

שפת C++ - תרגיל 3

תאריך הגשה:

הגשה מאוחרת (בהפחתת 10 נקודות):

הנחיות חשובות:

א. בכל התרגילים יש לעמוד בהנחיות הגשת התרגילים וסגנון כתיבת הקוד. שני המסמכים נמצאים באתר הקורס - הניקוד יכלול גם עמידה בדרישות אלו.

ב. אין להגיש קבצים נוספים שאינם קשורים לתרגיל.

ג. עליכם לקמפל עם הדגלים Wextra -Wall -Wvla -c example.cpp -o example.o ולוודא שהתוכנית מתקמפלת ללא אזהרות, תכנית שמתקמפלת עם אזהרות תגרור הורדה בציון התרגיל. למשל, בכדי לקמפל קובץ מקור בשם example.cpp יש להריץ את הפקודה:

```
g++ -Wextra -Wall -Wvla -c example.cpp -o example.o
```

ד. עליכם לוודא שהתרגילים שלכם תקינים ועומדים בכל דרישות הקימפול והריצה במחשבי בית הספר מבוססי מעבדי bit-64 (מחשבי האקווריום, לוי, השרת river). חובה להריץ את התרגיל במחשבי בית הספר לפני ההגשה. (ניתן לוודא שהמחשב עליו אתם עובדים הנו בתצורת bit-64 באמצעות הפקודה "uname -a" ויודא כי הארכיטקטורה היא 64, למשל אם כתוב x86_64).

ה. לאחר ההגשה, בדקו את הפלט המתקבל בקובץ ה-PDF שנוצר מה presubmission script בזמן ההגשה. באם ישנן שגיאות, תקנו אותן על מנת שלא לאבד נקודות.

ו. בדיקת הקוד לפני ההגשה, גם על ידי קריאתו וגם על ידי כתיבת בדיקות אוטומטיות (tests) עבורו היא אחריותכם

חישבו על מקרי קצה לבדיקת הקוד, בדיקות אוטומטיות יהיו חלק מבדיקת הקוד לציון

חלק מקבצי הבדיקה לדוגמה שמסופקים על ידי צוות הקורס נבדקים על ידי הבדיקה

האוטומטית בהגשה (presubmission script). שימוש בקבצי הבדיקה לדוגמה הוא

באחריותכם. במהלך הבדיקה הקוד שלכם ייבדק מול קלטים נוספים לשם מתן הציון.

משימת תכנות – מטריצה

באתר הקורס (בכתובת <http://www.cs.huji.ac.il/~labcpp/ex3/classMatrix.html>) יש תיאור של המחלקה Matrix, ממשו אותה.

הערות:

א. אין להשתמש בניהול זיכרון דינמי (new, delete וכמובן malloc ודומיו), עליכם להשתמש ב STL בתרגיל זה.

ב. בכל מקרה של שגיאה – השתמשו ב exceptions (והוסיפו לתיעוד כל פונקציה איזו שגיאה היא זורקת ובאילו מצבים, תעדו גם אם השגיאה עלולה להיזרק מפונקציה מובנית או מספרייה שאתם משתמשים בה, בחרו את סוגי השגיאות בהיגיון מתוך השגיאות הסטנדרטיות)

ג. אין מגבלת יעילות על הפונקציות (מעבר למימוש הטריטוריאלי), אולם מימוש יעיל של חישוב הדטרמיננטה (אלגוריתם מקבילי או כזה שרץ בזמן n^3 בגודל המטריצה ומטה) יזכה בבונוס. (נא לציין ב README איך מימשתם את הדטרמיננטה אם אתם מעוניינים בנקודות הבונוס)

ד. וודאו שהקובץ testMatrix.cpp מתקמפל ועובד עם המימוש שלכם (אל תגישו אותו, הוא יתווסף אוטומטית בבדיקה).

הקובץ הנ"ל לא בודק את כלל מקרי הקצה (למשל: אין שם שום בדיקה לנכונות של האיטרטורים).

ה. הפעילו שיקול דעת מתי הגיוני שאופרטורים יקראו זה לזה, מתי הגיוני ליצור פונק' נוספת ששני האופרטורים יקראו לה ומתי שווה להשאיר את אותו קוד בכל זאת (רמז: כשזה קוד ממש קצר)

ו. את הטיפוס iterator אתם רשאים לממש בעצמכם, לרשת ממחלקה אחרת, להגדיר כמה טיפוסים למקרים שונים, וכו'.

אתם רשאים להניח שמי שמריץ משתמש ב auto כדי לקבל את טיפוס האיטרטור.

ז. אופרטורים שמוגדרים כ friend – זו המלצה הנתונה למימוש פנימי שלכם, אינכם חייבים להגדיר אותם כ friend במימוש שלכם.

ח. אתם רשאים להוסיף פונקציות ואופרטורים כראות עיניכם.

ט. אינכם נדרשים להגיש makefile, עליכם לוודא שפקודת הקומפילציה הבאה עובדת בלי שגיאות או אזהרות (עם קובץ הבדיקה המסופק):

```
g++ -Wall -Wvla -Wextra -g -O3 -lm -std=c++11 matrixTest.cpp -o matrixTest
```

הגשה

עליכם להגיש קובץ בשם ex3.tar המכיל את הקובץ matrix.h (המחלקה Matrix היא גנרית ולכן צריכה להיות ממומשת בקובץ h) ואת ה README שלכם אם כתבתם אחד. ניתן ליצור קובץ tar כנדרש ע"י הפקודה:

```
tar cvf ex3.tar <list of files to submit>
```

לפני ההגשה, פתחו את הקובץ ex3.tar בתיקיה נפרדת ודאו שהקבצים מתקמפלים ללא שגיאות וללא אזהרות.

מומלץ גם להריץ בדיקות אוטומטיות וטסטרים שכתבתם על הקוד אותו אתם עומדים להגיש.

בתרגילים קודמים הגיעו אלינו מספר קבצים בפורמט שאינו tar, מה שהוביל לבעיות בבדיקה – נשמח אם תקפידו על הגשת קבצי tar.

בנוסף, אתם יכולים להריץ בעצמכם בדיקה אוטומטית עבור סגנון קידוד בעזרת הפקודה:

```
~labcpp/public/codingStyleCheck <file or directory>
```

כאשר <directory or file> מוחלף בשם הקובץ אותו אתם רוצים לבדוק או תיקייה שיבדקו כל הקבצים הנמצאים בה.

דאגו לבדוק לאחר ההגשה את קובץ הפלט (submission.pdf) ודאו שההגשה שלכם עוברת את ה-presubmission script ללא שגיאות או אזהרות.

בהצלחה!