

תרגיל בית 4 – C++

מועד ההגשה: האחראי על התרגיל:	יום ד', 14/06/2017 בשעה 23:59 שיראל יוזף	shirel@campus
----------------------------------	---	---------------

נא לקרוא היטב את כל התרגיל, כולל הקבצים המצורפים, לפני תחילת העבודה.
בתרגיל בית זה, ניצור מסד נתונים בו נאכסן נתונים של הסטודנטים, כלומר
מקצעות שהם למדו וציוני מעקב. יש לממש את המחלקות הבאות:

1. Course

מחלקה זו מתארת מקצוע בסיס ומחזיקה נתונים הבאים:

- מספר הקורס (מספר שלם). הניחו שלכל קורס יש מספר קורס ייחודי.
- שם הקורס (מחרוזת)
- מספר תרגילי בית להגשה (מספר שלם). ספירת תרגילי הבית מתחילה מ-0.
- משקל תרגילי הבית בציון הסופי (מספר בין 0 ו-1).
- ציון המבחן (מספר שלם)
- ציון בכל אחד מתרגילי הבית (מספר שלם)

ממשק המחלקה תומך במתודות הבאות:

- constructor – מקבל מספר הקורס, שם הקורס, מספר תרגילי הבית ומשקל תרגילי הבית.
- destructor – משחרר את כל הזיכרון דינאמי שהוקצה עבור Course
- getNum – מחזירה את מספר הקורס (מספר שלם)
- getName – מחזירה מצביע לעותק של שם הקורס (מחרוזת)
- getExamGrade – מחזירה את ציון המבחן (מספר שלם)
- getHwGrade – מקבלת את מספר סידורי של תרגיל הבית (מספר שלם) ומחזירה את ציון התרגיל (מספר שלם)
- getHwNum – מחזירה את מספר תרגילי הבית להגשה (מספר שלם)
- getHwWeigh – מחזירה את המשקל של תרגילי הבית בציון הסופי (מספר לא שלם)
- getHwAverage – מחזירה ממוצע של תרגילי הבית (מספר לא שלם)
- getCourseGrade – מחזירה ציון סופי בקורס (מספר שלם). הציון מחושב לפי נוסחה הבאה:
$$\text{course_grade} = \text{round}((1 - \text{hw_weigh}) * \text{exam_grade} + \text{hw_weigh} * \text{hw_average})$$
- setExamGrade – מקבלת את ציון המבחן (מספר שלם) ומוסיפה אותו לקורס. מחזירה true במקרה של הצלחה ו- false במקרה של כישלון.
- setHwGrade – מקבלת מספר סידורי של תרגיל בית (מספר שלם) וציון (מספר שלם) ומכניסה את הציון לקורס. מחזירה true במקרה של הצלחה ו- false במקרה של כישלון.

2. EE_Course

מחלקה זו מתארת מקצוע בפקולטה להנדסת חשמל. בנוסף לתכונות של מקצוע בסיסי, ב-EE_Course ניתן להוסיף פקטור לציון הסופי. בנוסף לנתונים של הקורס הבסיסי, המחלקה מחזיקה את הנתון הבא:

הטכניון - מכון טכנולוגי לישראל

הפקולטה להנדסת חשמל
מבוא למערכות תוכנה

– פקטור (מספר שלם). מאותחל ל-0 ע"י ה-constructor

בנוסף למתודות של Course, ממשק המחלקה תומך במתודות הבאות:

- `getFactor` – מחזירה את ערכו של הפקטור (מספר שלם)
- `setFactor` – מקבלת את ערכו של הפקטור (מספר שלם) ומוסיפה אותו לקורס. מחזירה ערך בוליאני `true`.

להבדיל ממחלקת Course, הציון הסופי של `EE_Course` מחושב ע"י הנוסחה הבאה:

```
course_grade = round( (1 - hw_weight)*exam_grade +  
                    hw_weight*hw_average) + factor
```

אם לאחר הוספת פקטור מתקבל ציון מעל 100, הציון הסופי הוא 100

3. CS_Course

מחלקה זו מתארת מקצוע בפקולטה למדעי המחשב. בנוסף לתכונות של מקצוע בסיסי, ב-`CS_Course` תרגילי הבית יכולים להיות "מגן" או "תקף". כמו כן ניתן להוסיף שם של ספר הקורס. בנוסף לנתונים של הקורס הבסיסי, המחלקה מחזיקה את הנתון הבא:

– דגל שאומר האם ציוני תרגילי הבית הם "מגן" או "תקף". ניתן כפרמטר ל-

constructor

– שם של ספר הקורס (מחרוזת). ניתן כפרמטר ל-constructor

בנוסף למתודות של Course, ממשק המחלקה תומך במתודות הבאות:

- `isTakef` – מחזירה `true` אם תרגילי הבית הם "תקף" ו-`false` אם תרגילי הבית הם "מגן".
- `getBook` – מחזירה מצביע לעותק של שם של ספר הקורס
- `setTakef` – מקבלת ערך בינארי (`true` או `false`) ומעדכנת את הדגל המתאים של הקורס. מחזירה `true`.
- `setBook` – מקבלת מצביע לשם של ספר הקורס ומעתיקה את שם הספר לתוך הקורס.

להבדיל ממחלקת Course, הציון הסופי שן `CS_Course` מחושב ע"י הנוסחות הבאות.

אם תרגילי הבית הם "תקף":

```
course_grade = round( (1 - hw_weight)*exam_grade + hw_weight*hw_average)
```

אם תרגילי הבית הם "מגן":

```
course_grade = max( round( (1 - hw_weight)*exam_grade +  
                    hw_weight*hw_average), exam_grade)
```

4. Person

מחלקה זו מתארת אדם ומחזיקה נתונים הבאים:

- מספר זיהוי של אדם (`integer`). תניחו שלכל אדם יש מספר זיהוי ייחודי.
- שם האדם (מחרוזת)

ממשק המחלקה תומך במתודות הבאות:

- `constructor` – מקבל שם האדם (מחרוזת) ומספר זיהוי שלו (`integer`)

הטכניון - מכון טכנולוגי לישראל

הפקולטה להנדסת חשמל
מבוא למערכות תוכנה

- destructor – משחרר כל זיכרון הדינאמי שהוקצע עבור Person
- getID – מחזירה את מספר הזהוי של אדם
- getName – מחזירה מצביע לעותק של שם האדם

5. Student

מחלקה זו מתארת סטודנט. בנוסף למאפיינים של Person, המחלקה מחזיקה נתונים הבאים:

- מערך של מצביעים לקורסים מהנדסת חשמל (EE_Course) שנלמדו ע"י סטודנט. מספר הקורסים חסום ע"י MAX_COURSE_NUM שהוגדר בקובץ Proj.h

- מספר הקורסים מהנדסת חשמל שנלמדו ע"י סטודנט (מספר שלם)
- מערך של מצביעים לקורסים ממדעי מחשב (CS_Course) שנלמדו ע"י סטודנט. מספר הקורסים חסום ע"י MAX_COURSE_NUM שהוגדר בקובץ Proj.h
- מספר הקורסים ממדעי מחשב שנלמדו ע"י סטודנט (מספר שלם)

בנוסף למתודות של Person, ממשחק המחלקה תומך במתודות הבאות:

- getCourseCnt – מחזירה את סך כל הקורסים שנלמדו ע"י הסטודנט בשתי הפקולטות.
- addEE_Course – מקבלת מצביע ל-EE_Course ומוסיפה את הקורס לסטודנט. מחזירה true במקרה של הצלחה ו-false במקרה של כישלון.
- addCS_Course – מקבלת מצביע ל-CS_Course ומוסיפה את הקורס לסטודנט. מחזירה true במקרה של הצלחה ו-false במקרה של כישלון.
- remCourse – מקבלת את מספר הקורס שנלמד ע"י סטודנט (EE_Course או CS_Course) ומסירה אותו (משחררת את הזיכרון שהוא תופס). מחזירה true במקרה של הצלחה ו-false במקרה של כישלון.
- getEE_Course – מקבלת מספר של EE_Course שנלמד ע"י הסטודנט ומחזירה מצביע לקורס. אם הקורס לא נלמד ע"י הסטודנט המתודה מחזירה NULL
- getCS_Course – מקבלת מספר של CS_Course שנלמד ע"י הסטודנט ומחזירה מצביע לקורס. אם הקורס לא נלמד ע"י הסטודנט המתודה מחזירה NULL
- getAvg – מחזירה ממוצע של ציוני הסטודנט (מספר שלם). ממוצע מחושב ע"י נוסחה הבאה:
$$\text{average_grade} = \text{round}(\text{sum_of_grades_in_all_courses} / \text{number_of_courses})$$
- print – מדפיסה את נתיני הסטודנט לפי פורמט הבא:

Student Name: *student_name*

Student ID: *student_id*

Average Grade: *average_grade*

EE Courses:

course_number course_name course_grade

.

.

CS Courses:

course_number course_name course_grade

את פורמט ההדפסה המדויק ניתן לראות בקובץ פלט המצורף.

6. StArray

מחלקה זו מתארת אוסף של סטודנטים ומחזיקה נתונים הבאים:

- מערך של מצביעים לסטודנטים. מספר הסטודנטים חסום ע"י MAX_STUDENT_NUM שהוגדר בקובץ Proj.h.
- מספר הסטודנטים שנמצאים כרגע באוסף (מספר שלם)

ממשק המחלקה תומך במתודות הבאות:

- constructor – מאתחל את כל המצביעים ל NULL וכן את מספר הסטודנטים ל-0.

- destructor – משחרר את כל זיכרון הדינאמי שנתפס ע"י הסטודנטים.

- addStudent – מקבלת מספר זיהוי של סטודנט (integer) ושם הסטודנט (מחרוזת). מייצרת סטודנט חדש ומוסיפה אותו למערך. מחזירה true במקרה של הצלחה ו- false במקרה של כישלון.

- addEE_Course – מקבלת פרמטרים הבאים:

- מספר הזיהוי של סטודנט
- מספר הקורס
- שם הקורס
- מספר תרגילי בית להגשה
- משקל ציוני תרגילי הבית בחישוב ציון הסופי

מייצרת EE_Course חדש ומוסיפה אותו לסטודנט המתאים. מחזירה true במקרה של הצלחה ו- false במקרה של כישלון. נכשלת אם הסטודנט המבוקש לא קיים במערכת.

- addCS_Course – מקבלת פרמטרים הבאים:

- מספר הזיהוי של סטודנט
- מספר הקורס
- שם הקורס
- מספר תרגילי בית להגשה
- משקל ציוני תרגילי הבית בחישוב ציון הסופי
- דגל "תקף" או "מגן"
- שם ספר הקורס

הטכניון - מכון טכנולוגי לישראל

הפקולטה להנדסת חשמל
מבוא למערכות תוכנה

מייצרת CS_Course חדש ומוסיפה אותו לסטודנט המתאים. מחזירה true במקרה של הצלחה ו false במקרה של כישלון. נכשלת אם הסטודנט המבוקש לא קיים במערכת.

- setHwGrade – מקבלת מספר זיהוי של סטודנט, מספר הקורס (EE_Course או CS_Course), מספר סידורי של תרגיל הבית וציון התרגיל. מכניסה את הציון לקורס המתאים של הסטודנט המתאים. מחזירה true במקרה של הצלחה ו- false במקרה של כישלון. נכשלת אם או סטודנט, או קורס, או תרגיל בית המבוקש לא קיימים במערכת
- setExamGrade – מקבלת מספר זיהוי של סטודנט, מספר הקורס (EE_Course או CS_Course) וציון המבחן. מכניסה את הציון לקורס המתאים של הסטודנט המתאים. מחזירה true במקרה של הצלחה ו- false במקרה של כישלון. נכשלת אם סטודנט או קורס המבוקש לא קיימים במערכת
- setFactor – מקבלת מספר הקורס ופקטור ומוסיפה את הפקטור לכל הסטודנטים שלמדו את הקורס. מחזירה true במקרה של הצלחה ו- false במקרה של כישלון.
- printStudent – מקבלת מספר זיהוי של סטודנט ומדפיסה את הפרטים שלו לפי פורמט ההדפסה המתואר במחלקת הסטודנט. מחזירה false אם הסטודנט המבוקש לא קיים במערכת ו- true במקרה של הצלחה.
- printAll – מדפיסה פרטים של כל הסטודנטים הקיימים במערכת.
- resetStArray – מוחקת את כל הסטודנטים שקיימים במערכת ומשחררת את הזיכרון שהם תופסים.

קבצים נתונים:

נתון לכם הקובץ Proj.h המכיל הגדרות של קבועים וטיפוסים בהם עליכם להשתמש בתרגיל. אין לשנות קובץ זה.

נתון לכם הקובץ main.cpp. קובץ זה מכיל "parser" עבור פקודות מהמשתמש שמתקבלות מהקלט הסטנדרטי. הפקודות משתמשות בפונקציות של ה- StArray אותן אתם נדרשים לממש ומתאימות להן בדיוק מבחינת פרמטרים. אין לשנות קובץ זה. להלן רשימת הפקודות שמתקבלות מהמשתמש:

- Add_Student
- Add_EE_Course
- Add_CS_Course
- Set_HW_Grade
- Set_Exam_Grade
- Set_Factor
- Print_Student
- Print_All
- Exit – פקודה זאת מסיימת את התכנית.

נתונים קבצי קלט לדוגמא: in1.txt, in2.txt וקבצי פלט לדוגמא: out1.txt, out2.txt

הטכניון - מכון טכנולוגי לישראל
הפקולטה להנדסת חשמל
מבוא למערכות תוכנה

תרגילון Bash:

כתבו script בשם break_word המקבל מחרוזת כפרמטר ומדפיס את אותיותיה בשורות נפרדות (לפי הסדר).

הטכניון - מכון טכנולוגי לישראל
הפקולטה להנדסת חשמל
מבוא למערכות תוכנה

הנחיות הגשה:

1. קבצי קוד חלקיים, וכן קבצי קלט ופלט לדוגמה, נמצאים בתיקייה:
~eesoft/hmw/hmw4
לפני תחילת העבודה, הורידו את הקבצים לחשבונכם באמצעות הפקודה:
cp ~eesoft/hmw/hmw4/* .
2. עברו היטב על הוראות ההגשה של תרגילי הבית המופיעים באתר טרם ההגשה! ודאו כי התכנית שלכם עומדת בדרישות הבאות:
 - התכנית קריאה וברורה
 - התכנית מתועדת היטב לפי דרישות התייעוד המופיעות באתר
 - התכנית מתקמפלת ללא שגיאות וללא warnings כלל, יש לקמפל עם הדגלים g-Wall בלבד.
 - התכנית רצה ללא דליפות זיכרון וגישות לא חוקיות לזיכרון כלל (בדיקה באמצעות valgrind)
 - התכנית נותנת פלט **זהה לחלוטין** לפלט הצפוי על כל קבצי הקלט שסופקו (בדיקה באמצעות פקודת diff על קבצי הפלט)
 - קובץ ה-makefile יוצר קובץ הרצה בשם הנדרש
3. יש להגיש קובץ tar יחיד המכיל את **כל הקבצים** שאתם נדרשים להגיש **ואותם בלבד** – ללא תתי-תיקיות. ודאו כי לא שכחתם את קובץ readme המכיל את פרטי הסטודנטים, וכן את ה-makefile במידה ונדרשתם.
4. שאלות בנוגע לתרגיל יש להפנות לפורום התרגיל ב-moodle בלבד – ניתן לשלוח שאלות במייל **למתרגל האחראי על התרגיל בלבד**, ורק במידה והשאלה מכילה פתרון חלקי.
5. סיכום מפרט התרגיל:

סעיף	תיאור
נושא התרגיל	C++
תאריך ההגשה	יום ד', 14/06/2014 בשעה 23:59
המתרגל האחראי על התרגיל	שיראל יוזף shirel@campus
תיקייה המכילה קבצים לשימוש הסטודנטים	~eesoft/hmw/hmw4
קבצי הקוד הנתונים	Proj.h main.cpp
קבצי הקלט והפלט הנתונים	in1.txt out1.txt in2.txt out2.txt
הקבצים שיש להגיש	readme makefile Course.h Course.cpp CS_Course.h CS_Course.cpp EE_Course.h EE_Course.cpp Person.h Person.cpp Student.h Student.cpp StArray.h StArray.cpp break_word
שם תכנית ההרצה הדרושה (הנוצרת ע"י makefile)	students
דגשים מיוחדים	

בהצלחה!