

תכנות מתקדם – 150024

תרגיל בית מספר 3

העמסת אופרטורים

שים לב:

- הקפד על קריאות התכנית ועל עימוד (Indentation).
- הקפד לבצע בדיוק את הנדרש בכל שאלה.
- בכל אחת מהשאלות יש להגדיר כל מחלקה ב 2 קבצים נפרדים – קובץ h. וקובץ cpp.
- בכל אחת מהשאלות יש להגדיר פונקציות עזר במידת הצורך עבור קריאות התכנית.
- יש להגיש את התרגיל על פי ההנחיות להגשת תרגילים (המופיע באתר הקורס) וביניהם: השתמש בשמות משמעותיים עבור המשתנים.
- יש לתעד את התכנית גם עבור פונקציות אותם הנך מגדיר וכן על תנאים ולולאות וקטעי קוד מורכבים, ובנוסף, **דוגמת הרצה לכל תכנית בסוף הקובץ!**

הערה חשובה: לתרגיל בית הזה מוגדר שבוע להגשה, אלא אם כן קיבלת הוראה אחרת מהמרצה שלך. תיבות ההגשה הפתוחות לא מהוות היתר להגשה באיחור.

שאלה מס' 1:

שים לב: מתרגיל זה ואילך יש להקפיד להגדיר const במקומות המתאימים. ובמידת האפשר להעביר פרמטרים by reference עם הגשה const.

חלק א:

- בתרגיל זה נמשיך לפתוח ולשפר את המחלקה Rational (המספרים הרציונליים) שהתחלת להגדיר בתרגיל בית 1 שאלה 2.
- העתק את המחלקה Rational כפי שכבר הגדרת ובצע בה את השינויים הבאים:
- מחק מכל הבנאים את שורת ההדפסה – מתרגיל 3 אין צורך בהדפסה בתוך הבנאים.
 - מחק את מתודות הבאות:
 - i. equal
 - ii. makeEquals
 - עדכן את המתודה print כך שבהדפסת המספר, אם במכנה שמור הוא 1 אז יש להדפיס רק את המונה.
 - תיקון טכני לתרגיל 1 שאלה 2 - החלף את שם המתודה getDenominator להיות getMechane
 - הוסף מתודה פרטית (private) reduce למצמ את השבר

חלק ב:

עליך להוסיף למחלקה את האופרטורים הבאים:

- = - הצבה ו-move assignment
- +, -, *, / - פעולות חשבון לפי כללי מספרים רציונאליים
- +, -, *, /=
- !=, ==, <=, >, < (שימו לב – ההשוואה צריכה להיות מתמטית לדוגמא $1/2 == 3/6$).
- ++, -- תחילי וסופי

הערות:

- תוצאות פעולות חשבון מחזירות את התוצאה לאחר צימצום לדוגמא $1/3 + 2/4 = 5/6$
- בפעולה של חילוק – מותר להניח שלא תתבקשו לחלק מספר עם 0. אין צורך לבדוק תקינות הקלט.

- נכתוב פתרון שלא משתמש ב"שכפול קוד". בגלל שהבדיקה של אי שוויון היא היפוך של בדיקת שוויון נשתמש במתודה של בדיקת שוויון שכבר כתבנו. כנ"ל לגבי בדיקת גדול ישתמש בבדיקה של קטן.

חלק ג:

נתון במודל קובץ בשם RationalMain.cpp המכיל תוכנית הראשית (main) שמטרתה לבדוק נכונות המחלקה שכתבת, הורד את הקובץ ובדוק את תקינות התוכנית שלך (יש להגיש את התוכנית הראשית ביחד עם הקוד שכתבת).

שאלה מס' 2:

1. הגדר/י מחלקה MyDate עבור ייצוג תאריך.
המחלקה תכלול את התכונות הבאות:

- יום (int)
- חודש (int)
- שנה (int)

הוסף למחלקה את המתודות הבאות (– שים לב - שחוץ מהמתודה print אין להדפיס שום דבר מתוך מתודות חוץ מאשר ERROR וכן אין להדפיס ERROR מה-main)

- constructor – בנאי המקבל כפרמטרים ערכים עבור התכונות: יום, חודש, שנה, עם ערכי ברירת מחדל - 1/1/1899 (כלומר, במידה ולא התקבל ערך עבור אחד או יותר מהשדות, הפרמטר המתאים יקבל את ערך ברירת המחדל התואם לו). על הבנאי לאתחל את התכונות בערכים שהתקבלו, אך לפני עליו לבצע בדיקת תקינות לתאריך. להלן רשימה של החודשים ומספר הימים שבכל חודש:

מספר חודש	שם חודש	מספר ימים
1	ינואר	31
2	פברואר	28 (29)
3	מרץ	31
4	אפריל	30
5	מאי	31
6	יוני	30
7	יולי	31
8	אוגוסט	31
9	ספטמבר	30
10	אוקטובר	31
11	נובמבר	30
12	דצמבר	31

מספר הימים בחודש פברואר בשנה מעוברת מסומן בסוגריים.
שנה היא מעוברת אם היא מתחלקת ב- 4 אך אינה מתחלקת ב- 100, או שהיא מתחלקת ב- 400.
הקלט לתוכנית צריך להיות בפורמט הבא dd/mm/yyyy כאשר dd, mm, yyyy - הם מספרים שלמים וחייביים המייצגים יום, חודש ושנה בהתאמה.
הטווחים המותרים הם:
ימים: $1 \leq dd \leq 31$
חודשים: $1 \leq mm \leq 12$
שנים: $1899 \leq yyyy \leq 9999$
שימו לב כי אין זה מספיק שהמספרים נמצאים בטווחים המותרים על מנת שהתאריך יהיה תקין.

לדוגמא: 2000 4 31 - אינו תאריך תקין מכיוון שבאפריל יש 30 ימים.
בכל מקרה של קלט שגוי (לא משנה באיזה מהשדות) יש להדפיס הודעה "Error date" ולאתחל את כל השדות בערכי ברירת מחדל: היום ב 1, החודש ב 1, והשנה ב 1899.

- copy constructor – בנאי העתקה.
- move constructor – בנאי הזזה
- מתודה setDate המעדכנת את התאריך. על המתודה לקבל כפרמטרים ערכים עבור כל התכונות. במידה והערכים תואמים את הטווחים התקינים (כאמור במתודה הבונה), על המתודה להציב את הערכים החדשים בתכונות. במידה ואחד הערכים אינו תואם לטווח המתאים לעיל, לא יתבצע כל שינוי בכל התכונות, ללא שום פלט.
- מתודה print אשר מדפיסה את התאריך בפורמט: dd/mm/yyyy (כאשר היום והחודש בני ספרה אחת או שתיים, והשנה בת ארבע ספרות).
- operator= . המתודה עובדת כמו האופרטור = הטבעי. מעתיקה את ערכי התכונות של האובייקט מצד ימין של הסימן שווה לתכונות של האובייקט בצד שמאל של הסימן שווה.
- move assignment
- ++ operator תחילי. המתודה משנה את התאריך למחרתו (כלומר - לתאריך של יום למחרת), ומעדכנת את התכונות בהתאם. (שימו לב שבסוף החודש, יום המחרת הוא היום הראשון של החודש הבא, ובסוף השנה – יום למחרת הוא היום הראשון של השנה הבאה לאחריה.)
- ++ operator סופי. המתודה משנה את התאריך למחרתו (כלומר - לתאריך של יום למחרת), ומעדכנת את התכונות בהתאם. (שימו לב גם כאן לתאריכים שבסוף החודש יום המחרת הוא היום הראשון של החודש הבא, ובסוף השנה – יום למחרת הוא היום הראשון של השנה הבאה לאחריה.)
- operator< - מחזיר true במידה והתאריך השמאלי קודם לתאריך הימני.
- operator> - מחזיר true במידה והתאריך השמאלי מאוחר יותר מתאריך הימני.
- operator== - מחזיר true אם התאריכים זהים.

כתוי תכנית ראשית אשר תקלוט מהמשתמש תאריך ("Enter a date") (שים לב לפורמט הקלט). על התכנית לאתחל אובייקט שהוא מופע של המחלקה MyDate בתאריך שהתקבל. התכנית תדפיס את התאריך (באמצעות המתודה שהגדרת).
לאחר מכן, בלולאה – התכנית תדפיס: "Enter a code", תקלוט מהמשתמש קוד פעולה, תבצע את הפעולה המתאימה לקוד בהתאם לרשום להלן, ואז תדפיס את ערכו של האובייקט.

שים לב! כדי שתוצאות הפעלת האופרטור ++ יודפסו – יש לבצע השמה של תוצאת הפעולה לתוך משתנה חדש. ולאחר מכן להדפיס את ערכו של המשתנה החדש.
לדוגמא עבור הפעולה של ++ תחילי יש לבצע:

```
d1 = ++d2;
d1.print();
cout << endl;
d2.print();
cout << endl;
```

ועבור הפעולה של ++ סופי גם יש לבצע:

```
d1 = d2++;
d1.print();
cout << endl;
d2.print();
cout << endl;
```

התכנית תסתיים כאשר ייקלט הקוד 0 (אפס).
קודי הפעולה:

- 1 - עדכון תאריך. התכנית תדפיס **Enter a date**, ותקלוט תאריך חדש מהמשתמש. התכנית תפעיל את המתודה `setDate`. לאחר מכן התכנית תדפיס את התאריך המעודכן.
- 2 - התכנית תדפיס את פעולת האופרטור `++` תחילי. (ראו דוגמא)
- 3 - התכנית תדפיס את פעולת האופרטור `++` סופי. (ראו דוגמא)
- 4 - התכנית תדפיס **Enter a date**, ותקלוט מהמשתמש תאריך לתוך עצם חדש. לאחר מכן, התכנית תפעיל את פעולת ההשמה (אופרטור `=`) - ותציב את העצם החדש בתוך העצם הישן (העצם החדש יופיע מצד ימין של אופרטור ההשמה והעצם הישן יופיע מצד שמאל של אופרטור ההשמה לאחר כך יודפס העצם הישן (עם הערכים החדשים שבו, כמובן))
- 5 - התוכנית תדפיס **Enter a date**, ותקלוט תאריך מהמשתמש לעצם חדש. התוכנית תדפיס את פעולת האופרטור `>` בין התאריך החדש לתאריך הישן (התאריך הישן משמאל לאופרטור) ואז נקודתיים ואז תוצאת הפעולה (`false` או `true`).
- 6 - התוכנית תדפיס **Enter a date**, ותקלוט תאריך חדש מהמשתמש. התוכנית תדפיס את פעולת האופרטור `<` בין התאריך החדש לתאריך הישן (התאריך הישן משמאל לאופרטור) ואז נקודתיים ואז תוצאת הפעולה (`false` או `true`).
- 7 - התוכנית תדפיס **Enter a date**, ותקלוט תאריך חדש מהמשתמש. התוכנית תדפיס את פעולת האופרטור `==` בין התאריך החדש לתאריך הישן (התאריך הישן משמאל לאופרטור) ואז נקודתיים ואז תוצאת הפעולה (`false` או `true`).
- 0 - התוכנית תפסיק את ביצוע הלולאה, ותסתיים.

דוגמא להרצת התוכנית:

```
Enter a date
-5/1/2012
Error date
1/1/1899
Enter a code
1
Enter a date
5/7/2010
5/7/2010
Enter a code
2
6/7/2010
6/7/2010
Enter a code
3
6/7/2010
7/7/2010
Enter a code
4
Enter a date
14/7/2010
14/7/2010
Enter a code
5
Enter a date
14/7/2010
14/7/2010 > 14/7/2010 : false
Enter a code
6
Enter a date
15/7/2010
14/7/2010 > 15/7/2010 : false

Enter a code
7
Enter a date
14/7/2010
14/7/2010 == 14/7/2010 : true
Enter a code
0
```

בהצלחה רבה!