

תרגילים

תרגיל 1

מזכירות המשרד לענייני סחבת מקבלת מדי יום מכתבים ומדוורות (מחלקת) אותם לנמענים השונים במשרד. כל מכתב שמתקבל במשרד נרשם (שם נמען ותאריך קבלה) ומקבל מספר סידורי שוטף (מספור רץ). במערכת התוכנה של המשרד קיים מודול **מכתבים**, המטפל במידע הקשור למכתבים השונים.

להלן תאור חלק המחלקה **מכתב** :

<p>נמען ()</p> <p>Java: String getRecipient () C#: string GetRecipient ()</p>	<p>פעולה המחזירה את הנמען של המכתב.</p>
<p>תאריך ()</p> <p>Java: Date getDate () C#: Date GetDate()</p>	<p>פעולה המחזירה את תאריך המכתב.</p>
<p>העבר (נמען)</p> <p>Java: void setRecipient (String recipient) C#: void SetRecipient (string recipient)</p>	<p>הפעולה משנה את שם הנמען של המכתב להיות נמען (חדש). הנחה : הנמען (החדש) תקין.</p>

להלן חלק משק המחלקה **אוסף-מכתבים** :

<p>הוסף-מכתב (תאריך, נמען)</p> <p>Java: void add (Date date, String recipient) C#: void Add (Date date, String recipient)</p>	<p>פעולה המקבלת את תאריך קבלת המכתב למשרד ואת שם הנמען, קובעת את מספרו הסידורי של המכתב ומוסיפה את המכתב לאוסף המכתבים. הנחה : התאריך והנמען תקינים.</p>
<p>קיבל? (תאריך, נמען)</p> <p>Java: boolean recived (Date date, String recipient) C#: bool Recived (Date date, string recipient)</p>	<p>הפעולה מחזירה "אמת" אם הנמען קיבל מכתב בתאריך הנתון, ו"שקר" אחרת. הנחה : התאריך והנמען תקינים.</p>

- א. ייצג את העצם **מכתב** וכתוב עבורו את המחלקה (שים לב! בתאור שלעיל מוצגות רק חלק מפעולות המחלקה.
- ב. ייצג את **אוסף-המכתבים** וציין כיצד יישמרו המכתבים באוסף (האם ישמרו בסדר אקראי או ממויינים בצורה כלשהי).
- ג. ממש את הפעולה **הוסף-מכתב** (תאריך, נמען) בהתאם לסדר במכתבים באוסף כפי שקבעת בסעיף ב.
- ד. כתוב פעולה בשם **רשימת-מכתבים** (נמען, תאריך), המקבלת שם של נמען ומחזירה רשימה המכילה את כל המכתבים שהנמען קיבל מהתאריך זה ועד היום.
- ה. מהי סיבוכיות האלגוריתם שכתבת בסעיף ג? נמק!

תכני 2

לקראת חופשת הקיץ החליטו במושב הלצים להשכיר חדרי נופש. בשלב הראשון החליטו להשכיר חדרים ב-31 ימי חודש אוגוסט בלבד. מזכיר המושב אסף את המידע הבא על חברי המושב המעוניינים להשכיר חדר: שם המשפחה, מספר הטלפון ומספר החדרים שברצונו להשכיר.

במערכת התוכנה של מזכירות היישוב קיים מודול **חדרים**, המטפל במידע הקשור לחדרים להשכרה. להלן חלק מהממשק **חדרים**:

<p>הזמן-חדר (תאריך1, תאריך2, שוכר)</p> <p>Java: PhoneNum booking (Date fromDate, Date toDate, String guestName)</p> <p>C#: PhoneNum Booking (Date fromDate, Date toDate, string guestName)</p>	<p>פעולה המקבלת שם של שוכר הרוצה לשכור חדר בין התאריכים: תאריך1 ותאריך2 (כולל). הפעולה מחזירה את מספר הטלפון של המשכיר ומעדכנת את אוסף החדרים בהתאם. במידה ואין חדר פנוי בתאריכים הנ"ל תחזיר הפעולה את המספר 0.</p> <p>הנחה: התאריכים תקינים (בין 1 ל-31)</p>
<p>רשימת-פנויים (תאריך)</p> <p>Java: List<String> available (Date date)</p> <p>C#: List<string> Available (Date date)</p>	<p>פעולה המחזירה רשימה המכילה את שמות כל המשכירים שיש להם חדר פנוי בתאריך הנתון. במידה ולמשפחה יש יותר מחדר אחד פנוי, יופיע שם המשפחה פעם אחת בלבד.</p> <p>הנחה: התאריך תקין.</p>

- א. ייצג באופן מפורט את אוסף החדרים.
- ב. ממש את הפעולה **הזמן-חדר** (תאריך1, תאריך2, שוכר)
- ג. ממש את הפעולה **רשימת-פנויים** (תאריך).
- ד. מהי הסיבוכיות של כל אחת מהפעולות שכתבת בסעיפים ב' ו-ג? נמק!

תרגיל 3

(בגרות)

ביציאה מהעיר "שיממון" יש טרמפיאדה. אנשים המעוניינים בטרמפ מגיעים לטרמפיאדה ומחכים. כאשר מכונית עוצרת בטרמפיאדה, מכריז הנהג על יעדו ועל מספר המקומות הפנויים במכוניתו. למכונית נכנסים אנשים לפי סדר העדיפויות הבא: קודם – חיילות, לאחר מכן – חיילים, ורק בסוף אזרחים. בתוך כל קבוצה (חיילות, חיילים, אזרחים) סדר הכניסה למכונית הוא לפי סדר ההגעה לטרמפיאדה. רוצים לבנות מודול תוכנה לניהול הטרמפיאדה.

א. ייצג את המידע על האנשים שמחכים בטרמפיאדה.

ב. כתוב אלגוריתם **אחזר-רשימת-נוסעים-ליעד** (יעד, מספר מקומות פנויים במכונית). האלגוריתם מקבל יעד נסיעה ומספר מקומות פנויים במכונית, מחזיר רשימה של הנוסעים שייכנסו למכונית ומעדכן את המידע השמור על האנשים שמחכים בטרמפיאדה, בהתאם לייצוג שכתבת בסעיף הקודם. למערכת התוכנה של הטרמפיאדה הוכנס מודול המטפל במסלולי הנסיעה מהעיר "שיממון" ליעדים שונים. במודול ממומשת הפעולה **אחזר-יעד-ביניים** (יעד):

<p>אחזר-יעד-ביניים (יעד)</p>	<p>הפעולה מחזירה רשימה של יישובים שבין העיר "שיממון" ובין היעד (כולל). הרשימה מסודרת לפי סדר היישובים לאורך מסלול הנסיעה.</p>
-------------------------------------	---

ג. כתוב אלגוריתם **אחזר-רשימת-נוסעים** (יעד, מספר מקומות פנויים במכונית). האלגוריתם מקבל יעד נסיעה ומספר מקומות פנויים במכונית ומחזיר רשימה של הנוסעים שייכנסו למכונית לפי הכלל הזה: תחילה ייכנסו למכונית אנשים המעוניינים להגיע ליעד של הנהג, לאחר מכן אנשים שיעדם הוא היישוב שלפני היעד של הנהג, וכך הלאה עד למילוי המקומות הפנויים במכונית. אם כמה אנשים מעוניינים להגיע לאותו יעד, תיקבע כניסתם למכונית לפי סוג הנוסע (חיילת, חייל, אזרח), ולפי סדר הגעתם לטרמפיאדה.



תרגיל 4

(בגרות)

במכללת "סוד ההצלחה" הוחלט למחשב את מערכת השעות. לצורך כך הוגדר מודול **מערכת שעות** האוגר בתוכו את הנתונים שלפניך: בכל יום מימי הלימודים א-ה ובכל שעה משעות הלימוד 08:00 – 16:00 נלמד קורס אחר. לכל שעה במערכת נאגרים הנתונים האלה: מספר הקורס, מספר המורה המעביר את הקורס, ורשימה של מספרי תעודות הזהות של התלמידים (הרשימה אינה מוגבלת באורכה).

להלן חלק ממשק המודול **מערכת-שעות**:

הפעולה מחזירה עותק של רשימת התלמידים בקורס שמספרו מספר-קורס. הנחה: מספר הקורס תקין.	אחזר-רשימת-משתתפים (מספר-קורס)
הפעולה מחזירה את מספר המורה בקורס שמספרו מספר-קורס. הנחה: מספר הקורס תקין.	אחזר-מורה (מספר-קורס)
הפעולה מחזירה את מספר הקורס המתקיים ב-יום, וב-שעה. אם לא מתקיים קורס בזמן הנתון, יוחזר המספר 0. הנחה: היום והשעה תקינים.	אחזר-מספר-קורס (יום, שעה)

- יצג את המידע במודול **מערכת-שעות**.
- כתוב אלגוריתם המממש את הפעולה **אחזר-רשימת-משתתפים** (מספר-קורס).
- כאשר מורה חולה יש להודיע לכל המשתתפים בקורסים שבהם הוא מלמד באותו יום על ביטול השיעורים ליום זה.
בנה פעולה **תלמידים-של-מורה** (מספר-מורה, יום), המחזירה רשימה של מספרי תעודות הזהות של התלמידים שמלמד המורה שמספרו מספר-מורה ב-יום. יש לוודא שכל מספר תעודת זהות אינו מופיע יותר מפעם אחת.
- מהי סיבוכיות זמן הריצה של הפעולה **תלמידים-של-מורה** כפונקציה של מספר התלמידים במכללה? נמק!

תרגיל 5

(בגרות)

מקטע-משותף-מקסימלי לשתי רשימות נתונות הוא תת-הרשימה הרצופה המקסימלית, המשותפת לשתיהן.

לדוגמא, עבור הרשימות L1 ו-L2 :

L1 =

1, 5, 6, 3, 4, 8, 9, 5, 4, 3, 6, 7

L2 = 5, 4, 4, 8, 9, 5, 3, 6

L3 = 4, 8, 9, 5 : המקטע-משותף-מקסימלי הוא הרשימה

נתונה הפונקציה :

function **list-compare** (L1, L2 : list-type; pos1, pos2 : pos-type; n : integer) : boolean ;

הפונקציה מקבלת שתי רשימות L1 ו-L2 ושני מצביעים pos1 ו-pos2 המצביעים על מקומות ב-L1 ו-L2 בהתאמה. הפונקציה מחזירה true אם האיברים החל מ-pos1 ברשימה L1 זהים ל-n האיברים החל מ-pos2 ברשימה L2, ו-false אחרת. אם מספר האיברים אחרי pos1 או pos2 קטן מ-n, תחזיר הפונקציה false.

א. כתוב פרוצדורה בסביבת העבודה, שתקבל שתי רשימות L1 ו-L2 ותחזיר את המקטע-המשותף-המקסימלי לשתיהן.

ב. מהי סיבוכיות זמן הריצה של הפעולה שמימשת בסעיף הקודם? נמק! אם נעזרת בפונקציה list-compare, הנח כי סיבוכיות זמן הריצה שלה, כפונקציה של n המועבר כפרמטר, היא O(n).

שים :/ הפונקציה list-compare נתונה, כלומר היא קיימת ואין צורך לממש אותה אלא להשתמש בה בלבד.



בילוי נעים !!

הילה