



تمرین برنامه نویسی پیشرفته

ماتریکس

استاد : بانو نرگس السادات بطحایان

محمد مهدی صفری

971235840

مقدمه

برنامه ی داده شده صرفاً برای یادآوری مطالب مبانی می باشد و برای آماده شدن برا مفاهیم درس پیشرفته که به صورت ماریکس تعبیه شده است

شروع

برنامه بصورت کامند لاینی نوشته شده است و با دستور **exit** برنامه تمام می شود

با دستور **getline** تمامی یک خطی که کاربر وارد کرده است را دریافت کرده و به صورت استرینگ به تابع

whatCommand فرستاده - این تابع بصورت

ماژولار در یک ماژول جداگانه نوشته شده است در اصل در این تابع ما هر کلمه موجود در استرینگ را داخل وکتوری از استرینگ ها پوش کرده و این وکتور را در انتهای تابع به تابع اصلی بر می گردانیم - در این تابع اگر تعداد کلمه ها بیشتر از ۶ و یا کمتر از ۲ باشد اکستنشنی پرتاب می شود و برنامه لیست کامندها را نمایش می دهد.

حال وکتوری را که هر خانه از آن یک کلمه است را در تابع اصلی دریافت کرده‌ایم را برای انجام عمل مورد انتظار مورد بررسی قرار می‌دهیم - همانطور که متوجه هستیم خانه • ([getCommand[0]) این وکتور دستور مورد نظر ما را تعیین می‌کند (add, is_diagonal, ...) و با if و else if ها توابعی که را که از ما خواسته‌اید را با توجه به خانه • و با آرگومان‌های مناسب فراخوانی میکنیم - بدیهی است که اگر کاربر دستور اشتباهی وارد کند برنامه لیست کامندها را نمایش می‌دهد.

توابع

و اما قسمت توابع که در قسمت قبل گفتیم - به صورت
ماژولار تمام توابعی را که از ما خواسته بودید در فایل
`func.cpp` پیاده سازی شده‌اند - وکتوری از اشاره‌گری به
یک ماتریکس (`mt`) تعریف کرده‌ایم که تمام ماتریس‌ها در
این وکتور ذخیره می‌شوند - در این فایل تابع مخصوصی
برای سرچ کردن نام در ماتریس‌ها داریم که اگر ماتریسی با
نام ارسال شده موجود باشد ایندکس مورد نظر از وکتور `mt`
را بر می‌گرداند در غیر اینصورت ۱- را بر می‌گرداند.
در بقیه تابع‌ها نیز توابع را پیاده سازی کرده‌ایم

و اما فایل `matrix.cpp` در این فایل یک کانستراکتور داریم که در آن نام ماتریکس و تعداد سطر و ستون را ذخیره می‌کنیم و مقدار `float**` را که عضو داده‌ای ماتریکس است را به صورت پویا یک آرایه دوبعدی از آن می‌سازیم و در دیستراکتور نیز همین آرایه ۲ بعدی پویا را `delete` می‌کنیم - از آنجایی که نوشتن تابع برای کلاس مجاز نبود عضوهای داده‌ای ماتریکس را نیز به صورت `public` در نظر گرفتیم که عملاً کار اشتباه‌ای که باید برای دریافت این مقادیر توابعی در نظر گرفته میشد.