

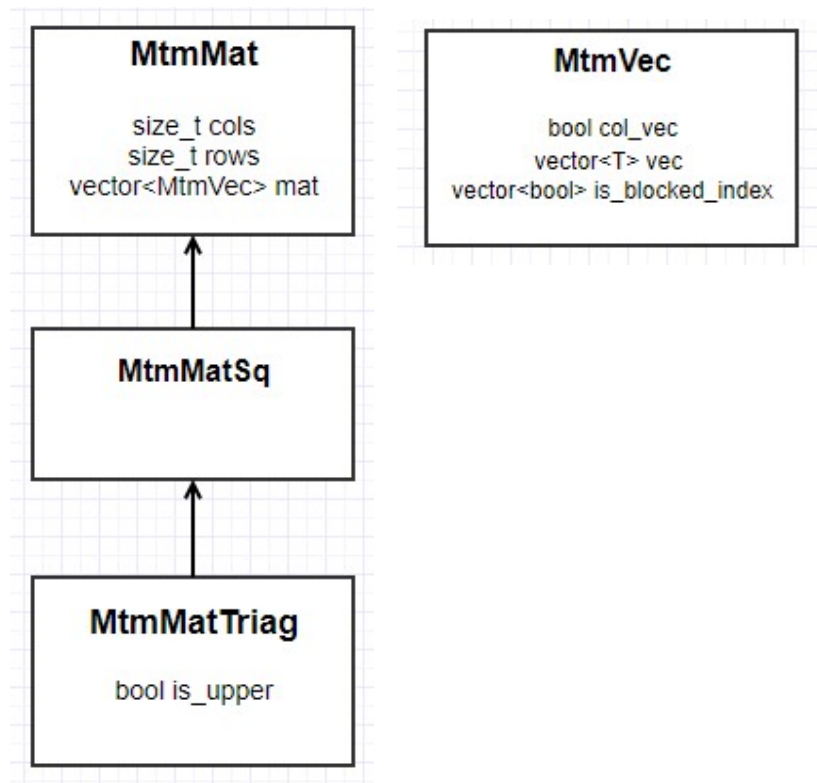
מבוא לתכנות מערכות - 234124

תרגיל בית 3 - חלק יבש

שם: נדב אביוב
ת.ז: 206023772

שם: מאי פליסטר
ת.ז: 207683418

בחרנו לממש את המערכת מבחינת יחסי הורשה בצורה הבאה:



מחלקת MtmMat מכילה את השדות הבאים:

- Row – שדה המכיל את מספר שורות המטריצה.
- Col – שדה המכיל את מספר עמודות המטריצה.
- `vector<MtmVec<T>> mat` - בחרנו לממש את המטריצה שלנו כווקטור של `MtmVec<m>`, כיוון שניתן להתייחס לכל הפעולות האלגבריות על מטריצות כפעולות אלגבריות על הווקטורים המרכיבים אותם. לכן במימוש הפעולות האלגבריות על המטריצות השתמשנו בפעולות שכבר מימשנו ב-MtmVec ובכך מנענו שיכפול קוד, וגם הפכנו את הקוד לקריא יותר.

מחלקת MtmMatSq :

- בחרנו לממש את המחלקה כיורשת מ-MtmMat. הלכנו לפי הכלל "B is a A" - מטריצה ריבועית היא סוג של מטריצה רגילה. המטריצה הריבועית נדרשת לבצע כמעט את אותם הפעולות שמטריצה רגילה צריכה לבצע, ולכן באמצעות ההורשה מנענו שיכפול קוד. נציין כי לא הוספנו שדות נוספים למtmMatSq.

מחלקת MtmMatTriag

- בחרנו לממש את המחלקה כיורשת מ MtmMatSq מאותן סיבות כמו במחלקה הקודמת- מטריצה משולשת היא סוג של מטריצה ריבועית וצריכה לבצע כמעט את אותם פעולות. בנוסף הוספנו שדה `is_upper` המציין אם המטריצה היא משולשת עליונה או תחתונה. כמו כן הוספנו פונקציות היחודיות למחלקה זו.

מחלקת MtmVec מכילה את השדות הבאים:

- Col_vec – שדה בוליאני המציין אם הווקטור הוא ווקטור עמודה או שורה.
- Vector<T> vec – ווקטור מספריית stl המכיל משתנים מסוג T.
- Vector<bool> is_blocked_index – ווקטור המציין אם האיבר הנמצא באותו אינדקס בווקטור vec ניתן לקריאה ולכתיבה. במקרה שאיבר כלשהו באינדקס is_blocked_index מוגדר כ-True, המשמעות היא שלא יהיה ניתן לגשת לאיבר הנמצא באינדקס דומה ב-vec באמצעות אופרטור []. הגדרנו שדה זה כ-True רק עבור האיברים במטריצות משולשות הנמצאים מעל/מתחת להאלכסון הראשי במטריצה (עבור משולשת תחתונה ועליונה בהתאמה), כדי למנוע השמה שגויה של איבר במטריצה באמצעות [] שיהפוך את המטריצה ללא משולשת.

2. את המחלקה של המטריצות האלכסוניות נממש בתור מחלקה יורשת של MtmVec. גודל הווקטור יהיה כגודל מספר השורות/טורים, וכל איבר בוקטור יכיל את האיבר שנמצא באלכסון (שיכול להיות מכל סוג כמובן, int, double וכדומה). מפני שצורתה של המטריצה האלכסונית ידועה ואנחנו יודעים מה מיקומו במטריצה של כל איבר בוקטור, לא נדרש עוד מידע לשמור. לכן, אם מספר השורות במטריצה הוא n , אנחנו נשתמש בזיכרון בסדר גודל של n , במקום בזיכרון בסדר גודל n^2 , בו היינו משתמשים אם היינו ממשים את המחלקה בתור מחלקה יורשת של אחת המטריצות.

3. את המחלקה היורשת של מטריצה דלילה נממש בתור מחלקה יורשת של MtmVec<T>. במקרה זה, גודל הווקטור יהיה כגודל האיברים השונים מאפס במטריצה, T_i יהיה טיפוס שיכול מידע לגבי האיבר, השורה והטור שבו הוא נמצא במטריצה. במחלקה היורשת של המטריצה הדלילה נוסיף גם שדה של מספר השורות והטורים במטריצה. כך נשתמש בזיכרון בסדר גודל קטן יחסית (כמספר האיברים השונים מ-0) במקום בסדר גודל של אלפי שורות וטורים.

4. הבעיה בפעולות בין סוגי משתנים שונים, היא שבשימוש בתבניות (בהן התבקשנו לעשות שימוש בתרגיל), חייבים לבצע תמיד את ההמרות במפורש בין הטיפוסים, מפני שהקומפיילר לעולם לא יבצע המרות אוטומטיות כדי להתאים ארגומנטים לפונקציות של התבנית. זאת בשונה מפונקציות רגילות, עבורן הקומפיילר מבצע המרות אוטומטיות כדי להתאים את המשתנים לפונקציה. גם עבור פונקציות רגילות שימוש מופרז בהמרות אוטומטיות עשוי לגרום לבעיות ולשגיאות בקוד (למשל המרות אוטומטיות עלולות לגרום לדו משמעות ולכישלון בקומפילציה).

עבור מחלקות חייבים לרשום במפורש את הארגומנטים לתבנית ולפונקציות התבנית כי הקומפיילר לא יכול להסיק אותם בעצמו. ולכן אם היינו רוצים לתמוך בפעולות בין משתנים שונים, היינו צריכים לבצע התאמות ספציפיות עבור המרות מפורשות מכל משתנה לכל משתנה.

5. כפי שלמדנו בהרצאות והתרגולים, לשימוש ב-namespace יש מספר יתרונות: ניתן לקבץ בצורה לוגית ונוחה פונקציות וטיפוסים הקשורים זה לזה, ניתן להשתמש בשמות ברורים וקצרים ללא חשש להתנגשויות, וניתן להחליף, לממש ולמזג מימושים שונים בקלות. בתרגיל השתמשנו בטיפוסים הקשורים זה לזה ובפונקציות שהיו שימושיות למחלקות רבות (כמו למשל פונקציות של מחלקת Dimensions או פעולות אלגבריות שונות בין מחלקות שונות כמו מטריצות ווקטורים). בנוסף, אם נרצה להגדיר סוגים נוספים של מטריצות או ווקטורים, יהיה ניתן להתאים אותם בקלות ולהשתמש בהרבה מהפונקציות הקיימות.

6. חשוב לעבוד עם רפרנסים לאובייקטים מסוג מטריצות ווקטורים במהלך התרגיל, מפני שהם טיפוסים מורכבים. אם מעבירים אותם לפונקציות השונות לא כרפרנס אלא by value, היה צריך להעתיק אותם כל פעם כשלא היה צריך לשנותם בפונקציה.