# 150018 – C++ -סדנא ב

### תרגיל בית מספר 3

#### העמסת אופרטורים

### <u>שים/י לב:</u>

- א. הקפד/י על קָרִיאוּת התכנית ועל עימוד (Indentation).
  - ב. הקפד/י לבצע בדיוק את הנדרש בכל שאלה.
- ג. בכל אחת מהשאלות יש להגדיר פונקציות במידת הצורך עבור קריאות התכנית.
- ד. יש להגיש את התרגיל על פי ההנחיות להגשת תרגילים (המופיע באתר הקורס) וביניהם: השתמש/י בשמות משמעותיים עבור המשתנים.

יש לתעד את התכנית גם עבור פונקציות אותם הנך מגדיר/ה וכן על תנאים ולולאות וקטעי קוד מורכבים, ובנוסף, **דוגמת הרצה לכל תכנית בסוף הקובץ!** הגשה יחידנית - אין להגיש בזוגות.

**הערה חשובה:** לכל תרגיל בית מוגדר <u>שבוע אחד בלבד</u> להגשה, אלא אם כן קיבלת הוראה אחרת מהמרצה שלך. <u>תיבות ההגשה הפתוחות לא מהוות היתר להגשה באיחור.</u>

#### שאלה מס' 1:

## שים/י לב: מתרגיל זה ואילך יש להקפיד להגדיר const במקומות המתאימים!!

השלם/י את הגדרת המחלקה Rational שהוצגה במעבדה (ניתן להיעזר במחלקה שכתבת בתרגיל בית 1), כך שתאפשר להשתמש במשתנה רציונאלי באופן טבעי.

עליך להוסיף למחלקה את האופרטורים:

+(חיבור),-(חיסור),\*(כפל),/(חילוק),++תחילי,++סופי,--תחילי,--סופי,=<,=>,<,>,==,=! במקרה בו נקלט ערך 0 עבור המכנה יש להודיע למשתמש <mark>"ERROR"</mark>, ולהציב במכנה 1.

כאשר המכנה שווה ל- 1, או כאשר המונה שווה ל- 0. בהדפסת המספר יש להדפיס רק את המונה. יש לצמצם את השבר לפני הדפסתו.

השתמש/י במחלקה שיצרת והרץ<sup>'</sup>וי תכנית ראשית אשר תקלוט מהמשתמש שני מספרים רציונאליים (<mark>enter two rational numbers:</mark>) בפורמט של מונה/מכנה. על התכנית לחשב ולהדפיס את התוצאות של כל אחת מהפעולות שלהלן.

# יש להשתמש ב-main המצורף עבור הרצת התכנית:

```
סדנא ב- +++ – 150018 – תשפ"א
                                                                                                      בס"ד
5 מתוך 2 מתוך 2 מתוך 5 מתוך
                   case add:r1.print(); cout << " + "; r2.print(); cout << " = "; (r1 +</pre>
r2).print(); cout << endl;</pre>
                   case sub:r1.print(); cout << " - "; r2.print(); cout << " = "; (r1 -</pre>
r2).print(); cout << endl;</pre>
                   case mult:r1.print(); cout << " * "; r2.print(); cout << " = "; (r1 *</pre>
r2).print(); cout << endl;</pre>
                   case divi:r1.print(); cout << " / "; r2.print(); cout << " = "; (r1 /</pre>
r2).print(); cout << endl;</pre>
                   case addPre:cout << "++"; cout << "("; r1.print(); cout << ") = ";</pre>
(++r1).print(); cout << endl;
case addPost: cout << "("; r1.print(); cout << ")++"; cout << " = ";
(r1++).print(); cout << endl;</pre>
                   case subPre:cout << "--"; cout << "("; r1.print(); cout << ") = "; (--</pre>
r1).print(); cout << endl;
case subPost:cout << "("; r1.print(); cout << ")--"; cout << " = "; (r1-).print(); cout << endl;
                  case big:r1.print(); cout << " > "; r2.print(); cout << " ? "; (r1 > r2) ?
cout << "no"; cout << endl;</pre>
cout << "yes" : cout <<
                   case small:r1.print(); cout << " < "; r2.print(); cout << " ? "; (r1 < r2) ?</pre>
cout << "yes" : cout << "no"; cout << endl;</pre>
case bigEq:r1.print(); cout << " >= "; r2.print(); cout << " ? "; (r1 >= r2) ? cout << "yes" : cout << "no"; cout << endl;
case smallEq:r1.print(); cout << " <= "; r2.print(); cout << " ? "; (r1 <= r2) ? cout << "yes" : cout << "no"; cout << endl;</pre>
                   case eequal:r1.print(); cout << " == "; r2.print(); cout << " ? "; (r1 == r2)</pre>
? cout << "yes" : cout << "no"; cout << endl;</pre>
case notEqual:r1.print(); cout << " != "; r2.print(); cout << " ? "; (r1 != r2) ? cout << "yes" : cout << "no"; cout << endl;</pre>
                   cout << "enter 1 to cont. 0 to exit." << endl;
         }
         return 0;}
                                                                                 דוגמא להרצת התכנית:
```

[YS1] עם הערות: לשנות את התכנית הראשית, כך שתקבל פלטים ותנסה כמה דברים, ולא רצף אחד כמו פה. (צריך לעשות switch) אפשר לתת תכנית ראשית

```
enter two rational numbers:
1/2 3/4
1/2 + 3/4 = 5/4
1/2 - 3/4 = -1/4
1/2 * 3/4 = 3/8
1/2 / 3/4 = 2/3
++(1/2) = 3/2
(3/2)++=3/2
 -(5/2) = 3/2
 3/2)--= 3/2
1/2 > 3/4? no
1/2 < 3/4? yes
1/2 >= 3/4? no
1/2 <= 3/4 ? yes
1/2 == 3/4 ? no
1/2 != 3/4 ? yes
enter 1 to cont. 0 to exit.
0
```

#### :2 'שאלה מס'

- 1. הגדר/י מחלקה עבור ייצוג תאריך. המחלקה תכלול את התכונות הבאות:
  - יום (int) •
  - (int) חודשׁ •
  - שנה (int)

ולכל הפחות. את המתודות הבאות:

- בנאי המקבל כפרמטרים ערכים עבור התכונות: יום, חודש, שנה, עם ערכי ברירת מחדל 1/1/1920 (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 הנתונים באופן הבא:
  - עבור התכונה יום: במידה והתקבל מספר שאינו בין 1-30, תודפס ההודעה: "Error day", והתכונה יום תקבל את הערך 1.
  - עבור התכונה חודש: במידה והתקבל מספר שאינו בין 1-12, תודפס ההודעה: "Error month", והתכונה חודש תקבל את הערך 1.
    - עבור התכונה שנה: במידה והתקבל מספר שאינו בין 1920-2099, תודפס ההודעה: <mark>"Error year"</mark>, והתכונה חודש תקבל את הערך 1920.
    - כמובן ייתכן ותודפסנה כמה הודעות שגיאה עבור תאריך אחד שהתקבל.
      - 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.
- ++ operator **תחילי**. המתודה משנה את התאריך למחרתו (כלומר לתאריך של יום למחרת), ומעדכנת את התכונות בהתאם. במידה והשנה לאחר השינוי חורגת מ-2099, יש להכניס לכל התכונות את הערכים 1/1/1/1920.
- ++ operator סופי. המתודה משנה את התאריך למחרתו (כלומר לתאריך של יום למחרת), ומעדכנת את התכונות בהתאם. במידה והשנה לאחר השינוי חורגת מ-2099, יש להכניס לכל התכונות את הערכים 1/1/1920.
- =+ operator. המתודה משנה את התאריך, לתאריך שיחול כעבור מספר הימים שמימין לאופרטור, ומעדכנת את התכונות בהתאם. במידה והשנה לאחר השינוי חורגת מ-2099, יש להכניס לכל התכונות את הערכים 1/1/1920.
  - סperator > המתודה בודקת קדימות בין שני תאריכים. במידה והתאריך שמשמאל לאופרטור מאוחר יותר מהתאריך שמימין לאופרטור, המתודה תחזיר אמת, אחרת המתודה תחזיר שקר.
  - סרפרוליה בודקת קדימות בין שני תאריכים. במידה והתאריך שמשמאל לאופרטור מוקדם יותר מהתאריך שמימין לאופרטור, המתודה תחזיר אמת, אחרת המתודה תחזיר שקר.
    - .operator== •
  - מתודה אשר מדפיסה את התאריך בפורמט: dd/mm/yyyy (כאשר היום והחודש בני ספרה אחת או שתיים, והשנה בת ארבע ספרות).

עבור ערכים בוליאניים יש להדפיס true או false. הערה: יש להניח כי כל החודשים הם בני 30 ימים.

כתוב∕י תכנית ראשית אשר תקלוט מהמשתמש תאריך (<mark>"Enter a date") (שים לב לפורמט הקלט</mark>). <mark>על התכנית לאתחל</mark> אובייקט שהוא מופע של המחלקה Date בתאריך שהתקבל. התכנית תדפיס את התאריך (באמצעות המתודה שהגדרת). לאחר מכן, בלולאה – התכנית תדפיס:"What do you want to do", תקלוט מהמשתמש קוד פעולה, תבצע את הפעולה המתאימה לקוד בהתאם לרשום להלן.

[YS2] עם הערות: להוסיף בנאי הזזה. נבקש הדפסות מכל הבנאים. שימי לב שזה מחייב אם יקבלו ויחזירו מהאופרטור עם & ובלי וכן הלאה.

**[YS3] עם הערות:** אני רוצה שיקלטו ויאתחלו, ולא שיבנו עצם ואחכ יעשו לו set (נוהג פטול שקיים) ולכן ההדפסה מהבנאי צריכה להיות נכונה – עם הערכים וכו. הייתי מקפידה על const

```
בס"ד
```

```
סדנא ב- ++C – 150018 – תשפ"א
5 תרגיל בית מספר 3 - עמוד 4 מתוך
```

שים/י לב! יש לבצע את פעולת האופרטור ואת ההדפסה של התאריך בפעולה אחת (כגון: (dateForExample++.print();

התכנית תסתיים כאשר ייקלט הקוד 1- (מינוס אחד). קודי הפעולה:

- 1 עדכון תאריך. התכנית תדפיס <mark>Enter a date,</mark> ותקלוט תאריך חדש מהמשתמש. התכנית תפעיל את המתודה setDate. לאחר מכן התכנית תדפיס את התאריך המעודכן.
  - 2 התכנית תדפיס את פעולת האופרטור ++ תחילי.
  - 3 התכנית תדפיס את פעולת האופרטור ++ סופי.
- 4 התכנית תדפיס <mark>Enter # days , ותקלוט מספר מהמשתמש . התכנית תדפיס את פעולת באופרנון =+ ושל התאריר עם המספר שנקלו</mark>
- תדפיס את פעולת האופרטור =+ של התאריך עם המספר שנקלט. 5 - התכנית תדפיס Enter a date, ותקלוט תאריך מהמשתמש עבור עצם חדש. התכנית תדפיס את פעולת האופרטור < (העצם הישן משמאל לאופרטור).
  - 6 התכנית תרפיס <mark>Enter a date , ותקלוט תאריך חדש מהמשתמש.</mark> התכנית תדפיס את פעולת האופרטור > (התאריך הישן משמאל לאופרטור).
  - 7 התכנית תׄדפיס <mark>Enter a date , ותקלוט תאריך חדש מהמשתמש.</mark> התכנית תדפיס את פעולת האופרטור == (התאריך הישן משמאל לאופרטור).
    - 1- התכנית תפסיק את ביצוע הלולאה, ותסתיים. דוגמא להרצת התכנית: (אין להדפיס את ההערות אלא רק את הערכים)

```
Enter a date
קלט מהמשתמש//<mark>5/1/2012.</mark>
Error day
1/1/2012הדפסת הערך שהוצב בתוך התאריך/
What do you want to do
Enter a date
קלט מהמשתמש/<mark>5/7/2010</mark>
הדפסת התאריך אחרי השינוי//<mark>5/7/2010</mark>
What do you want to do
הדפסת אופרטור ++ תחילי על התאריך/<mark>/6/7/2010</mark>
What do you want to do
הדפסת אופרטור ++ סופי על התאריך/<mark>/6/7/2010</mark>
What do you want to do
Enter # days
הדפסת הוספת 7 ימים לתאריך<mark>//14/7/2010</mark>
What do you want to do
Enter a date
14/7/2010
>: false
What do you want to do
Enter a date
14/7/2010
==: true
What do you want to do
```

### <u>שאלה מס' 3:</u>

השלם/י את הגדרת המחלקה MyString שהוצגה בהרצאה (ניתן להיעזר בדוגמאות המופיעות ה-github), כך שתאפשר להשתמש במחרוזות באופן טבעי.

עליך להוסיף למחלקה לפחות את המתודות הבאות:

.. א. אופרטורים: >,<,=,>=, (ע"פ שוויון לקסיקוגרפי).

ב. אופרטור [] עבור הצבה ואיחזור. במקרה של חריגה יש להדפיס הודעת ERROR ג. מתודה בשם insert המקבלת מחרוזת str ומיקום index. המתודה אינה משנה את המחרוזת המקורית ומחזירה מחרוזת חדשה המהווה הכנסה של str למחרוזת המקורית החל ממקום index. במקרה בו index אינו תקין כך שלא ניתן לבצע את ההכנסה הרצויה, תודפס ההודעה: ERROR ותוחזר מחרוזת ריקה. הצהרת המתודה הינה:

MyString insert( int index, const char\* str )

כתוב/י תכנית ראשית אשר תקלוט שתי מחרוזות, a,b כל אחת בשורה נפרדת, וכן מספר ח, על התוכנית לבצע את הפעולות הבאות:

- או <mark>a=b או a>b א. להדפיס הודעה המייצגת את היחס בין המחרוזות (לקסיקוגרפית):</mark>
- ב. ואת המחרוזת המתקבלת כאשר מפע<sup>י</sup>לים את המתודה insert, כך שהמחרוזת a, משובצת במחרוזת b, החל מהמיקום ח. במידה והקלט אינו תקין (כך שלא ניתן להפעיל את המתודה insert), תודפס ההודעה: <mark>ERROR</mark> ותוחזר מחרוזת ריקה.
  - ג. בנוסף על התכנית לֹקלוט תו כלשהו וערך index ולשנות את התו במיקום ה-mdex במחרוזת החדשה להיות התו שנקלט (יש להשתמש באופרטור [ ]) ולהדפיס את המחרוזת המעודכנת

דוגמאות להרצת התכנית:

| World               | Hello          | Hello   | Hello<br>Hello                             | Hello<br>Hello    |
|---------------------|----------------|---|--|-------------------|
| <mark>Hello</mark>  | Hello          | World   | World                                      | World             |
| <mark>5</mark>      | <mark>2</mark> | <mark>2</mark>  | <mark>0</mark>                             | <mark>8</mark>    |
| <mark>a&gt;b</mark> | a=b            | a <b< td=""><td>a<b< td=""><td>a<b< td=""></b<></td></b<></td></b<> | a <b< td=""><td>a<b< td=""></b<></td></b<> | a <b< td=""></b<> |
| HelloWorld          | HeHellollo     | WoHellorld  | HelloWorld                                 | ERROR             |
| <u>!</u>            | <mark>h</mark> | <mark>?</mark>  | <mark>-</mark>                             | J                 |
| <mark>5</mark>      | <mark>2</mark> | <mark>10</mark>   | <mark>6</mark>                             | <mark>3</mark>    |
| Hello!orld          | Hehellollo     | ERROR   | HelloW-rld                                 | ERROR             |
|                     |                |   |  |                   |

בהצלחה רבה!

[**YS4] עם הערות:** קוראים לה mystring הזה בכוונה – כדי שיבחינו בינה לבין string.h של STL ושל א

**(YS5) עם הערות:** בואי נשנה את זה קצת. אפילו לשנות את הסדר בין הפעולות.