



**زیربرنامه:**

Read\_STL

|  |  |  |
| --- | --- | --- |
| **توسعه دهندگان** | مرتضی نامور |  |
| میثم وکیلی | D:\Darsi\Karshenasi Arshad\Others\usc logo.gif |
| **تهیه کنندگان مستند** | مرتضی نامور | |
| **تاییدکنندگان** | مرتضی نامور | |
| **تاریخ تنظیم سند** | 15/08/1394 | |
| **شناسه سند** | **MC5F001F2** | |
| **زبان برنامه‌نویسی** | **Fortran 90** | |

1. وظایف

در این زیربرنامه اطلاعات موجود در یک فایل STL خوانده می شود. خروجی این زیربرنامه تعداد نقاط و تعداد مثلث های تشکیل دهنده شبکه، شماره نقاط تشکیل دهنده هر مثلث و در نهایت مختصات هر کدام از نقاط تشکیل دهنده شبکه می باشد.

1. توضیحات و تئوری­ها

بطور کلی فایل های STL در بیشتر نرم افزارهای CAD و یا تولید شبکه تولید و یا قابل خوانده شدن می باشد. این فایل های هندسه و یا یک شبکه را بصورت مثلث هایی ذخیره می کنند و دارای یک فرمت استاندارد است که در آن مختصات سه نقطه تشکیل دهنده هر کدام از مثلث ها بصورت زیر ذخیره شده است:

Facet Normal Ni Nj Nk

Outer Loop

Vertex XV1 YV1 ZV1

Vertex XV2 YV2 ZV2

Vertex XV3 YV3 ZV3

Endloop

Endfacet

که در آن Ni Nj Nk بردار عمود بر مثلث در راستای محورهای مختصات و XVi YVi ZVi مختصات هر کدام از سه نقطه تشکیل دهنده مثلث می باشد. همچنین اگر هندسه مربوط به فایل STL دارای چندین بخش باشد، اطلاعات هر سطح بعد از نام آن قرار داده می شود.

در این برنامه ابتدا مختصات هر کدام از نقاط تشکیل دهنده مثلث ها در آرایه های دو بعدی ذخیره می شود. سپس به هر کدام از نقاط یک شماره اختصاص داده می شود و بطور همزمان نقاط تشکیل دهنده مثلث ها بصورت شماره نقاط (بجای مختصات آنها) ذخیره می گردد.

لازم است توجه شود که این فایل باید بصورت ستونی (که در بخش راهنمای کاربری برنامه "ReadSTLFile" توضیح داده شده است) درآورده شود تا این زیربرنامه توانایی خواندن آن را داشته باشد.

1. بخش­های زیربرنامه

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

1. باز کردن فایل

فایل حاوی اطلاعات فایل STL باز می شود. باید دقت شود که نام فایل باید STLIn.stl باشد.

1. عبور از دو خط ابتدای فایل

دو خط اول فایل حاوی اطلاعات مورد نیاز در مورد مثلث های شبکه نیستند. بنابراین با دو بار اجرای دستور Read از آنها عبور می کنیم.

1. مقداردهی اولیه به متغیرها

در این بخش ابتدا متغیر NC که نشان دهنده تعداد مثلثهای شبکه است با صفر مقدار دهی می شود. متغیر EOF (End Of File) نشان دهنده رسیدن به انتهای فایل می باشد. بنابراین تا زمانی که به انتهای فایل نرسیده ایم این متغیر مقدار صفر دارد، و زمانیکه به انتهای فایل برسیم مقدار این متغیر برابر 1 خواهد شد.

1. خواندن تعداد مثلث ها

هر هفت خط در فایل ورودی نشان دهنده مختصات نقاط تشکیل دهنده یک مثلث می باشد که دو خط اول و دو خط آخر حاوی اطلاعاتی در مورد مختصات مثلث نبوده و سه خط میانی حاوی سه نقطه تشکیل دهنده مثلث می باشد. در اینجا در یک حلقه تکرار به اندازه تمام مثلث های موجود، مختصات نقاط تشکیل دهنده آنها را خوانده و در آرایه های مربوطه ذخیره می کند.در این بخش فایل ورودی خط به خط خوانده شده و تا زمانی که به انتهای فایل برسیم اطلاعات مثلثهای موجود خوانده شده و در متغیرهای مربوطه قرار می گیرد. شرط داخل حلقه بررسی می کند که آیا به آخرین خط فایل رسیده ایم یا نه. یعنی تا زمانیکه متغیر EOF مخالف 1 باشد، این حلقه ادامه پیدا می کند.

1. شمارش تعداد مثلثها

در این بخش ابتدا توسط دستور Read یک خط خوانده می شود. این خط در واقع خط دوم از بخش مربوط به اطلاعات هر مثلث در فایل STL ورودی می باشد که حاوی عبارت outer loop است. چون این عبارت برای ما اهمیتی ندارد با یک دستور Read از آن می گذریم. در خط بعد یک واحد به متغیر حاوی تعداد مثلثهای شبکه (NC) اضافه می کنیم.

1. خواندن مختصات نقاط تشکیل دهنده مثلث

مختصات نقطه تشکیل دهنده یک مثلث در خطوط سوم، چهارم و پنجم قرار دارد که در این جا در یک حلقه تکرار مختصات هر سه نقطه خوانده می شود. متغیر TempStr که در دستور Read استفاده شده است فقط برای این است که عبارت vertex که در ابتدای خط حاوی مختصات نقاط تشکیل دهنده هر مثلث در فایل STL ورودی نوشته شده است، در این متغیر قرار بگیرد و شرایط برای قرار گرفتن مختصات نقاط در متغیرهای مربوطه فراهم شود.

1. عبور از دو خط انتهای اطلاعات هر مثلث

خط ششم و هفتم حاوی اطلاعات مورد نیاز در مورد مثلث های شبکه نیست. این دو خط به ترتیب حاوی عبارات endloop و endfacet می باشند. بنابراین با دوبار اجرای دستور Read از اطلاعات مربوط به این مثلث خارج می شویم.

1. خواندن خط بعد از اطلاعات هر مثلث

در این بخش، بعد از اینکه خواندن اطلاعات مربوط به یک مثلث در بخش قبل پایان رسید، یک سطر از فایل خوانده می شود و در متغیر RowStr قرار می گیرد. در بخش بعد این متغیر مورد تحلیل قرار خواهد گرفت.

1. بررسی متغیر RowStr

در این بخش بررسی می شود که آیا به انتهای فایل یا انتهای یک بخش از اطلاعات شبکه رسیده ایم یا خیر. چون برخی از فایلهای stl ممکن است دارای چندین بخش باشند که اطلاعات هر بخش با عبارت solid شروع شده و با عبارت endsolid به پایان می رسد. بنابراین در شرط If بیرونی بررسی می شود که آیا 8 کاراکتر اول متغیر RowStr که در بخش قبل خوانده شد برابر با رشته endsolid می باشد یا خیر. اگر برابر باشد دو حالت ممکن است رخ دهد: یا خط بعدی حاوی اطلاعات مربوط به یک بخش جدید از شبکه است، یا به انتهای فایل رسیده ایم. بدین منظور، بعد از شرط If بیرونی، یک خط از فایل خوانده شده و در متغیر RowStr قرار می گیرد. سپس در شرط If داخلی بررسی می شود که اگر 5 کاراکتر اول متغیر RowStr برابر با عبارت solid بود، بنابراین به یک بخش جدید از اطلاعات شبکه رسیده ایم و باید عملیات خواندن اطلاعات مثلث های آن بخش از شبکه انجام شود. در غیر اینصورت به انتهای فایل رسیده ایم و با اجرای قسمت Else، مقدار متغیر EOF برابر 1 می شود. در نتیجه در اجرای بعدی حلقه بخش 4، شرط ادامه حلقه نقض شده و از حلقه خارج می شویم.

1. مقداردهی اولیه

در ابتدای برنامه مقدار بسیار کوچک 0.00000001 در پارامتر Eps ذخیره می گردد تا در مراحل بعدی از آن استفاده گردد. همچنین شمارنده تعداد نقاط درون میدان NP در ابتدای زیربرنامه برابر صفر قرار داده می شود.

1. بررسی تمام مثلث های موجود

در یک حلقه تکرار هر سه نقطه تمام مثلث های موجود بررسی و شماره گذاری می شود. بنابراین دو حلقه تکرار یکی به تعداد مثلث های موجود و دیگری مربوط به سه نقطه تشکیل دهنده مثلث وجود دارد.

1. آیا نقطه مورد بررسی تکراری می باشد؟

در اینجا یک نقطه از مثلث مورد بررسی با تمام نقاط شماره گذاری شده مقایسه می گردد و درصورتیکه بر یکی از آنها منطبق باشد، ابتدا شماره آن نقطه بعنوان یکی از نقاط تشکیل دهنده مثلث در آرایه مربوطه ذخیره شده و سپس با استفاده از دستور Goto از حلقه موجود خارج و از انجام مرحله بعد صرفنظر شده و نقطه بعدی بررسی می شود.

1. شماره گذاری نقاط غیر تکراری و تکمیل نقاط تشکیل دهنده مثلث

در صورتیکه نقطه مورد بررسی تکراری نباشد وارد این مرحله می شویم. بنابراین در صورتیکه وارد این مرحله شویم، نقطه مورد بررسی تکراری نیست و بنابراین یک واحد به تعداد نقاط موجود اضافه شده و مختصات نقطه مورد بررسی در آرایه های جدید مربوطه ذخیره می گردد و شماره اختصاص داده شده به نقطه بعنوان نقطه تشکیل دهنده مثلث ذخیره می شود.