



**زیربرنامه:**

Analyse()

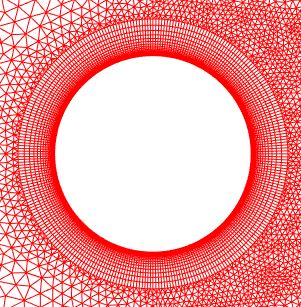
|  |  |  |
| --- | --- | --- |
| **توسعه دهندگان** | مرتضی نامور |  |
| **تهیه کنندگان مستند** | مرتضی نامور | |
| **تاییدکنندگان** | مرتضی نامور | |
| **تاریخ تنظیم سند** | 22/02/1394 | |
| **شناسه سند** | **MC2F003F1** | |
| **زبان برنامه‌نویسی** | **Fortran 90** | |

1. وظایف

وظیفه این متد، بخش بندی و حوزه بندی شبکه هندسی است. در واقع این متد به هر المان یک شماره اختصاص می دهد که این شماره، شماره حوزه ایست که المان در آن واقع شده است.

1. توضیحات و تئوری­ها

هر بخش از یک شبکه هندسی که دارای المان های مشابه و متصل به یکدیگر باشد یک حوزه (Zone) نامیده می شود. حوزه ها معمولا از یک سمت به مرزهای داخلی یا خارجی متصل هستند و از یک سمت به لایه های رابط .



1. دو حوزه متفاوت در یک شبکه هندسی

در شکل 1 ، بخش بین مرز داخلی (سطح دایره) و لایه رابط که دارای المان های مربعی است حوزه شماره 1 می باشد. همچنین بخش بین لایه رابط و مرزهای خارجی نیز حوزه شماره 2 می باشند.

1. بخش­های زیربرنامه
2. مقداردهی اولیه به متغیر شماره حوزه

بدون توضیح

1. اختصاص عدد 0 به شماره حوزه هر المان مورد بررسی در ابتدای فرآیند

بدون توضیح

1. اختصاص شماره حوزه به هر المان

در این بخش با استفاده از یک حلقه بر روی تمام المان ها حرکت کرده و هر المان که عدد حوزه آن (CellZnum) برابر 0 باشد را برابر با مقدار موجود کنونی در متغیر شماره حوزه (ZNum) قرار می دهیم. سپس این المان را به آرایه ای به نام C\_stack اضافه می کنیم. وظیفه این آرایه ذخیره المان­هایی ست که هنوز شماره حوزه­ای به آن­ها اختصاص داده نشده است.

1. مقداردهی شماره حوزه به المان­های درون C\_stack

وظیفه این بخش که در ادامه بخش قبل و درون همان حلقه قرار گرفته است آن است که همیشه به المان موجود در درایه اول (اندیس 0) آرایه C\_stack ، مقدر عدد حوزه کنونی را اختصاص دهد. در اینجا از یک حلقه شرطی استفاده شده است و این فرآیند مادامی که شمارنده آرایه C\_stack یعنی (n) برابر صفر نشده است ادامه پیدا می کند.

1. پیدا کردن المان­های همسایه المان قبل

این بخش نیز در ادامه بخش قبل قرار گرفته و درون حلقه شرطی بخش قبل تعریف می شود. در این قسمت با نوشتن یک حلقه که بر روی اضلاع المان موجود در درایه اول آرایه C\_stack ، اضلاعی را می­یابیم که مقدار پارامتر state آن­ها برابر 1 باشد (بدان معنی که المان­های دو طرف این ضلع از یک نوع هستند یا هر دو مثلثی یا هردو مربعی) . پس از یافتن این اضلاع، با استفاده از دستور find المان­های متصل به این ضلع را پیدا می کنیم. طبیعتاَ یکی از این المان ها همان المان موجود در درایه اول آرایه C\_stack است. لذا در اینجا آن المانی که برابر با المان موجود در درایه اول C\_stack نباشد و همچنین عدد حوزه آن المان (CellZnum) برابر 0 باشد (یعنی هنوز هیچ عدد حوزه ای به آن داده نشده باشد) را پیدا کرده و به آرایه C\_stack اضافه می کنیم و شمارنده این آرایه را نیز یک واحد افزایش می دهیم.

1. جابجایی المان­های موجود در C\_stack

با توجه به اینکه در مراحل قبل تمامی فرآیند­ها بر روی المان موجود در درایه اول آرایه C\_stack انجام گرفت و اکنون این المان دارای شماره حوزه هست و المان­های مجاور آن نیز پیدا شده­اند، در این بخش کافی است جای آخرین المان موجود در آرایه C\_stack را با اولین المان عوض کنیم. حال المان جدیدی درون درایه اول C\_stack قرار گرفته و مجدداَ تمامی فرآیندها از بخش 4 تا این بخش بر روی این المان جدید تکرار می شود. در انتهای این بخش نیز یک واحد از شمارنده آرایه C\_stack کم می شود.

1. افزایش عدد حوزه

پس از اینکه تمامی المان­های درون یک حوزه شماره گذاری شوند، دیگر المانی درون C\_stack باقی نمی­ماند که شماره حوزه به آن اختصاص داده نشده باشد. در این حالت شمارنده آرایه C\_stack یعنی n برابر 0 شده و برنامه از حلقه شرطی بخش 4 خارج می شود. قبل از اینکه برنامه مجدداً به حلقه بخش 3 وارد شود کافیست پس از اتمام حلقه شرطی بخش 4 ، یک واحد به عدد حوزه (ZNum) اضافه کنیم. در این صورت تمامی المان­هایی که در دور بعدی پیدا می شوند شماره حوزه جدیدی را دریافت می کنند.