصلاح ضلع عمودی سابق

در این بخش اولین ضلع عمودی جدید را بدین صورت می سازیم که گره اول آن همان گره جدید ساخته شده باشد و گره دوم آن برابر با گره اول ضلع عمودی سابق باشد(توجه شود این گره بر روی لایه ورودی واقع نشده است). همچنین در اینجا ضلع عمودی سابق نیز اصلاح می شود. بدین صورت که فقط گره اول ان برابر با گره جدید ساخته شده خواهد شد. در انتهای این بخش یک واحد به تعداد اضلاع شبکه هندسی اضافه می شود.

1. مطابق بخش 16

تنها تفاوت آن با بخش 16 آن است که در این حالت ضلع اول در خارج حلقه ساخته شده و اصلاح شده است. مابقی اضلاع درون این حلقه ساخته می شوند.

1. مطابق بخش 17
2. مطابق بخش 19
3. مطابق بخش 20
4. پیداکردن مجدد ضلع مشرک المان اول و آخر

بدون توضیح

1. اضافه کردن چهار ضلع (دو ضلع افقی و دو ضلع عمودی) به لیست اضلاع متصل به اولین گره جدید

در این بخش چهار ضلع را به لیست اضلاع متصل به اولین گره اضافه می کنیم. این چهار ضلع عبارتند از :

الف) اولین ضلع افقی ساخته شده

ب) آخرین ضلع افقی ساخته شده

ج) اولین ضلع عمودی ساخته شده

د) اولین ضلع عمودی اصلاح شده

1. اضافه کردن چهار المان به لیست المان­های متصل به اولین گره جدید

در این بخش چهار المان که شامل دو المان جدید ساخته شده و دو المان اصلاح شده می باشند به لیست المان­های متصل به اولین گره جدید اضافه می شوند. این المان­ها عبارتند از:

الف) اولین المان جدید

ب) آخرین المان جدید

ج) اولین المان اصلاح شده

د) آخرین المان اصلاح شده

1. اضافه کردن المان­ها و اولین ضلع جدید به لیست اجزاء گره دوم اولین ضلع جدید

همانطور که تا اینجا مشخص است تمام عملیات مربوط به اولین گره ها و اولین اضلاع را در خارج از حلقه انجام می دهیم. این بار به لیست اضلاع متصل به گره دوم اولین ضلع جدید ساخته شده باید همان ضلع را اضافه کنیم. همچنین اولین و آخرین المان جدید را نیز بایستی به لیست المان­های متصل به آن اضافه کنیم.

1. حذف اولین ضلع عمودی اصلاح شده از لیست اضلاع متصل به گره دوم اولین ضلع عمودی جدید

در این بخش باید ضلع قدیمی که قبلا متصل به این گره بود را از لیست اضلاع متصل به آن حذف کنیم زیرا اکنون اولین ضلع عمودی جدید به آن متصل شده است.

1. حذف اولین و آخرن المان اصلاح شده از لیست المان­های متصل به گره دوم اولین ضلع عمودی جدید

بدون توضیح

1. اضافه کردن اضلاع جدید ساخته شده به لیست اضلاع متصل به گره­های جدید

در این بخش عملیات بخش 96 را برای تمامی گره­های جدید ساخته شده به غیر از گره اول انجام می دهیم. البته لازم به ذکر است که در اینجا هنوز اضلاع اصلاح شده را به این گره­ها اضافه نمی کنیم و این کار را در بخش­های بعدی انجام خواهیم داد. در این بخش صرفا دو ضلع افقی و یک ضلع عمودی را به این گره­ها اضافه می کنیم.

1. حذف اضلاع اصلاح شده از لیست اضلاع متصل به گره­های دوم اضلاع عمودی جدید

این بخش مشابه بخش 99 است با این تفاوت که برای مابقی گره­های دوم اضلاع عمودی جدید نوشته شده است.

1. حذف المان­های اصلاح شده از لیست المان­های متصل به گره دوم اضلاع عمودی جدید

این بخش مطابق بخش 100 است با این تفاوت که برای مابقی گره­های دوم اضلاع عمودی جدید نوشته شده است.

1. اضافه کردن اضلاع عمودی جدید و المان­های جدید به لیست اضلاع و المان­های متصل به گره­های دوم اضلاع عمودی جدید

بدون توضیح

1. اضافه کردن اضلاع اصلاح شده بهمراه چهار المان به لیست اجزاء متصل به گره­های جدید ساخته شده

در ابتدای این بخش اضلاع عمودی اصلاح شده را که در بخش 101 به گره­های جدید اضافه نکردیم، اضافه می کنیم. سپس دو المان جدید و دو المان اصلاح شده قدیمی را به لیست المان­های مصل به گره­های جدید اضافه می کنیم.

در اینجا متد Add(SURFACE,Int) به اتمام می رسد.