

מכללת הדסה, החוג למדעי המחשב

תכנות מונחה עצמים ופיתוח משחקים

סמסטר ב', תשע"ט

תרגיל 2

תאריך אחרון להגשה:

קמפוס הנביאים: יום ב', ג' ניסן, 08/04/19 בשעה 23:59
קמפוס שטראוס: יום ג', כ"ו אדר ב', 02/04/19, בשעה 23:59

מטרת התרגיל:

תרגול תבניות (Templates).

תיאור כללי:

בתרגיל זה נבנה אפליקציית מסוף (Terminal, Console) המשמשת למילוי טופס להזמנת טיסה. המשתמש מכניס מספר פרטים אישיים וכן פרטים על יעד טיסתו, המטבע בו הוא משלם והמחלקה בה הוא רוצה לנסוע. התוכנה מוודאת שהפרטים תקינים, כולל אימות של פרטי הטיסה מול המידע הידוע מראש על האפשרויות השונות.

בסיס העבודה:

אתם מקבלים קובץ main.cpp המכיל את הפונקציה main ומספר פונקציות עזר, ועליכם להוסיף קבצים מתאימים עם הקוד הנדרש כדי לגרום לקובץ להתקמפל ולרוץ. את הקובץ הזה אסור לשנות. אם ישנה בעיה בקובץ, אנא שלחו זאת בפורום. שימו לב שהקוד בקובץ בעצם מגדיר דרישות מסוימות שהמחלקות שנכתוב חייבות לממש כדי שהקובץ הזה אכן יתקמפל ויפעל כראוי.

תיאור כללי של ריצת התוכנית:

1. התוכנית רצה בקונסול, ללא ממשק גרפי, כפי שצוין לעיל.
2. התכנית מציגה למשתמש שדה אחר שדה בבקשה למלא את הפרטים הנדרשים. כל השדות הם חובה ואם לא מילא את כולם, התכנית תמתין.

3. לאחר שהמשתמש סיים למלא את הטופס, התוכנית בודקת את תקינות השדות.
4. אם קיימים שדות לא תקינים, התוכנית תציג למשתמש הודעה על כך, ותציג את כל השדות שהזין כשליד השדות הבעייתיים תופיע שגיאה מתאימה. (או אחרי רשימת השדות, אם הבעיה היא בהתאמה בין שדות). כעת התוכנית תבקש מהמשתמש להזין מחדש את המידע בדומה לסעיף 2, אולם הפעם היא תבקש זאת רק עבור השדות הלא-תקינים, ותחזור לסעיף 3.
5. אם כל השדות תקינים, התוכנית תציג הודעה המודה על ההרשמה, תציג את המידע שהזין, ותצא.

רשימת השדות וסוגי המידע החוקיים עבורם:

1. שם – לא יכול להכיל ספרות. אין צורך לבדוק יותר מכך, כלומר יכולים להופיע בשם מקפים, פסיקים וכו'.
 2. ת.ז. – מאומתת לפי ספרת הביקורת. את האלגוריתם ניתן למצוא בוויקיפדיה בערך "ספרת ביקורת".
 3. שנת לידה – לצורך התרגיל, הגיל המינימלי להזמנת טיסה הוא 15. לגבי גיל מקסימלי, הנחת העבודה שלנו היא שאדם מעל גיל 120 לא חי וזו כנראה טעות. אין צורך להבדיל בהודעת השגיאה בין המקרים של גיל צעיר מידי ומבוגר מידי, מספיק לומר שזה לא בטווח. (מספיק לבדוק רק אם ההפרש בין השנה הנוכחית לשנת הלידה הוא בין 15 ל-120)
 4. יעד טיסה – המשתמש יכניס מספר בין 1 ל-5 המייצג את יעדי הטיסות הבאים:
 - 1 - רומא
 - 2 - פראג
 - 3 - ניו יורק
 - 4 - הודו
 - 5 - תאילנד
- אם המשתמש מכניס מספר קטן מ-1 או גדול מ-5, זו תהיה שגיאה באימות השדה הזה (וכמו בשנת הלידה, גם כאן אין צורך להבדיל בין סוגי השגיאות ומספיק לציין שהמספר לא בטווח האפשרי). אם המשתמש הכניס ערך תקין, יוצג שם היעד ולא המספר שהמשתמש הכניס.
5. מחלקת הטיסה – המשתמש יכניס מספר בין 1 ל-3 המייצג את סוגי המחלקות השונות:
 - 1 - מחלקה ראשונה

○ 2 - מחלקת עסקים

○ 3 - מחלקת תיירים

אם המשתמש מכניס מספר קטן מ-1 או גדול מ-3, זו תהיה שגיאה באימות השדה הזה (ושוב, אין צורך להבדיל בין סוגי השגיאות ומספיק לציין שהמספר לא בטווח האפשרי). אם המשתמש הכניס ערך תקין, יוצג שם המחלקה ולא המספר שהמשתמש הכניס.

כאן אתם נדרשים לעוד סוג של אימות:

- אם המשתמש הזמין טיסה לרומא, לפראג - הוא יכול לבחור רק את המחלקה הראשונה או את מחלקת עסקים בלבד.

- אם המשתמש הזמין טיסה לניו יורק - הוא יכול לבחור כל אחת מהמחלקות.

- אם המשתמש הזמין טיסה לתאילנד או להודו - הוא יכול לבחור רק את מחלקת עסקים או מחלקת תיירים בלבד.

אם יש כאן שגיאה באימות שני השדות יחד, תהיה שגיאה באימות הטופס.

6. סוג המטבע בו הוא משלם:

○ 1 - שקל

○ 2 - אירו

○ 3 - דולר

גם כאן, בחלק של אימות השדה, מספיק לזהות את זה שהמספר שנבחר איננו בטווח, לא משנה אם זה כי הוא קטן מידי או גדול מידי, ובתצוגה, אם המשתמש הכניס ערך תקין, יוצג שם המטבע ולא המספר שהמשתמש הכניס.

בנוסף, אתם נדרשים לבדוק שאם הזמין טיסה לניו יורק הוא לא יכול לבחור לשלם באירו (כי הוא טס מישראל לניו יורק אז רק שקל או דולר) ואם הזמין טיסה לפראג או לרומא, הוא לא יכול לשלם בדולר. ושוב, זה ייתפס כחלק מאימות הטופס.

מסכי הדגמה לפלט התוכנית:

המשתמש מזין נתונים (הקלט מסומן בצהוב):

+-----+

| Hello and welcome! |

| Please fill in the fields below |

+-----+

What is your name?

Yehezkel2

What is you ID?

123456789

What is your year of birth?

2010

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand)

7

Which class you want?

(1 - First Class, 2 - Business Class, 3 - Economy Class)

5

In what currency will you pay?

(1 - NIS, 2 - Euro, 3 - Dollar)

3

התוכנית מציגה למשתמש הודעת שגיאה ואת תוכן הטופס עם הודעות שגיאה מתאימות ומבקשת מחדש רק את השדות שאינם תקינים. נשים לב שסוג המטבע נחשב תקין כרגע במימוש הספציפי הזה כי האימות של השדה בפני עצמו בודק רק שמספר הוא בטווח שבין 1 ל-3 והאימות מול היעד לא מוגדר היטב כשהערך בשדה היעד עצמו לא תקין ולכן לא תופיע שגיאה בשלב הזה. בהמשך להודעת השגיאה התוכנית תציג מחדש את השדות שהיו בהם שגיאות ותחכה לקלט חדש מהמשתמש:

+-----+

| There was an error in at least one of the fields! |

| Please correct the error(s) |

+-----+

What is your name? = Yehezkel2 Can't contain digits

What is your ID? = 123456789 Wrong control digit

What is your year of birth? = 2010 Out of range

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand) = 7 Out of range

Which class you want?

(1 - First Class, 2 - Business Class, 3 - Economy Class) = 5 Out of range

In what currency will you pay?

(1 - NIS, 2 - Euro, 3 - Dollar) = Dollar

What is your name?

rut

What is your ID?

543700421

What is your year of birth?

1990

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand)

1

Which class you want?

(1 - First Class, 2 - Business Class, 3 - Economy Class)

3

כעת, המשתמש הכניס את כל השדות תקינים על פי אימות כל שדה לעצמו אבל יש בעיה בחוסר
התאמה בין השדות. ולכן כעת התוכנית תציג את השגיאה הקיימת ותחכה שוב לקלט מהמשתמש
בשדות השגויים:

+-----+

| There was an error in at least one of the fields! |

| Please correct the error(s) |

+-----+

What is your name? = rut

What is your ID? = 543700421

What is your year of birth? = 1990

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand) = Rome

Which class you want?

(1 - First Class, 2 - Business Class, 3 - Economy Class) = Economy Class

In what currency will you pay?

(1 - NIS, 2 - Euro, 3 - Dollar) = Dollar

Destination and class don't match

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand)

1

Which class you want?

(1 - First Class, 2 - Business Class, 3 - Economy Class)

2

עכשיו יש עוד בעיה באימות בין שדות בטופס ולכן יוצג למשתמש:

+-----+

| There was an error in at least one of the fields! |

| Please correct the error(s) |

+-----+

What is your name? = rut

What is your ID? = 543700421

What is your year of birth? = 1990

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand) = Rome

Which class you want?

(1 - First Class, 2 - Business Class, 3 - Economy Class) = Business Class

In what currency will you pay?

(1 - NIS, 2 - Euro, 3 - Dollar) = Dollar

Destination and currency don't match

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand)

1

In what currency will you pay?

(1 - NIS, 2 - Euro, 3 - Dollar)

2

לסיום, לאחר שכל השדות תקינים, יוצג סיכום על המסך:

+-----+

| Thank you! |

| This is the data you sent: |

+-----+

What is your name? = rut

What is your ID? = 543700421

What is your year of birth? = 1990

What is your flight destination?

(1 - Rome, 2 - Prague, 3 - New York, 4 - India, 5 - Thailand) = Rome

Which class you want?

(1 - First Class, 2 - Business Class, 3 - Economy Class) = Business Class

In what currency will you pay?

(1 - NIS, 2 - Euro, 3 - Dollar) = Euro

מה לא נדרש:

- לא נדרש להתמודד עם קלט לא תקין במקרה שהמשתמש מכניס טקסט במקום מספרים.
- לא נדרש לאפשר להכניס רווחים בשם.

- אין צורך להקפיד שההדפסה למסך תהיה זהה למוצג לעיל. העיקר שמוצגות הודעות שגיאה באופן דומה למוצג לעיל.

הערות לתיכון התוכנית:

ברור שצריך לחשוב על תיכון ראוי שיתאים לקוד הקיים כדי לאפשר לו להתקמפל ולרוץ באופן תקין, אולם חשוב גם שלא נצטמצם אך ורק למה שנדרש בקוד הקיים ונשאיר מקום להתפתחות עתידית בפרט בנושאים כמו אפשרות להוספת מספר validators לאותו שדה.

הערות נוספות:

- רשימת ה-include יים בקובץ main.cpp יכולה לרמוז על המחלקות הנדרשות, אולם בהחלט ייתכן שקיימות מחלקות שלא נכללות כאן ישירות.
- שימו לב למחלקה ValuesToNames, המשמשת לשלושה שדות. כפי שנאמר לעיל (וניתן לראות בדוגמאות) היא מקבלת ערך מספרי, אבל בהדפסה (עבור ערך תקין) מודפס שם ולא מספר!
- שימו לב גם ל-validator יים השייכים לטופס עצמו (ראו בקוד של פונקציית ה-main). אחד מוודא את ההתאמה בין היעד ובין המחלקה בה המשתמש בוחר לטוס והשני את ההתאמה בין היעד לבין סוג המטבע בו הוא משלם.
- בדיקה ברמת אימות הטופס שנכשלת גורמת לכך שנבקש מהמשתמש מחדש את שני השדות שאליהם היא קשורה, ולא משנה האם השדות הללו תקינים כל אחד בפני עצמו.
- חשוב לזכור: מחלקה תבניתית (Template class) יכולה גם לרשת ממחלקה אחרת, כמו שכל מחלקה אחרת יכולה לרשת, ובפרט היא יכולה לרשת ממחלקה לא תבניתית.

הערות טכניות שימושיות:

- כדאי לשים בהערה את הקוד שעדיין לא כתבנו עבורו את המחלקות המתאימות, כדי לבדוק תוך כדי כתיבה שהקוד שכבר כתבנו מתקמפל ועובד.
- כדי לבדוק האם תו מסוים הוא ספרה, ניתן להשתמש בפונקציה isdigit().
- כדאי מאוד לעבור על הקוד של ה-main כדי להבין מה בדיוק צריך לעשות ומה כבר נכלל בקוד של ה-main. למשל, אין צורך לממש קוד שבודק מהי השנה הנוכחית, זה כבר כלול.

קובץ ה-README:

יש לכלול קובץ README שיקרא README.doc, README.docx או README.txt (ולא בשם אחר).

קובץ זה יכיל לכל הפחות:

1. כותרת.
2. פרטי הסטודנט: שם מלא כפי שהוא מופיע ברשימות המכללה, ת"ז.
3. הסבר כללי של התרגיל.
4. **תיכון (design): הסבר קצר מהם האובייקטים השונים בתוכנית, מה התפקיד של כל אחד מהם וחלוקת האחריות ביניהם ואיך מתבצעת האינטראקציה בין האובייקטים השונים.**
5. רשימה של הקבצים שנוצרו ע"י הסטודנט, עם הסבר קצר (לרוב לא יותר משורה או שתיים) לגבי תפקיד הקובץ.
6. מבני נתונים עיקריים ותפקידיהם.
7. אלגוריתמים הראויים לציון.
8. באגים ידועים.
9. הערות אחרות.

יש לתמצת ככל שניתן אך לא לוותר על אף חלק. אם אין מה להגיד בנושא מסוים יש להשאיר את הכותרת ומתחתיו פסקה ריקה. תכתבו ב-README כל דבר שרצוי שהבודק ידע כשהוא בודק את התרגיל.

אופן ההגשה:

הקובץ להגשה: יש לדחוס כל קובץ הקשור לתרגיל, למעט מה שיצוין להלן, לקובץ ששמו exN_name.zip, כאשר N הוא מספר התרגיל ו-name הוא השם המלא. במקרה של הגשה בזוג, שם הקובץ יהיה לפי התבנית exN_name1_name2.zip, עם שמות המגישים בהתאמה (ללא רווחים; גם בשמות עצמם יש להחליף רווחים בקו תחתי או להצמיד את שני חלקי השם).

לפני דחיסת תיקיית ה-Solution שלכם יש למחוק את הפריטים הבאים:

- תיקיות בשם debug, release או x64 (לרוב יש יותר מתיקיה אחת בשם זה, אחת בתיקיית ה-Solution ואחת בתיקיית כל פרויקט).

- תיקייה (לפעמים מוסתרת!) בשם vs. שאמורה להיות בתיקייה עם ה-Solution.

ככלל אצבע, אם קובץ ה-zip שוקל יותר ממ"ב אחד או שניים, כנראה שלא מחקתם חלק מהקבצים הבינאריים המוזכרים.

את הקובץ יש להעלות ל-Moodle של הקורס למשימה המתאימה.

הגשה חוזרת: אם מסיבה כלשהי סטודנט מחליט להגיש הגשה חוזרת יש לוודא ששם הקובץ זהה לחלוטין לשם הקובץ המקורי. אחרת, אין הבודק אחראי לבדוק את הקובץ האחרון שיוגש.

כל שינוי ממה שמוגדר פה לגבי צורת ההגשה ומבנה ה-README עלול לגרור הורדת נקודות בציון.

מספר הערות:

1. שימו לב לשם הקובץ שאכן יכלול את שמות המגשים.
 2. שימו לב שעליכם לשלוח את תיקיית ה-solution כולה, לא רק את קובצי הקוד שיצרתם. עקבו אחרי ההסבר המפורט באתר, במקרה שאתם לא בטוחים איך למצוא את התיקייה. תרגיל שלא יכלול את ה-solution, לא יתקבל וידרוש הגשה חוזרת (עם כללי האיחור הרגילים).
- המלצה כללית: אחרי שהכנסתם את הקובץ להגשה, העתיקו אותו לתיקייה חדשה, חלצו את הקבצים שבתוכו ובדקו אם אתם מצליחים לפתוח את ה-solution שבתוכו ולקמפל את הקוד. הרבה טעויות של שכחת קבצים יכולות להימנע על ידי בדיקה כזו.

בהצלחה!