



**زیربرنامه:**

Cell.cpp

|  |  |  |
| --- | --- | --- |
| **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

هدف از تشکیل این کلاس ذخیره المان های یک شبکه هندسی به صورت اشیاء مجزا با خواص مشخص است.

1. توضیحات و تئوری­ها

بدون تئوری

1. بخش­های زیربرنامه
2. تعریف متغیر­های Global استفاده شده در کل پلتفرم

در این بخش متغیر­هایی که به عنوان متغیر Global در کل برنامه شناخه می شوند، تعریف شده است.

1. تعریف متد area

خروجی این متد از نوع double بوده و هدف آن محاسبه مساحت یک المان است . در اینجا اشاره به این نکته ضروری است که ما مساحت یک المان مثلثی را با استفاده از قاعده **هرون** محاسبه می کنیم. همچنین مساحت یک المان مربعی را با تبدیل آن المان به دو المان مثلثی و مجددا استفاده از قاعده هرون ، محاسبه خواهیم کرد. اما رابطه هرون در یک مثلث به صورت زیر بیان می شود :

در این رابطه S بیانگر مساحت ، P بیانگر نصف محیط مثلث ، a، b و c نیز طول هر یک از اضلاع مثلث می باشد.

در ابتدای این متد ابتدا بررسی می شود که تعداد اضلاع المان چند است. در صورتی که تعداد اضلاع برابر با 3 باشد متد وارد قسمت اول می شود. در این قسمت تعدادی متغیر از نوع double تعریف می شود. اولین متغیر a می باشد که در نهایت جذر آن به عنوان خروجی این تابع در این حالت بازگردانده می شود. متغیر دوم p می باشد که برابر با نصف محیط این المان است. سه متغیر بعدی که به ترتیب l0 ، l1 و l2 نام دارند برابر با طول سه ضلع این المان می باشد. پس از آن متغیر a طبق رابطه هرون محاسبه شده و از آن جذر گرفته می شود و به عنوان خروجی برگردانده می شود. در بخش دوم و برای المان های چهار ضلعی ابتدا گره های دو ضلع از مربع را (اضلاعE[0] و E[2] ) به گونه ای مرتب می کنیم که گره های با اندیس0 آن دو یکسان باشد. آنگاه مسلماًگره های با اندیس 1 این دو ضلع دو سر قطر مربع خواهند بود. اکنون قطر را به عنوان یک ضلع جدید تعریف کرده و دو مثلث جدید می سازیم. مساحت چهارضلعی اولیه برابر با مساحت دو مثلث جدید است.

1. تعریف متد Cosin

خروجی این متد از نوع double است و هدف آن محاسبه کسینوس زاویه اضلاع متصل به یک گره در داخل یک المان است. بدین منظور ابتدا باید مشخص شود که کدام اضلاع المان به این گره متصل هستند. پس از آن دو گره دیگر این اضلاع مشخص شوند. پس از مشخص شدن این دو گره می توانیم با استفاده از گره اولیه و این دو گره پیدا شده دو بردار بسازیم و با استفاده از رابطه محاسبه زاویه بین دو بردار ، این زاویه را محاسبه می کنیم.

این فرمول به صورت زیر است:

1. تعریف متد perimeter

خروجی این متد از نوع double بوده و هدف آن محاسبه مقدار محیط یک المان است. بدین منظور فقط بایستی طول اضلاع یک المان را بدست آورده و آن ها را با یکدیگر جمع کنیم.

1. تعریف متد across(int)

متد across متدی است که فقط برای المان های چهار گرهی بکار می رود و هدف آن مشخص کردن آن ضلعی از المان است که با ضلع ورودی نقطه اشتراک ندارد و درواقع مقابل ضلع ورودی قرار گرفته است. از این متد فقط زمانی می توان استفاده کرد که المان های یک حوزه که همه ی آن ها مربعی هستند توسط متد order که مربوط به کلاس ZONE می باشد به صورت مرتب شده در آمده باشند . متد order را در جای خود به طور کامل توضیح خواهیم داد در اینجا همینقدر لازم است بدانیم که اضلاع المان های مربعی پس از مرتب شدن به گونه ای در آرایه E آن المان قرار می گیرند که در درایه های با اندیس 0 و 1 دو ضلع مقابل بهم و در درایه های با اندیس 2 و 3 دو ضلع دیگر مقابل بهم قرار می گیرند.

متد across در این قسمت یک متغیر int به عنوان ورودی گرفته که این متغیر در واقع همان اندیس ضلع مورد نظر است. با توجه به توضیحات ارائه شده ضلع مقابل به ضلع ورودی طبق شرایط زیر تعیین می شود :

* اگر اندیس ورودی 0 باشد آن گاه ضلع مقابل دارای اندیس 1 است
* اگر اندیس ورودی 1 باشد آن گاه ضلع مقابل دارای اندیس 0 است
* اگر اندیس ورودی 2 باشد آن گاه ضلع مقابل دارای اندیس 3 است
* اگر اندیس ورودی 3 باشد آن گاه ضلع مقابل دارای اندیس 2 است

1. تعریف متد across(Edge)

این متد کاملا مشابه متد بخش 5 است و تنها تفاوت آن این است که ورودی متد بجای شماره اندیس ، آدرس یک ضلع از اضلاع آن المان می باشد. در این متد ابتدا بررسی می شود که ضلع ورودی دارای چه شماره اندیسی در داخل المان است و پس از آن مشابه متد قبل ضلع مقابل شناسایی می شود.

1. تعریف متد is\_in(Edge)

خروجی این متد از نوع bool بوده و هدف آن این است که مشخص کند آیا ضلعی که به عنوان ورودی به متد داده شده است ، عضوی از این المان می باشد یا خیر. ورودی متد از نوع EDGE\* می باشد که در واقع آدرس یکی از اضلاع شبکه است. برای بررسی اینکه آیا این ضلع عضو المان است کافیست که این ضلع با تک تک اضلاع المان مقایسه شده تا نتیجه مشخص شود.

1. تعریف متد center

خروجی این متد از نوع bPOINT می باشد و هدف آن ساختن یک گره در وسط یک المان است . مختصات این نقطه میانگین مختصات نقاط یک المان است که هم در حالت المان سه گرهی و هم در حالت المان 4 گرهی معتبر است.