



יסודות מזעי המחשב

יסודות-1 מעבדה

מהדורת ניסוי

המחלקה להוראת המדעים
מכון ויצמן למדע, רחובות



יפעת קוליקנט

מחבר:

דר' מרדכי בן-ארי

צוות הפיתוח:

דר' דוד גינות

חנה מחלב

נורית רייך

יפעת קוליקנט

הדפסה ועריכה במחשב:

אורנה עמר

מירב קויפמן

עיצוב עטיפה והפקה:

אגי (רחל) בוקשפן

צוות הפיתוח מודה לדורית כהן
על הערותיה המועילות על
המהדורה הקודמת.

אין לשכפל, להעתיק, לצלם, להקליט, לתרגם, לאחסן במאגר מידע, לשדר או לקלוט בכל דרך או
אמצעי אלקטרוני, אופטי או מכני או אחר כל חלק שהוא מהחומר שבספר זה.
שימוש מסחרי מכל סוג שהוא בחומר הכלול בספר זה אסור בהחלט אלא ברשות מפורשת
בכתב מהמוציא לאור.

©

כל הזכויות שמורות למכון ויצמן למדע
ומשרד החינוך התרבות והספורט

נדפס בישראל תשנ"ז - 1997
מהדורה מתוקנת: תשי"ס - 2000
דפוס ניידט בע"מ

תוכן הענינים

מבוא

פרק 1 – הכרה בסיסית של סביבת הפיתוח

- 7. משימה 1 – הכרות עם תפריטים
- 10. משימה 2 – הכרות ראשונית עם העורך, הדפסת ושמירת קובץ

פרק 2 – המחשת התהליך קלט ← עיבוד ← פלט

- 15. משימה 1 – הכרות בסיסית עם סביבת העבודה : load, compile, run
- 17. משימה 2 –
- 17. משימה 3 –
- 18. משימה 4 –
- 18. משימה 5 –
- 18. משימה 6 –

פרק 3 – אבני בניין של אלגוריתם למחשב

- 19. משימה 1 – הדפסת מחרוזת, עבודה בגושים, הידור והרצת תכנית
- 23. משימה 2 – תרגול read, write, writeln הקניית המושגים בהירות הפלט ורוחב שדה ההדפסה
- 26. משימה 3 – ההבדלים בין read-readln ו write-writeln
- 29. משימה 4 – חשיבות בהירות-הדו-שיח משתמש - תוכניות מחשב
- 32. משימה 5 – פתרון בעייה ע"י תוכנית עם משפטי השמה
- 33. משימה 6 – טיפול בשגיאות תחביר הקשורות בהצהרת משתנים
- 36. משימה 7 – הכרת מנפה השגיאות (watch ו trace), המחשת טבלת מעקב
- 40. משימה 8 – טיפול בפלט של מספר מטיפוס ממשי
- 43. משימה 9 – תרגול div , mod

פרק 4 – פונקציות תקניות

- 45. משימה 1 – זימון לא נכון של פונקציות תקניות
- 47. משימה 2 – תרגול זימון פונקציות, abs ו trace

פרק 5 – משפט if

- 49. משימה 1 – הכרת הביטוי הבוליאני, תרגול משפט if
- 52. משימה 2 – הפנמת התחביר של משפט ה-if
- 54. משימה 3 – ישום אלגוריתם, עם משפט if (בעית. הפלינדרום)
- 55. משימה 4 – מילוי טבלת מעקב לתוכנית עם משפט if
- 59. משימה 5 – חידוד ההכרות עם הפונקציות succ ו pred
- 61. משימה 6 – ישום תוכנית עם בלוקים של if
- 62. משימה 7 – תרגול תחביר של משפט if מורכב

משימה 8 – ישום אלגוריתם עם משפט if המורכב מקשר or 64

פרק 6 – נכונות

משימה 1 – תיקון אלגוריתם לא נכון, בעזרת מנפה השגיאות 67

פרק 7 – לולאות

משימה 1 – הכרת לולאות for 71

משימה 2 – תוכנית שגויה ולא מעומדת עם צובר 74

משימה 3 – פיתוח וישום אלגוריתם המשתמש במונה 77

משימה 4 – לימוד נקודה עצירה, תרגול לולאת while 78

משימה 5 – פיתוח וישום אלג' לתוכנית המשתמשת בלולאות while 83

משימה 6 – לולאה אינסופית, הפסקת הרצה ע"י המשתמש 85

משימה 7 – פיתוח וישום אלג' לתוכנית המשתמשת בזקיף 87

פרק 8 – יעילות

משימה 1 – הבהרת המושג יעילות 89

פרק 9 – פונקציות

משימה 1 – מעקב אחרי זימון וביצוע של פונקציה בתוכנית 92

משימה 2 – פיתוח פונקציה ותוכנית המשתמשת בפונק' 98

משימה 3 – פיתוח וישום אלג' לתוכנית הקוראת לפונק' מתוך לולאה. 99

פרק 10 – מערכים

משימה 1 – מעקב אחר משתנה מטיפוס מערך במהלך ביצוע התוכנית 101

משימה 2 – פיתוח וישום אלגוריתם, המשתמש במערך 104

משימה 3 – המחשת המושג יעילות מקום וחשיבותו 105

נספח פסקל 108

פרק 1

משימה 1

מטרה: הפעלת סביבת הפיתוח והתבוננות בתפריטים שלה.

1. הדלק את המחשב.

2. אם הנך עובד במחשב ללא תקליט (hard disk), הכנס את תקליטון סביבת הפיתוח ועבור לבצע את סעיף 3. אחרת דלג על סעיף זה ועבור לבצע את סעיף 3.

3. הקש turbo כדי לעבוד עם סביבת הפיתוח, והקש על המקש Enter. אם לא הצליחה הפעולה, תופיע הודעת שגיאה על המסך. אחרת תוך שניות, ישתנה המסך וסביבת הפיתוח תהיה ערוכה לקבל את הוראותיך.

4. התבונן במסך וזהה את האיזורים המסומנים הבאים

1. תפריט ראשי (main menu)

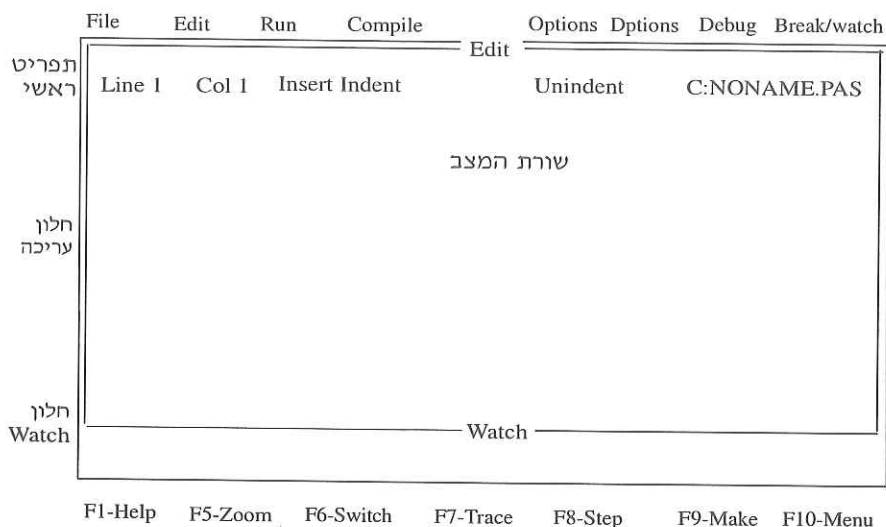
2. חלון עריכה (Edit Window)

– בגרסה 5.5 מופיעה בראשו המילה Edit

– בגרסאות מתקדמות מופיע שם הקובץ

3. תפריט מהיר (Quick Reference)

4. חלון ה-watch (Watch Window)



הערה:

בגרסה 5.5 ישנה גם שורת ססטוס, בשורה הראשונה של המסך.

5. **שים ♥** כי ההוראה File בתפריט הראשי מסומנת.

הקש בחיצים ימינה ושמאלה למעבר בין ההוראות השונות בתוכנית.

6. הקש את מקשי החיצים עד שההוראה File שוב מסומנת.

הקש מקש Enter לבחירת ההוראה File שבתפריט הראשי.

כתוצאה מכך נפתח תת-תפריט (sub menu). הקש בחיצים מעלה ומטה.

7. חזור לתפריט הראשי בעזרת החיצים. בחר בתת-תפריט אחר (Run, Compile).

או (Options), על-ידי מעבר עם החיצים והקשה על המקש Enter.

8. חזור להוראה file. בחר, בעזרת החיצים, בהוראה Quit. הקש על המקש Enter.

יצאת מסביבת הפיתוח.

9. להפעלת התפריט הראשי הקש F10.

ליציאה מהתפריט הקש F10.

סיכום:

1. בעזרתהיציג ניתן לעבור בין ההוראות השונות.
2. בחירת הוראה או תת-תפריט, נעשית על-ידי סימון ההוראה בעזרת החיצים והקשת Enter.
3. ליציאה מסביבת הפיתוח, בחר בתת-תפריט File
בחר בהוראה Quit.

משימה 2

מטרה: לימוד מקשי העריכה בסביבת הפיתוח, הדפסת קובץ, שמירת קובץ.

במשימה זו תלמד כיצד לערוך קובץ.

שים לב - מעתה כשנרצה שתקיש על המקש Enter, נרשום: הקש Enter.

0. היכנס לסביבת הפיתוח.

1. כעת ניצור קובץ חדש:

בחר בתת-תפריט File

ובחר בהוראה New

(תזכורת: בעזרת החיצים, סמן את ההוראה File והקש Enter.

סמן בעזרת החיצים את ההוראה New, והקש Enter)

שים לב שהסמן מהבהב. הסמן המהבהב מציין כי חלון העריכה פעיל עתה.

הכרת מקשי עריכה עיקריים

2. היכן עומד הסמן?

3. כתוב את שמך.

4. הקש על המקש Enter.

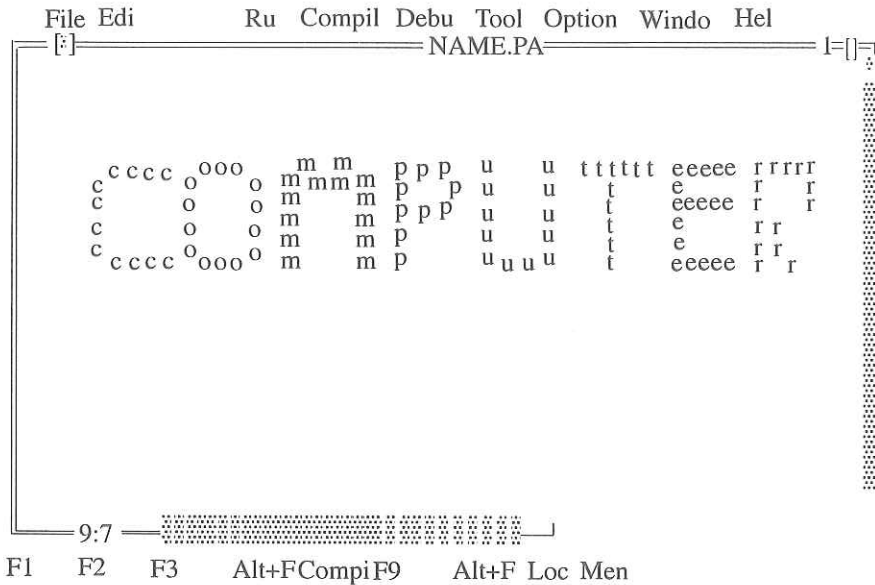
היכן עומד הסמן?

מה תפקידו של מקש Enter?

5. א. הקש את התאריך, באופן הבא: שנה: חודש: יום
למשל, 01:09:1998
כדי להקיש את התו נקודותיים (:), לחץ על המקש SHIFT ומבלי לשחרר אותו לחץ על המקש שבו מופיע התו נקודותיים (:).
- ב. רד לתחילת שורה הבאה בעזרת Enter.
- ג. הקש את שם כיתתך. לחץ על מקש Backspace (←) מה קרה?
מה תפקידו של מקש זה?
6. א. בעזרת החיצים הבא את הסמן לתחילת השורה הראשונה.
- ב. לחץ על מקש delete. מה קרה?
מה תפקידו של מקש זה?
- ג. מה ההבדל בינו לבין המקש Backspace ← Del?
7. לחץ על המקש End. היכן עומד הסמן?
לחץ על מקש Home. היכן עומד הסמן?
8. הבא את הסמן לתחילת השורה הראשונה. כתוב אם שם עריך.
שים לב ששם עריך מיוחד לשורה ושהתמליל שמימין לסמן מוזז ימינה.
9. מחק את כל השורות שכתבת: מחיקת שורה מתבצעת באופן הבא:
א. הבא את הסמן בעזרת החיצים למקום כלשהו בשורה שברצונך למחוק.

ב. לחץ על המקש Ctrl ומבלי לשחררו הקש y.

10. כתוב את שמך בגדול על פני כל המסך.



11 שמור את מה שכתבת בקובץ על התקליטון:

א. הקש F10 להפעלת התפריט.

ב. בחר בתת-תפריט File, בעזרת החיצים → ו ←, הקש Enter.

ג. בחר בהוראה SAVE, בעזרת החיצים ↑ ו ↓, הקש Enter.

ד. הקש את שם הכוון בו נמצא התקליטון, בצמוד לו הקש \ :

ובצמוד הקש שם שבחרת לקובץ. הקש Enter.

למשל את התקליטון נמצא בכוון A, ובחרת לקרוא לקובץ TARGIL1.

הקש \TARGIL1 : A, הקש Enter.

4. הדפס את הקובץ:

א. הקש F10 להפעלת התפריט.

ב. בחר בתת-תפריט File, בעזרת החיצים, והקש Enter.

ג. בחר בהוראה Write TO, בעזרת החיצים והקש Enter.

ד. הקש PRN והקש Enter.

5. צא מסביבת הפיתוח באופן הבא:

א. הקש F10 להפעלת התפריט.

ב. בעזרת החיצים → ו ←, בחר בתת-תפריט File, והקש Enter.

ג. בעזרת החיצים ↑ ו ↓, בחר בהוראה Quit, והקש Enter.

סיכום

1. מעבר לתפריט הראשי מתבצע על-ידי הקשת F10.

2. בחירה של תת-תפריט או הוראה - מעבר עד להוראה הרצויה בעזרת החיצים והקשת Enter.

3. הקש Del משמש למחיקת התו עליו נמצא הסמן.

4. המקש Backspace משמש למחיקת התו שמשמאל לסמן.

5. המקש Insert משמש כמעבר ממצב של דחיפת תווים למצב של כתיבה על תמליל קיים.

6. מחיקת שורה שהסמן נמצא עליה היא על-ידי הקשת Ctrl ו- y.

7. שמירת קובץ:

א. בחר בתת-תפריט File

ב. בחר בהוראה Save.

ג. הקש שם קובץ \ : שם כונו.

8. הדפסת קובץ:

א. בחר בתת-תפריט File.

ב. בחר בהוראה Write to.

ג. הקש: print.

9. יציאה מסביבת הפיתוח.

א. הקש F10 להפעלת התפריט.

ב. בחר בתת-תפריט File.

ג. בחר בהוראה Quit.

פרק 2

משימה 1

מטרה: המחשת קלט פלט, הכרה בסיסית של סביבת העבודה:

load, compile, run.

1. טען את הקובץ compsc.pas באופן הבא:

א. הקש F10 (להפעלת התפריט)

ב. בחר בתת-תפריט FILE בעזרת החיצים $\rightarrow$ ו $\leftarrow$. הקש ENTER

ג. בחר בהוראה OPEN בעזרת החיצים $\uparrow$ ו $\downarrow$. הקש ENTER.

הקש את שם הקובץ הרצוי

הקש ENTER

2. התוכנית תופיע על המסך.

כעת הדר את התוכנית באופן הבא:

א. הקש F10 (להפעלת התפריט)

ב. בחר בתת-תפריט COMPILE בעזרת החיצים $\rightarrow$ ו $\leftarrow$. הקש ENTER

ג. בחר בהוראה COMPILE בעזרת החיצים $\uparrow$ ו $\downarrow$. הקש ENTER.

בזמן ההידור נבדקת תקינות התחביר של התוכנית. אם התחביר תקין מיוצר קובץ הרצה.

על המסך תופיע ההודעה: Compile successful, press any key:

הקש על מקש כלשהו.

3. הרץ את התוכנית, באופן הבא:

א. הקש F10 (להפעלת התפריט)

ב. בחר בתת-תפריט RUN בעזרת החיצים $\rightarrow$ ו $\leftarrow$. הקש ENTER

ג. בחר בהוראה RUN בעזרת החיצים ↑ ו- ↓ . הקש ENTER.

הרצת תוכנית היא מתן הוראה למחשב לבצע את התוכנית.

במהלך ביצוע התוכניות השונות בפרק זה תתבקש להקליד קלט מסויים, ועל המסך יופיע הפלט.

התבונן בפלט שהתקבל במסך באופן הבא:

א. הקש F10 (להפעלת התפריט)

ב. בחר בתת-תפריט DEBUG: בעזרת החיצים → ו- ← הקש ENTER

ג. בחר בהוראה USER SCREEN בעזרת החיצים ↑ ו- ↓ . הקש ENTER.

ענה על סעיפים 4 ו 5:

4. מהו הקלט שהקשת?

5. תאר את הפלט שנתקבל

6. סגור את התכנית.

לחץ על מקש ALT (בשורה התחתונה של לוח המקשים) ומבלי לשחרר אותו, הקש F3.

משימה 2

1. טען את הקובץ EVEN.PAS, הדר והרץ כפי שלמדת במשימה הקודמת.
2. כאשר תתבקש להקליד נתונים הקש 17 ואחר כך הקש ENTER.
3. תאר את הפלט שנתקבל
4. הרץ את התוכנית שוב עבור הקלט 14. (כלומר כאשר תתבקש להקליד נתונים, הקש 14 ואחר כך הקש ENTER). התבונן בפלט וענה:
5. מהו הקלט שהקשת? 14
6. תאר את הפלט שנתקבל
7. מה מבצעת התוכנית?

משימה 3

- א.
1. טען את הקובץ LETTER.PAS
 2. הדר והרץ.
 3. מה מבצעת התוכנית?
 4. מהו הקלט ומהו הפלט?
- הקלט הוא:
- הפלט הוא:
- ב.
1. טען את הקובץ NAME.PAS
 2. הדר והרץ.
 3. מה מטרת התוכנית?

משימה 4

1. טען את הקובץ NOI.PAS
2. הדר והרץ.
3. מה מבצעת התוכנית?

משימה 5

1. טען את הקובץ PYRAMIDS.PAS
2. הדר והרץ.
3. מה מבצעת התוכנית?

משימה 6

במהלך משימה זו תשחק מול המחשב את המשחק בול פגיעה. המחשב בוחר מספר בן 4 ספרות ועליך לנחש אותו. אם ניחשת נכונה ספרה במספר תופיע הספרה במקום כוכבית, אם ניחשת ספרה אף טעית במיקומה יחשב הדבר כפגיעה.

1. טען את הקובץ B00LP.PAS
2. הדר והרץ.
3. מה הקלט של התוכנית?
4. מהו הפלט של התוכנית?

פרק 3

משימה 1

מטרה: הדפסת מחרוזות, עבודה בגושים, הידור והרצת תוכנית.

1. כתוב תוכנית המדפיסה כרטיס ביקור.

בכרטיס יופיעו שמך הפרטי ושם משפחתך, כיתתך, ושם בית ספרך.

רן בן-דוד י מקיף א

וודא שבראש סדרת הפקודות כתבת begin, ושבסוף התוכנית כתבת end.

2. כעת הדר את התוכנית:

בחר בתת-תפריט COMPILE (תזכורת: כדי לבחור תת-תפריט, הקש F10, לך

בעזרת החיצים עד לתת-תפריט הרצוי, והקש ENTER).

בחר בהוראה COMPILE (לך בעזרת החיצים עד לתת-תפריט הרצוי והקש

ENTER).

המהדר בודק כעת את תקינות התחביר בתוכנית. אם יש שגיאות תחביר, תופיע

השגיאה בשורה מודגשת בראש המסך. תקן את תוכניתך והדר אותה עד שלא

יהיו בה שגיאות תחביר.

כאשר אין שגיאות, תופיע ההודעה COMPILE SUCCESSFUL, PRESS ANY

KEY. לחץ על מקש כלשהו.

3. הרץ את התוכנית, באופן הבא:

בחר בתת-תפריט RUN

בחר בהוראה DEBUG

4. כדי לראות את הפלט

בחר בתת-תפריט DEBUG

בחר בהוראה USER SCREEN.

5. כעת נרחיב את התוכנית, כך שתדפיס שלושה כרטיסי ביקור.

בכרטיס השני נוסיף את מספר טלפון.

בכרטיס השלישי נוסיף את עיר המגורים.

ניתן לקצר את כתיבת התוכנית על-ידי שכפול הגוש שכתבנו קודם.

גוש (block) הוא קטע קובץ - החל מתו אחד וכלה במספר כלשהו של תווים. לעתים, נוח לבצע פעולת עריכה מסוימת על גוש. למשל אם ברצוננו, למחוק או להעתיק, או להעביר למקום אחר מספר שורות, אזי ניתן לסמן את השורות האלו כגוש ובאמצעות הוראה אחת לבצע את הפעולה הרצויה על כל השורות יחד.

כדי לשכפל (להעתיק) קטע מסוים, נסמן את הגוש ונעתיקו באופן הבא:

- א. סמן את הגוש: 1. מקם את הסמן בתחילת הקטע שברצונך לשכפל.
 2. לחץ ביד אחת על מקש <CTRL> וביד השניה על המקש K, עזוב את המקשים ולחץ על המקש B.
(מעכשיו נסמן זאת B <CTRL>K).
 3. מקם את הסמן לאחר סוף הקטע שברצונך לשכפל.
 4. הקש K <CTRL>K.
- כלומר, ביד אחת לחץ על מקש ctrl וביד השניה לחץ על המקש K, אחר כך לחץ עוד הפעם על המקש K.

כעת הגוש מסומן וניתן להפעיל עליו מגוון של פעולות.

הגוש המסומן יופיע בצבע שונה משאר התווים.

- ב. העתק: 1. הבא את הסמן למקום הרצוי.
2. <CTRL>KC.
- ג. בטל הצגת גוש מסומן: הקש <CTRL>KH.
6. לאחר שהעתקת את הגוש המשך לערוך את תוכניתך. הוסף את מספר הטלפון לכרטיס אחד, ואת עיר המגורים לכרטיס אחר.
7. כעת הדר את התוכנית:
 שוב, בחר בתת-תפריט COMPILE
 בחר בהוראה COMPILE.
8. הרץ את התוכנית.
 בחר בתת-תפריט RUN
 בחר בהוראה RUN.
9. כדי לראות את הפלט.
 בחר בתת-תפריט DEBUG

סיכום

1. כדי שהמחשב יבצע תוכנית שכתבנו, יש להדר את התוכנית ולהריצה:
 א. הידור תוכנית - בדיקת תחביר ויצור קובץ הרצה
 בחר בתפריט COMPILE
 בחר בתת-תפריט COMPILE
 ב. הרצת התוכנית - ביצוע התוכנית
 בחר בתפריט DEBUG
 בחר בתת-תפריט RUN

2. כדי לראות את הפלט (אחרי שהרצנו את התוכנית)

בחר בתפריט DEBUG

בחר בתת-תפריט USER SCREEN.

3. עבודה בגושים מתבצעת בשני שלבים

א. סימון הגוש

ב. פעולה על הגוש: העתקה (שכפול) או מחיקה או העברה

פירוט

א. סימון הגוש: 1. מקם הסמן בתחילת הגוש

2. הקש <CTRL>KB

3. מקם הסמן בסוף הגוש

4. הקש <CTRL>KK.

ב. פעולה על הגוש:

(i) העתקה : מקם הסמן במקום אליו ברצונך להעתיק את הגוש

המסומן הקש <CTRL>KC

(ii) העברה : מקם הסמן במקום אליו ברצונך להעתיק את הגוש

המסומן הקש <CTRL>KV

(iii) מחיקה : הקש <CTRL>KY

ביטול סימן של גוש הוא על ידי הקש <CTRL>KH.

משימה 2

מטרה: תרגול read, write, writeln.

הקניית המושגים בהירות הפלט ורוחב שדה ההדפסה.

התבונן בתוכנית באה.

הקלט של תוכנית זו הינו שלושה מספרים שלמים,
פלט התוכנית הוא המספרים מודפסים לפי סדר קליטתם.

1. הקלד את התוכנית או טען את הקובץ three.pas.
(תזכורת: הקש F10 להפעלת התפריט, בחר תת-תפריט FILE בחר הוראה
(.OPEN

```
program ThreeNumbers (input , output) ;  
var  
  A , B, C: integer ;  
begin  
  writeln (' enter הקש שלושה מספרים, הקש enter');  
  read ( A ) ;  
  read ( B ) ;  
  read ( C ) ;  
  writeln (' הפלט המבוקש הוא:');  
  write ( A ) ;  
  write ( B ) ;  
  write ( C ) ;  
  writeln  
end.
```

2. הדר את התוכנית עד שלא יהיו בה שגיאות תחביר.

(תזכורת - כדי להדר: הקש F10 להפעלת התפריט

בחר בתת תפריט COMPILE

בחר בהוראה (COMPILE)

3. הרץ את התוכנית, עם הקלט 1 2 3 לפי הסדר משמאל לימין.

(תזכורת - כדי להרץ: הקש F10 להפעל התפריט בחר בתת תפריט RUN בחר

בהוראה (RUN)

4. התבונן בפלט שהתקבל.

(תזכורת - כדי להתבונן בפלט: הקש F10 להפעלת התפריט בחר בתת תפריט

DEBUG בחר בהוראה (USER SCREEN)

העתק אותו לכאן

5. ערכי A, B ו C הודפסו צמודים זה לזה ולכן אפשר לטעות בקלות ולחשוב

שהודפס מספר אחד בלבד ולא שלושה.

כדי להכניס רווחים בין משתנה אחד לשני, נגדיל את רוחב שדה ההדפסה.

רוחב שדה ההדפסה הינו מספר התווים על המסך שאנו מקצים

כדי להדפיס ביטוי. אם מספר התווים בביטוי להדפסה קטן מרוחב

שדה ההדפסה מתווספים רווחים משמאל, להשלמת שדה ההדפסה.

ברצוננו לשמור שני תווים להדפסה של המשתנה C, כלומר רוחב שדה ההדפסה

המבוקש למשתנה זה הינו 2.

לשם כך שנה את הפקודה write(C) לפקודה החדשה write(C:2) שנה את רוחב

שדה ההדפסה של המשתנה B ל 5 ושל המשתנה A ל 7.

הדר את התוכנית והרץ אותה.

העתק את הפלט שהתקבל

6. כדי להפוך את הפלט לברור יותר, אפשר להוסיף, בעזרת מחרוזות, מעט תיעוד.

שנה את הדפסת המשתנה A מ- `write(A:2)` ל- `write('A = ',A:2)`.

שנה בהתאם את ההדפסה של B ושל C. הדר את התוכנית והרץ אותה.

העתק את הפלט שהתקבל

7. שפר את ההדפסה כך שהמשתנים לא יודפסו צמודים זה לזה. (רמז: הוסף רוחב

שדה להדפסת המחרוזת.)

8. צמצם את שלוש פקודות ה- `write` לאחת, מבלי לפגוע בצורת ההדפסה.

9. סכם את שלוש פקודות ה- `read` לפקודה אחת מבלי לפגוע בצורת הקלט.

סיכום

1. אפשר לשפר את בהירות הפלט על ידי הקפדה על רוחב שדה הדפסה נאות

והדפסת מחרוזות שבהן יש תיעוד של הפלט.

הגדרת רוחב שדה ההדפסה היא על-ידי המשפט:

(רוחב שדה ההדפסה המבוקש: הביטוי להדפסה) `write/writeln`.

משימה 3

מטרה: ההבדלים בין read-readln ו write-writeln, הדפסת שורה ריקה.

לפניך תוכנית שגויה.

1. הקלד את התוכנית או טען את הקובץ three.pas וערוך אותו, עד שתקבל התוכנית הבאה. (תזכורת: הקש F10 להפעלת התפריט. בחר בתפריט File. בחר בהוראה Open).

```
program chap3-3(input, output);
var
  A , B, C: integer ;
begin
  writeln ('הקש שני מספרים, הקש enter, הקש עוד שני מספרים');
  readln ( A ) ;
  read ( B ) ;
  read ( C ) ;
  writeln ('          :הפלט המבוקש הוא');
  write ( A ) ;
  write ( B ) ;
  write ( C )
end.
```

2. הדר את התוכנית עד שלא יהיו בה שגיאות תחביר.
(תזכורת - כדי להדר: הקש F10 להפעלת התפריט
בחר בתת תפריט Compile
בחר בהוראה Compile).

3. הרץ את התוכנית, עם הקלד 1 2 לפי הסדר משמאל לימין. הקפד להקיש רווח

3 4

בין נתון לנתון.

(תזכורת - כדי להריץ הקש: F10 להפעלת התפריט בחר בתת תפריט RUN בחר בהוראה RUN).

4. התבונן בפלט שהתקבל

(תזכורת - כדי להתבונן בפלט: F10 להפעלת התפריט בחר בתת תפריט DEBUG בחר בהוראה USER SCREEN)

העתק אותו לכאן

5. כדי להכניס רווחים בין משתנה אחד לשני, הגדל את רוחב שדה ההדפסה, לכל משתנה. הדר והרץ את התוכנית, עם אותם נתוני קלט.

העתק את הפלט שהתקבל

6. מדוע קיבל המשתנה B את הערך 3 ולא את הערך 2?

7. לאן נעלם המספר 4?

נתבונן שוב בתוכנית. ביצענו שלוש הוראות קליטת נתונים.

(הקש שלושה מספרים, הקש enter, הקש עוד שני מספרים) writeln

readln (A) ;

read (B) ;

read (C) ;

לכן, נקלט הנתון הראשון למשתנה A, הנתון השני למשתנה B, והנתון השלישי למשתנה C. הנתון הרביעי לא נקלט לאף משתנה ולכן נעלם.

8. שנה את קלט תוכניתך כך שכשתריץ את תוכניתך, ערכו של A יהיה 1, ערכו של B יהיה 2, וערכו של C יהיה 3. הדר והרץ את התוכנית. מהו הקלט המבוקש?
9. מטרת התוכנית היא להדפיס את המספר הראשון, ובשורה מתחתיו להדפיס את המספר השני ואת המספר השלישי. אם מעונינים שבסיום משפט פלט יוצגו הנתונים הבאים בשורה חדשה עלינו להשתמש ב `writeln`.
- שנה הוראה אחת בתוכניתך על מנת שהנתונים יודפסו בהתאם למטרה. הדר והרץ את התוכנית, עבור אותם נתוני קלט.
10. הוסף לפני ההוראה `write(a)`, בשורה נפרדת את ההוראה `writeln`, הדר והרץ, מה תפקידה של הוראה זו?
11. השלם: עבור הקלט 8 9 6 יהיה הפלט:

סיכום

1. אם אנו מעונינים שבסיום ביצוע משפט קלט או משפט פלט יוצגו או יקראו הנתונים משורה חדשה, עלינו להשתמש בפקודות `readln` ו `writeln`.
2. כדי להעביר את ה"ראש המדפיס" לראשית שורה חדשה יש להשתמש בהוראה `writeln` (ללא פרמטרים). אם הראש המדפיס נמצא בתחילת שורה ריקה, תגרום הוראת `writeln` ליצירת שורת רווח.

משימה 4

מטרה: חשיבות בהירות הדו-שיח בין משתמש בתכנית למחשב.

1. התבונן בתכנית Game.

```
program Game (input , output) ;  
var N, M : integer ;  
    Guess, Mult: integer ;  
begin  
    readln ( N , M ) ;  
    readln ( Guess ) ;  
    Mult := N * M ;  
    writeln (Guess);  
    writeln (Mult);  
end.
```

התכנית קולטת שני מספרים וניחוש של המכפלה שלהם. התכנית מחשבת את מכפלת המספרים שנקלטו, ופולטת את ניחוש המכפלה שלהם ואת המכפלה הנכונה.

2. טען את הקובץ Game.PAS.

והדר את התכנית Game.

3. הרץ את התכנית Game עם נתונים כלשהם, לפי בחירתך.

שתי בעיות עולות בזמן הרצת התכנית:

1. קשה למשתמש לדעת מתי להקיש נתונים וכמה נתונים להקיש.

2. קשה לקרוא ולהבין את הפלט.

שִׁים ♥

כאשר מתבצע משפט `readln`, מושהה ביצוע התכנית עד להקשת נתונים על לוח המקשים.

רצוי כי בתכנית תופיע הודעה על השלב בו יש להקיש את הנתונים. הודעה כזאת ניתנת ליישום באופן הבא:

```
writeln ('הקש שני מספרים');  
readln ( N , M );
```

כך, לפני ביצוע משפט `readln`, תופיע על המסך ההודעה הקש שני מספרים ורק לאחר מכן יתבצע המשפט `readln`.

4. תקן את קליטת הנתונים בתכנית Game כך שהדו-שיח בין המשתמש בתכנית למחשב יהיה יותר ברור.

לא תמיד המשתמש בתכנית מכיר תכניות פסקל או יודע כיצד להריצן בסביבת הפיתוח ולכן יש להנחות אותו במפורט מה עליו לעשות, למשל כך:

```
writeln ('הקש שני מספרים שלמים כאשר תו רווח מפריד ביניהם') ;  
writeln ('לאחר מכן, הקש Enter');  
readln ( N , M );  
writeln ('נחש את המכפלה שלהם') ;  
writeln ('הקש את הניחוש שלך והקש Enter');  
readln ( Guess );
```

5. הרץ את התכנית Game שוב כדי להיווכח ביתרונות ההודעות לפני ביצוע משפט `readln`.

6. תקן את התכנית Game כך שהפלט שלה יופיע בליווי הסבר.
למשל, במקום
writeln (Guess);
יופיע המשפט
writeln (Guess, 'ניחוש הוא');

7. הדר והרץ את תוכניתך כדי לוודא את נכונותה.

8. שמור את התכנית Game בתקליטון.

סיכום ולקחים

- ★ במהלך ביצוע תכנית מתנהל דו-שיח בין משתמש בתכנית למחשב, המורכב מביצוע הוראות קלט והוראות פלט.
- ★ בזמן הרצת תכנית יש להודיע למשתמש בה מתי יש להקיש נתונים ואת מספרם.
- ★ יש ללוות את הפלט בהודעות המסבירות אותו.

משימה 5

מטרה: פתרון בעייה על-ידי תוכניות עם משפטי השמה.

התבונן בשאלה 7 מתוך השאלות הנוספות בספר הלימוד שלך.

לנוחותך העתקנו אותה לכאן:

מחיר כרטיס כניסה לבריכת השחיה העירונית הוא 20 ש"ח למבוגר ו 12 ש"ח לילד. פתח אלגוריתם, אשר הקלט שלו הוא מספר המבוגרים ומספר הילדים שביקרו בבריכה ביום מסוים, והפלט שלו הוא סכום הכסף לגבייה עבור הכרטיסים. יישם את האלגוריתם בשפת פסקל.

1. הקלד את ישום הפתרון (התוכנית) עבור בעיה זו.

2. הדר עד שלא תהינה שגיאות תחביר.

3. הרץ את תוכניתך וודא את נכונות תוכניתך.

משימה 6

מטרה: טיפול בשגיאות תחביר הקשורות בהצהרת משתנים.

1. התבונן בתוכנית השגויה הבאה.

2. הקלד את התוכנית/טען את הקובץ smml.pas.

```
program sum (input, output);  
  {הקלט: שלושה מספרים שלמים}  
  {הפלט: סכום המספרים ומכפלתם}  
var  
  Sum , Mult: integer ;  
begin  
  writeln ('הקש שלושה מספרים שלמים');  
  readln (N1, N2, N3) ;  
  Sum := N1+ N2+ N3 ;  
  Mult := N1* N2* N3 ;  
  writeln ('Sum:=' ,Sum) ;  
  writeln ('Mult =' ,Mult) ;  
end.
```

3. הדר את התוכנית.

המהדר מודיע על מזהה בלתי מוכר: Unknown identifier.

המהדר נתקל במזהה בלתי מוכר: N1. המשתנה N1 אינו מוכר למהדר משום שלא הוצהר בהצהרת המשתנים של התוכנית (משפט var).

4. בדוק: האם יש עוד משתנים שלא הוצהרו?

תקן את הצהרת המשתנים, והדר את תוכניתך.

המהדר מודיע כי קיימים בתוכנית שני מזהים בשם Sum.
Duplicate identifier (Sum).

שים לב כי שם התוכנית וכן שמו של אחד המשתנים הוא Sum.

5. תקן את שם תוכניתך ל-SumMult.
הדר את תוכניתך עד שלא יהיו בה שגיאות תחביר.

6. הרץ את תוכניתך עבור הקלטים שבטבלה והשלם את הטבלה לפי הפלט שיתקבל:

פלט		קלט		
SUM	MULT	N1	N2	N3
		5	3	1
		2	6	7
		3	-3	3

7. השלם: עבור קלט של שלושה מספרים A, B ו-C יהיה הפלט:

8. כעת הרץ את התוכנית עבור הקלט 8 12 3.4 לפי הסדר משמאל לימין.
ההרצה מופסקת לאחר קריאת הנתון הראשון ומתקבלת ההודעה הבאה על שגיאת ריצה:

Invalid numeric format

השגיאה נגרמה משום שהמשתנה N1 מטיפוס שלם בעוד שהנתון שהוקש מטיפוס ממשי.

9. כעת הרץ את התוכנית עבור הקלט 8 12 3.0 לפי הסדר משמאל לימין.

ההרצה מופסקת שוב לאחר קריאת הנתון הראשון, כיוון שגם 3.0 הוא מטיפוס ממשי.

האם מתקבלת אותה הודעה?

10. הרץ את התוכנית עבור הקלט 2 4 36000.

מהו הפלט שקיבלת? _____

האם הוא הגיוני?

מדוע?

המספר 36000 גדול מהמספר מטיפוס שלם (INTEGER) הגדול ביותר שאפשר להשתמש בו. ערך המספר הגדול ביותר משתנה ממחשב למחשב והוא תלוי במספר הסיביות המוקצות לתא בזיכרון הראשי.

במחשבים תואמי PC, מורכב תא משתנה מטיפוס שלם מ-16 סיביות:

15 סיביות למספר וסיבית אחת המציינת אם הוא חיובי או שלילי.

15 סיביות יכולות ליצג תחום ערכים הנע בין המספר 32768- למספר 32767.

סיכום לקחים

1. למשתנה ולתוכנית לא יכול להיות אותו השם.
2. כל משתנה המופיע בתוכנית חייב להיות מוצהר בהצהרת משתנים של התוכנית (משפט var).
3. יש להקפיד כי בזמן הצרת תוכנית לא יושם במשתנה מטיפוס שלם ערך החורג מתחום הערכים האפשריים לטיפוס שלם במחשב שאיתו עובדים.
4. במשתנה מטיפוס שלם אי אפשר להשים מספר ממשי.

משימה 7

מטרה: הכרת מנפה השגיאות (trace ו watch) המחשת טבלת מעקב.

1. התבונן בתוכנית הבאה.

```
program chap3-7(input, output);
var
  Number1 , Number2 , Digit1, Digit2 : integer ;
begin
  writeln ('הקש שתי ספרות');
  readln (Digit1 , Digit2) ;
  Number1 := Digit1*10+ Digit2 ;
  Number2 := Digit2*10+ Digit1 ;
  writeln ('numbers are: ',Number1:4 ,Number2:4)
end.
```

2. הקלד את התוכנית/טען את הקובץ digit.pas, והדר אותה.

3. ברצוננו למלא את טבלת המעקב בקלות.

לשם כך נריץ את התוכנית בשלבים, צעד אחר צעד, שורה אחר שורה.

בחר בתת תפריט RUN

בחר בהוראה TRACE INTO. התבונן בקיצור המופיע לידה (F7 או F8).

מהו הקיצור המופיע?

לאחר שבחרת בהוראה TRACE INTO התרחש שינוי על המסך.

תאר את השינוי:

4. מנפה השגיאות נעצר לפני ביצוע השורה המוארת.

הקש את הקיצור או בחר שוב בתת-תפריט RUN ובהוראה TRACE INTO.
איזו שורה מוארת כעת?

שים ♥ בכל פעם שתקיש את הקיצור (או תבחר באופציה TRACE INTO מתוך תת-תפריט RUN, יבצע המחשב **שורה** אחת מהתוכנית ויאיר את השורה הבאה לביצוע.

5. נצפה בערכי המשתנים בזמן הרצת התוכנית. ניתן לבצע זאת על ידי הכנסתם לחלון

ה-watch (חלון הצפייה) :

בחר בתת תפריט watch/break

בחר בהוראה add watch

הקש Digit1.

בחלון ה-watch הופיע המשתנה ולידו ערכו הנוכחי.

6. הכנס באופן דומה את המשתנים Number1, Number2, Digit1 לחלון ה-watch.
העתק את ערכי המשתנים:

Digit1:

Digit2:

Number1:

7. נריץ את התוכנית עבור הקלט 3 6 משמאל לימין

הרץ את התוכנית בשלבים, על ידי הקשת הקיצור (שמצאת בסעיף ד) ומלא במקביל את טבלת המעקב, בהתאם לערכי המשתנים בחלון ה-watch.
זכור: השורה המוארת עוד לא בוצעה.

תזכורת: כדי לצפות בפלט: בחר בתת תפריט DEBUG

בחר בהוראה USER SCREEN

המשפט הבא לביצוע	Digit1	Digit2	Number1	Number2	פלט
Writeln ('שתי ספרות')					
readln (Digit1, Digit2)					
Number1 := Digit1*10 ..					
Number2 := Digit2*10 ..					
writeln ('numbers . . .					

8. השלם: התוכנית מקבלת כקלט

התוכנית מציגה כפלט:

סיכום

1. מנפה השגיאות הוא כלי שמספקת לנו סביבת הפיתוח כדי לעקוב אחרי ביצוע תוכנית. המעקב מתבצע על מנת לאתר שגיאות שלנו. המעקב מתבצע באמצעות התבוננות במשתנים בחלון ה watch, וביצוע התוכנית בשלבים.

2. לביצוע שורה אחת בתוכנית:

בחר בתת תפריט Run

בחר בהוראה Trace into

או הקש את הקיצור שמצאת בסעיף ג.

3. לצפייה בערכי משתנה בזמן ביצוע התוכנית:

בחר בתת-תפריט Watch/Break

בחר בהוראה Add Watch

הקש את המשתנה המבוקש.

טוב לדעת

אם הנך באמצע תהליך ניפוי ורצונך להפסיקו או להתחילו מחדש.

בחר בתת-תפריט Run

בחר בהוראה Reset

או הקש את הקיצור המופיע ליד ההוראה Reset.

משימה 8

מטרה: טיפול בפלט של מספר מטיפוס ממשי.

לפניך שאלה 3 מהתרגילים המסכמים בפרק 3 שבספר הלימוד.

הקלט הוא טמפרטורה הנתונה במעלות פרנהיט.

הפלט הוא ערך הטמפרטורה במעלות צלסיוס. ההמרה מיצוג של טמפרטורה במעלות

פרנהיט (F) למעלות צלסיוס (C), מתבצעת על-ידי הנוסחה: $C = 5/9 * (F - 32)$.

נתונה תוכנית הממירה טמפ' הנתונה במעלות פרנהיט למעלות צלסיוס.

1. השלם את התוכנית:

```
program LabConvert (input, output);  
var  
    F : real ; { הטמפ' במעלות פרנהיט }  
    C : real ; { הטמפ' במעלות צלסיוס }  
begin  
    writeln ('הקש טמפ במעלות פרנהיט') ;  
    readln  
    C :=  
    writeln (C, 'ההמרה למעלות צלסיוס היא')  
end.
```

2. הקלד את התוכנית והדר אותה עד שלא יהיו שגיאות תחביר.

3. הרץ את תוכניתך עבור הקלט 97.0.

4. מהו הפלט שנתקבל?

זו אינה טעות, אך הפלט אינו בהיר ויש לפרשו. למשל:

א. $5.675445\text{E}+07$ משמעו $5.675445 \cdot 10^7$, כלומר 56,754,450

ב. $9.47\text{E}-03$ משמעו $9.47 \cdot 10^{-3}$, כלומר $9.47 \cdot 0.001$, כלומר 0.00947

5. פרש את הפלט שקיבלת

6. ברצוננו להפוך את הפלט לבהיר יותר. לשם כך נקבע את רוחב שדה ההדפסה:

הרחב את תוכניתך כך שרוחב שדה ההדפסה של המשתנה C, יהיה 7, באופן הבא:

`writeln (C:7)`

הרץ שוב עבור הקלט 97.

מהו הפלט שנתקבל?

7. ניתן לקבוע את מספר הספרות מימין לנקודה, על-ידי הגדרת רוחב-השבר.

הרחב את הדפסת המשתנה C באופן הבא:

`writeln (C:7:2)`

הרץ עבור הקלט 97.0.

מהו הפלט שנתקבל?

מספר הספרות מימין לנקודה הוא 2, משמע רוחב השבר הוא 2.

רוחב השבר הינו מספר הספרות הרצוי מימין לנקודה.

8. שנה את רוחב שדה ההדפסה של C ל 9 ואת רוחב השבר ל-3.

הדר והרץ, עבור הקלט 97.0.

מהו הפלט שנתקבל?

על המסך? C כמה תווים שה"כ תופס ערך המשתנה
כמה תווים שה"כ תופסות הספרות שמימין לנקודה?

סיכום

1. כדי לשפר את בהירות הפלט של מספר מטיפוס ממשי, יש להקפיד על רוחב שדה
הדפסה נאות ורוחב שבר.

רוחב שבר הינו מספר הספרות מימין לנקודה הרצוי.
הגדרת רוחב שדה ורוחב שבר היא על ידי המשפט:

(רוחב שבר: רוחב שדה ההדפסה: משתנה) `write/writeln`

עבור רוחב שדה הדפסה m ורוחב שבר n , $m > n$, המשתנה יודפס בעזרת m
תווים, מתוכם n מימין לנקודה.

משימה 9

מטרה: תרגול div , mod .

הרץ את שאלה 3.35 שבספר הלימוד.

לנוחותך, העתקנו אותה לכאן.

פתח וישם בשלבים אלגוריתם, אשר הקלט שלוד הוא פרק זמן הנתון בדקות והפלט שלו הוא מרכיבי השעות והדקות בפרק הזמן הנתון. למשל עבור הקלט 90 יהיה הפלט 30 1 שפירושו שעה ושלושים דקות, עבור הקלט 30 יהיה הפלט 0 30, ועבור הקלט 300 יהיה הפלט 0 5.

1. הבא 3 דוגמאות מייצגות של קלט.

2. הקלד את ישום הפתרון (התוכנית) והדר אותו עד שלא יהיו שגיאות תחביר.

3. הרץ את תוכניתך עבור הקלטים המופיעים בשאלה וודא שהפלט המתקבל נכון.

4. אם יש צורך (הפלט אינו נכון) נפה את השגיאות מתוכניתך, על-ידי שימוש במנפה השגיאות (trace , watch).

5. הרץ את תוכניתך עבור הקלטים שבחרת בסעיף 1 ומלא את הטבלה:

פרק הזמן בדקות	פרק הזמן בשעות ובדקות

6. רשות

הרחב את תוכניתך כך שעבור קלט שהוא פרק זמן בדקות, יהיה הפלט מרכיבי הימים השעות והדקות שבפרק הזמן הנתון.

פרק 4

משימה 1

מטרה: תיקון שגיאות תחביר בתכנית הנובעות מזימון לא נכון של פונקציות תקניות ותיקון התכנית כך שתשיג את מטרתה.

1. התבונן בתכנית השגויה MyAbs

```
program MyAbs (input , output) ;  
var Num: integer ;  
begin  
  writeln ('הקש מספר שלם ');  
  readln (Num);  
  writeln ( Num, ' המספר שנקלט הוא: ');  
  abs (Num);  
  writeln ('          :ערכו המוחלט הוא:');  
  writeln (Num);  
end.
```

ניתן לנחש כי מטרת התכנית היא לקלוט מספר ולפלוט את ערכו המוחלט.

2. טען את הקובץ absolute.pas.

הדר את התכנית MyAbs .

המהדר מודיע על פניה לא חוקית למשתנה:

Invalid variable reference

abs (Num) הינו ביטוי בשפת פסקל ואינו משפט אשר יכול להופיע כפי שהוא בתכנית פסקל.

את ערך הביטוי abs(Num) אפשר לפלוט (באמצעות משפטי write ,writeln) או לשים במשתנה (באמצעות משפט השמה).

3. תקן את התכנית MyAbs על-פי המוצע בהסבר לעיל.

4. הדר את התכנית MyAbs עד שלא יהיו בה שגיאות תחביר.

5. בחר שתי דוגמאות קלט מיצגות.

6. הרץ את התכנית MyAbs, עבור הקלטים שבחרת ומלא את הטבלה:

קלט	פלט

סיכום ולקחים

זימון פונקציה תיקנית הינו ביטוי בשפת פסקל ומותר להשתמש בו בכל מקום בתכנית בו יכול להופיע ביטוי, למשל: מימין לסימן = : במשפט השמה או ברשימת הפלט של משפט write.

משימה 2

לפניך בעיה ופתרון לא מושלם שלה. עליך להשלים את הפתרון.

1. קרא את הבעיה ואת תחילת הפתרון:

בעיה

פתח אלגוריתם שהקלט שלו הוא מספר שלם דו ספרתי חיובי והפלט שלו הוא הערך המוחלט של הפרש ספרותיו.

פתרון

שלב א': בחינת דוגמאות קלט-פלט:

עבור הקלט 18 יהיה הפלט 7,

ועבור הקלט 52 יהיה הפלט 3.

שלב ב': תת-משימות:

קליטת המספר

פירוק המספר לספרותיו

חישוב הערך המוחלט של הפרש ספרותיו

הצגה כפלט של הערך המוחלט של הפרש הספרות

שלב ג': בחירת משתנים ותיאור תפקידם:

Num : integer ; {המספר הדו ספרתי}

Units, Tens: integer ; {ספרת האחדות וספרת העשרות}

Diff: integer; {ההפרש בין ספרת האחדות לספרת העשרות}

AbsDiff:integer ; {הערך המוחלט של ההפרש}

שלב ד': אלגוריתם:

קלוט את המספר ב-Num

חשב $\text{Num mod } 10$ והשם ב-Units

חשב _____ והשם ב- Tens

חשב _____ והשם ב- Units

חשב _____ והשם ב- Diff

חשב _____ והשם ב- AbsDiff

הצג כפלט את AbsDiff.

2. השלם את האלגוריתם.

3. כתוב את התוכנית והדר אותה. הקפד גם על דו-שיח ברור בין המחשב למשתמש.
(כך שלפני קליטת הנתונים תוצג למשתמש הודעה, אילו נתונים עליו להקיש).

4. הרץ את תוכניתך ומלא את הטבלה הבאה:

קלט	פלט
31	
99	
29	

5. הרחב את פתרוןך כך שעבור קלט של מספר שלם דו-ספרתי (חיובי או שלילי) יציג כפלט את הערך המוחלט של הפרש ספרותיו. למשל עבור הקלט 31- יתקבל הפלט 2.

פרק 5

משימה 1

מטרה: הכרת הביטוי הבוליאני, תרגיל משפט if, העמקת מיומנות שימוש במנפה השגיאות: שימוש ב GotoCursor.

התבונן שוב בבעיה 1, שבפרק 5.

לנוחיותך, העתקנו אותה שוב:

בעיה 1

בכתה אשר בה ישנם בנים ובנות מעוניינים לסדר את התלמידים במעגל כך שכל בן יעמוד בין שתי בנות וכל בת תעמוד בין שני בנים, אך הדבר אינו תמיד אפשרי, אלא תלוי במספרי הבנים והבנות בכתה. פתח וישם אלגוריתם אשר הקלט שלו הוא מספר הבנים ומספר הבנות בכתה והפלט שלו הוא הודעה המציינת האם ניתן או לא ניתן לסדר את תלמידי הכתה בן-בת לסירוגין במעגל.

לפניך ישום הפתרון של בעיה 1

```
program circle(input,output);
```

```
{תכנית לבדיקה האם ניתן לסדר תלמידים במעגל, בן-בת לסירוגין}
```

```
var
```

```
  Boys : integer ;
```

```
  Girls : integer ;
```

```
begin
```

```
writeln ('הקש מספר בנים ומספר בנות:')
```

```
readln (boys , Girls);
```

```
if Boys = Girls then
```


writeln (ניתן לסדר את התלמידים במעגל, בן-בת לסירוגין)

else

writeln ('לא ניתן לסדר את התלמידים במעגל, בן-בת לסירוגין')

end.

א. הקלד את התוכנית/טען את הקובץ circle.pas.

כעת נריץ את התוכנית בשלבים :

ב. בעזרת החיצים, הבא את הסמן לתחילת השורה בה מופיע התנאי `if boys = girls`

. בחר בתת-תפריט RUN

. בחר הוראה GOTO CURSOR.

כאשר תתבקש להקליד נתונים, הקלד 5 5.

ג. מהו השינוי שחל בשורה זו?

המחשב מבצע את פקודות התוכנית עד לשורה זו.

שים ♥ ששורה זו טרם בוצעה.

ד. הכנס לחלון ה WATCH את המשתנים boys ו girls.

(בחר בתת-תפריט BREAK/WATCH,

בחר בהוראה Add watch.

ה. הכנס לחלון ה WATCH את התנאי `boys = girls`.

מה מופיע בחלון ה WATCH ?

ו. בצע TRACE עד לסיום התוכנית. על אילו שורות דילג המחשב?

מדוע?

ז. כעת נריץ את התוכנית בשלבים עם הקלט 20 22.
בצע את התוכנית עד לשורה בה מופיע התנאי `if boys = girls`
(בעזרת `GOTO CURSOR`).
מה מופיע בחלון ה `WATCH` ?

ח. בצע `TRACE` עד לסיום התוכנית.
על אילו הוראות דילג עתה המחשב?
מדוע?

סיכום

עבודה עם מנפה השגיאות

1. כדי לבצע תוכנית עד שורה מסוימת (לא כולל) :
הבא את הסמן אל השורה הרצויה.
בחר בתפריט `RUN`
בחר בתת-תפריט `GOTO CURSOR`
המחשב יבצע את התוכנית עד לשורה המוארת.

2. ניתן לבצע `WATCH` גם על ביטוי בוליאני.

פסקל

3. כאשר מופיע `if` בוחר המחשב איזו הוראה לבצע ואיזו לא, לפי קיום או אי קיום התנאי.

משימה 2

1. הקלד את התוכנית השגויה הבאה:

```
program Wrong (input , output);
var
  A , B : integer ;
  Min : integer ;
begin ('הקש שני מספרים שלמים:');
  writeln ('הקש שני מספרים שלמים:');
  readln (A , B) ;
  if A > B then
    Min := B
  else
    Min := A ;
  writeln(Min);
end.
```

2. הדר את התוכנית.

3. המהדר מודיע לך על SYNTAX ERROR והסמן מורה לך על ה **else**.

הסיבה לכך היא שלפני **else** אסור לרשום ; .

תקן את התוכנית. הדר עד שלא יהיו בה שגיאות תחביר.

4. מה מטרת התוכנית?

5. מה יהיה הפלט עבור הקלט 6 6? הרץ ובדוק.

6. הבא שתי דוגמאות קלט מייצגות, הרץ איתן את התוכנית ומלא את הטבלה הבאה.

קלט	פלט	
דוגמא 1	7 3	
דוגמא 2	3 7	

סיכום

פסקל

1. מבנה קטע תוכנית עם תנאי:

```

if condition then
    statement1
else
    statement2 ;

```

שים לב! בסיום הפקודה, שמיקומה מעל ה- ELSE, אין ; .

מתבצעת בזמן ההרצה.

משימה 3

מספר אשר ניתן לקרוא אותו משמאל לימין ומימין לשמאל נקרא פלינדרום. למשל 787 פלינדרום ו 788 איננו פלינדרום.

1. פתח אלגוריתם אשר הקלט שלו הוא מספר שלם חיובי תלת ספרתי, והפלט שלו הוא הודעה האם המספר הנתון הוא פלינדרום.

2. ישם את האלגוריתם על ידי תוכנית פסקל.

3. הבא שתי דוגמאות קלט מייצגות, הרץ ורשום את הפלט בטבלה.

קלט	פלט	
		דוגמא 1
		דוגמא 2

משימה 4

מטרה: העמקת המיומנות במנפה השגיאות; הכנסת ביטוי בוליאני ל-watch, מחיקת ביטוי מה-watch.

התבונן שוב בבעיה 2, שבפרק 5.
לנוחיותך, העתקנו אותה לכאן:

בעיה 2

תלמידי כיתות י', בבית הספר יוצאים לטיול שנתי באוטובוסים, ויש להזמין מספר אוטובוסים מתאים להסעת התלמידים, כך שלכל תלמיד יהיה מקום ישיבה ומספר האוטובוסים יהיה קטן ככל האפשר.
פתח וישם אלגוריתם אשר הקלט שלו הוא מספר התלמידים בכיתות י' ומספר המושבים באוטובוס, והפלט שלו הוא מספר האוטובוסים שיש להזמין.

לפניך ישום הפתרון של בעיה 2

```
program busorder (input , output);
{תוכנית לחישוב מספר אוטובוסים להזמנה}
var
  Students      : integer ;
  SeatPerBus    : integer ;
  Buses         : integer ;
  Leftover      : integer ;
begin
  writeln ('הקש מספר תלמידים ותפוסת אוטובוס') ;
  readln (Students, SeatPerBus) ;
  Buses := Students div SeatPerBus ;
  Leftover := Students mod SeatPerBus ;
```

```

if Leftover > 0 then
    Buses := Buses + 1 ;
writeln (Buses , ' מספר האוטובוסים להזמנה הוא ')
enend

```

1. הקלד את התוכנית/ טען את הקובץ buses.pas.

2. הכנס לחלון ה-WATCH את כל המשתנים שבתוכנית.
(תזכורת)

בחר בתת-תפריט Break/Watch
בחר בהוראה AddWatch.

3. הכנס לחלון ה-WATCH את הביטוי $leftover > 0$.
בחלון ה-WATCH יופיעו המשתנים עם הערות can not access this symbol, או עם ערכים שגויים, שכן עוד לא השמת ערכים נכונים למשתנים. השמת הערכים

4. הרץ את התוכנית בשלבים, עבור הקלט 220 40, על ידי ביצוע TRACE, ומלא את טבלת המעקב לאחר כל ביצוע של פקודה. (העתק את הערכים מחלון ה-WATCH). שים לב שהשורה המוארת הינה השורה הבאה לביצוע, כלומר עוד לא בוצעה.

פלט	Buses	LeftOver > 0	Left Over	Seat PerBus	Students	הפקודה הבאה לביצוע
						Writeln (יחקש מספר תלם . . .) ;
						readln (Student, SeatPerBus)
						Buses := Students div . .
						LeftOver := students mod ..
						if Leftover >0 then
						writeln (Buses , ' . . . ')

5. כעת הרץ את התוכנית בשלבים עבור הקלט 20 220, ומלא את טבלת המעקב הבאה.

פלט	LeftOver > 0	Buses	Left Over	Seat PerBu	Students	הפקודה הבאה לביצוע
						WriteLn (זקש מספר חלם . . .) ;
						readLn (student, SeatPerBus)
						buses := students div ..
						leftover := students mod ..
						if leftover > 0 then
						buses := buses + 1 ;
						writeln (buses , ' . . . ')

6. על איזו שורה דילג המחשב?

מדוע?

7. הפוך את חלון ה-WATCH לחלון הפעיל. (על ידי F6).

שים לב לשורה התחתונה על המסך, שהשתנתה.

הבא את הסמן אל הביטוי $leftover > 0$ ומחק אותו על-ידי DEL.

כך מוחקים שורה מחלון ה-WATCH.

סיכום

מנפה השגיאות

1. F6 הופך את החלון הפעיל ללא פעיל, ואת החלון שאינו פעיל לפעיל.

2. מחיקת שורה מחלון ה-WATCH הינה על ידי:

א. הפיכת חלון ה-WATCH לחלון הפעיל.

ב. הבאת הסמן לשורה אותה רוצים למחוק.

ג. הקשת DEL.

פסקל

3. קיים ל IF מבנה מצורת

if condition then statement ;

משימה 5

1. הקלד את התוכנית הבאה/טען את הקובץ. dig.pas

```
program Digit (input , output);  
var  
Num, NumSucc , NumPred  : integer ;  
begin  
  readln (Num) ;  
  NumSucc := Num + 1 ;  
  NumPred := Num - 1 ;  
  writeln ('num =' , Num, 'Succ=' ,NumSucc, 'pred=' , NumPred)  
end.
```

2. הדר את התוכנית עד שלא יהיו שגיאות תחביר.

3. הרץ את התוכנית עבור הקלטים

0

4

9

ומלא את הטבלה.

קלט (Num)	פלט
0	
4	
9	

4.4. מה מטרת התוכנית?

5. כעת שנה את התוכנית כך שתבצע את מטרתה ותשתמש בפונקציות succ ו pred

על מנת להשים ערכים ב NumPred ו NumSucc .

$\text{NumSucc} := \text{succ}(\text{Num})$

$\text{NumPred} := \text{pred}(\text{Num})$

6. הרץ את תוכניתך, עבור אותם הקלטים שבסעיף 3 וודא שאכן מבצעת התוכנית

את מטרתה.

7. שנה את תוכניתך כך שהקלט יהיה תו והפלט יהיה התו שנקלט, התו שלפניו

והתור שאחריו.

8. הדר את תוכניתך, הרץ אותה עבור הקלטים $d, a, 2, 0$ ומלא את הטבלה הבאה:

קלט	פלט
d	
a	
2	
0	

9. השלם:

הפונקציות Pred ו Succ פועלות על משתנים מטיפוס

ועל משתנים מטיפוס

משימה 6

הרץ את שאלה 5.20 מספר הלימוד.

לנוחותך, העתקנו אותה לכאן שוב.

פתח אלגוריתם וישמו כתוכנית בפסקל, כאשר הקלט הוא שתי אותיות והפלט הוא הדפסתן בשתי שורות נפרדות; בשורה הראשונה בסדר עולה ובשורה השניה בסדר יורד.

משימה 7

1. התבונן בתוכנית הבאה:

```
program Bonus (input , output);  
var  
    Grade    : integer;  
    Units    : integer;  
    GradeWithBonus : real ;  
begin  
    writeln(' יהקש את מספר יחידות הלימוד ואת הציון ');  
    readln (Units, Grade);  
    if Grade >= 65 and Units = 5 then  
        begin  
            writeln ('a good student');  
            GradeWithBonus := Grade*1.3  
        end  
    else  
        GradeWithBonus := Grade;  
    writeln(GradeWithBonus:6:2, 'הציון עם הבונוס הוא')  
end.
```

2. מה מטרת התוכנית?

3. טען את הקובץ Bonus.Pas

הדר את התוכנית.

נתקלת בהודעת המהדר : THEN expected

הסמן עומד על הביטוי הבוליאני השני.

הסיבה לכך היא שלא הקפת בסוגריים את שני הביטויים
הבוליאניים ולכן לא ידע המהדר שהתנאי כולל את שני
הביטויים.

הדר את התוכנית עד שלא יהיו בה שגיאות תחביר.

4. כעת נריץ התוכנית בשלבים עם הקלט 5 68.

בעזרת GO TO CURSOR עצור את הרצת התוכנית בשורת התנאי.
(הבא את הסמן לשורה זו, בחר Run, בחר GOTO CURSOR).

5. הכנס את הביטוי הבוליאני $\text{Grade} \geq 65$ לחלון ה-WATCH.

הכנס את הביטוי הבוליאני $\text{Units} = 5$ לחלון ה-WATCH.

הכנס את הביטוי הבוליאני המורכב $(\text{Grade} \geq 65) \text{ AND } (\text{Units} = 5)$.

6. מה מופיע בחלון ה-WATCH?

7. הרץ שוב את התוכנית עד לשורת התנאי, עבור הקלט 4 95.

8. מה מופיע בחלון ה-WATCH?

מה פלט התוכנית?

9. זכור: כדי שיתקיים תנאי מורכב משני תנאים פשוטים שביניהם
קשר and, צריכים שני התנאים הפשוטים מהם הוא מורכב
להתקיים.

סיכום

פסקל

אם מופיע בתוכנית תנאי מורכב, יש להקיף בסוגריים עגולים כל ביטוי בוליאני
פשוט.

משימה 8

מטרה: הבנת הקשר or.

במשימה זו עליך ליישם לתוכנית פסקל, אלגוריתם נתון:

הקלט של האלגוריתם הבא הוא מספר דו ספרתי חיובי והפלט שלו הוא הודעה האם הספרה 3 מופיעה במספר זה או לא.

משתנים עיקריים:

Number - המספר הדו ספרתי

Units - ספרת האחדות

Tens - ספרת העשרות

אלגוריתם

קלוט את Number

פרק אותו לספרת האחדות ולספרת העשרות שלו והשם ב-Units וב-tens בהתאמה.

אם $(Units = 3)$ או $(Tens = 3)$ אזי

הצג כפלט: הספרה 3 מופיע במספר

אחרת

הצג כפלט: הספרה 3 אינה מופיעה במספר.

1. ישם את האלגוריתם הנתון על-ידי תוכנית בפסקל.

2. בחר 4 דוגמאות קלט מיצגות, הרץ את התוכנית עבור הדוגמאות שבחרת ומלא בעזרתן את הטבלה הבאה.

הקלט Number	Tens = 3	Units = 3	הפלט
(1)			
(2)			
(3)			
(4)			

3. האם יש מקום להוסיף עוד דוגמה מיצגת? נמק

פרק 6

משימה 1

מטרה: חזרה על ההוראות של מעקב אחר ביצוע תכנית וצפייה בערכי ביטויים.
הדגמה של תהליך ניפוי שגיאות באמצעות מנפה השגיאות. הבהרת המושג
נכונות תוכניות.

1. התבונן בתכנית Order.

```
program Order (input , output) ;  
var A, B, C, Temp: Char;  
begin  
  writeln ('הקש שלושה תווים');  
  readln (A, B, C);  
  if A > B then  
    begin  
      Temp := A; A := B; B := Temp  
    end;  
  if B > C then  
    begin  
      Temp := B ; B := C ; C := Temp  
    end;  
  writeln (A : 2 , B : 2, C : 2)  
end.
```

קלט התכנית Order הוא שלושה תווים. מטרת התכנית היא להדפיס את
התווים בסדר עולה.

2. טען את הקובץ ord.pas.

הדר את התכנית Order.

3. שמור את התכנית Order בתקליטון המעבדה שלך.

4. הכנס את המשתנים A, B, C ו-Temp לחלון ה-watch:

תזכורת: הכנסת משתנה לחלון ה-watch

בחר בתת תפריט watch/Break

בחר בהוראה Add watch

הקש את המשתנה.

נבחר בדוגמאות מייצגות של קלט לבדיקת נכונות האלגוריתם:

1. 'z' 'y' 'x'

2. 'y' 'z' 'x'

3. 'z' 'x' 'y'

4. 'x' 'y' 'z'

שים ♥ כי כאשר הנך מקיש קלט מטיפוס תווי יש להקיש אותו ללא הגרשיים.

שים ♥ כי במנפה השגיאות היחידה המוארת הקטנה ביותר היא שורה ולא משפט. אם מופיעים מספר משפטים בשורה אחת, כולם יבוצעו כתוצאה מהקש Trace Into או הקשת הקיצור.

הרץ את התוכנית עבור כל אחת מ 4 הדוגמאות ומלא את הטבלה הבאה.

6. חשוב על עוד שתי דוגמאות מייצגות של קלט שלא ניתנו לעיל.

א.

ב.

7. עקוב אחר ביצוע התכנית (באמצעות מפנה השגיאות) עבור הקלט: 'y' 'x' 'z' ועבור הקלט 'y' 'z' 'x'.

קלט	פלט	נכון (כן/לא)
z x y		
y z x		

האם הפלט שקיבלת הוא נכון?

שים ♥ כי יש עובדים בסביבת הפיתוח בעזרת מפנה השגיאות (כמו בטבלת מעקב), על מנת לבדוק נכונות של תכנית צריך לבחור דוגמה מייצגת של קלט טובה; בחירה טובה תביא לכך שנגלה שהתכנית שגויה.

שים ♥ כי אם לא היית בוחר בדוגמה מייצגת של קלט טובה, השגיאה לא הייתה מתגלית.

מפנה השגיאות מאפשר לפתח טבלת מעקב במחשב, במקום עם נייר ועפרון. השורה המוארת, יחד עם רשימת המשתנים, הינה מצב התכנית.

8. עקוב אחר ביצוע התכנית (באמצעות מפנה השגיאות) עבור הקלט 'x' 'z' 'y'.

9. התבונן בחלון ה-watch.

שים ♥ לערכי המשתנים A, B, C (בחלון הצפייה) לאחר ביצוע המשפט **if** השני.
שים ♥ שערכים אלו עדיין אינם ממוינים.

10. תקן את השגיאה בתכנית שכך שתהיה נכונה.

שים ♥ כי האחריות לתיקון מוטלת עליך, ומפנה השגיאות לא יכול לעזור לך בכך!

11. הדר את התכנית.

12. הרץ את התכנית של שתי הדוגמאות המייצגות של קלט עבורן היה הפלט לא נכון כדי לוודא את נכונותה.

13. שמור את התכנית על תקליטון המעבדה שלך.

סיכום

1. מנפה השגיאות וצפייה בחלון מאפשרים מעקב אחרי מצבי ביצוע של תכנית.
2. האחריות לבחירת דוגמאות מייצגות של קלט אשר יביאו לגילוי שגיאה בתכנית (אם היא קיימת), ותיקון השגיאה בתכנית מוטלת על כותב התכנית, ומנפה השגיאות לא יכול לעזור בכך.

פרק 7

משימה 1

מטרה: הכרת לולאות FOR.

1. התבונן שוב בבעיה 1 שבפרק 7. לנוחותך, העתקנו אותה שוב:

בעיה 1

יש להמיר רשימה של מחירים מייצוג בדולרים לייצוג בשקלים. פתח ויישם אלגוריתם אשר הקלט שלו הינו שער ההמרה מדולרים לשקלים ואחריו רשימה של ארבעה מחירים הנתונים בדולרים, והפלט שלו הוא הערך של כל אחד מארבעת המחירים בשקלים.

2. לפניך ישום הפתרון של בעיה 1:

```
program Convert (input,output);
{קלט: שער המרה מדולרים לשקלים, רשימה של 4 מחירים בדולרים}
{פלט: הערכים השקליים של 4 המחירים הנתונים בדולרים}

var
  Rate: real ; {שער המרה מדולרים לשקלים}
  DollarPrice: real ; {מחיר בדולרים}
  ShekelPrice: real ; {מחיר בשקלים}
  I: integer ; {משתנה בקרה}

begin
  writeln ('הקש שער ההמרה מדולרים לשקלים:');
  readln (Rate);
  for I := 1 to 4 do
```

```

begin
    writeln ('הקש מחיר בדולרים:');
    readln(DollarPrice);
    ShekelPrice := DollarPrice*Rate;
    writeln (ShekelPrice:6:2,' (ערכו של המחיר בשקלים הוא ');
end
end.

```

3. הקלד את התוכנית/טען את הקובץ Convert.pas.

4. הכנס את כל משתני התוכנית לתוך חלון ה-WATCH.

5. הרץ בשלבים את התוכנית עבור הקלט

```

3.04
1000.30  12.50  20.60  12.00

```

וענה תוך כדי כך על השאלות הבאות:

(קרא את השאלות לפני שאתה מתחיל את ההרצה בשלבים).

(להרצת שורה אחת:

בחר בתת-תפריט RUN

בחר בהוראה TRACE INTO

או הקש את הקיצור שכתוב ליד ההוראה TRACE INTO).

6. מהו ערכו של המשתנה I לאחר ביצוע המשפט **for I := 1 to 4 do** בפעם הראשונה?

בכמה גדל המשתנה I בכל פעם?

האם ערכו של המשתנה DollarPrice השתנה יחד עם ערכו של המשתנה I?

מתי השתנה ערכו של ShekelPrice?

7. האם ערכו של Rate משתנה במהלך ביצוע התוכנית? מדוע?

8. מהו המשפט הבא לביצוע לאחר ביצוע משפט ה end הראשון?

9. ספור כמה פעמים התבצעה הלולאה?

10. השלם את הטבלה, לפי פלט התוכנית שהרצת:

מחיר בדולרים	מחיר בשקלים
1000.30	
12.50	
20.60	
12.00	

משימה 2

מטרה: חשיבות עריכה ועימוד תוכנית, אתחול צוברים

לפניך תוכנית שגויה:

הקלט: מספר הציונים של תלמיד, ואחריו ציוני התלמיד

הפלט הרצוי: ממוצע ציוניו של התלמיד.

```
program Grades (input , output);
var
  NumberOfGrades : integer ;
  Sum, I , Grade  : integer ;
  Average          : real ;
begin
  Sum := 0 ;
  writeln ('הקש את מספר הציונים') ;
  readln (NumberOfGrades) ;
  writeln ('הקש את הציונים') ;
  for I := 1 to NumberOfGrades do
    readln (Grade) ;
    Sum := Sum + Grade ;
  Average := Sum / NumberOfGrades ;
  writeln (' Average', Average : 6:2)
end.
```

1. טען את הקובץ grade.pas.

2. הדר את התוכנית והרץ אותה עבור הקלט 100 85 78 98 4.

– מהו הפלט שנתקבל?

– האם זהו הפלט לו ציפית?

נסה למצוא את השגיאה. כדי לאתר שגיאות ביתר נוחות, עמד את התוכנית: רפד את התוכנית ברווחים כך שגוף הלולאה יהיה פנימה מראש הלולאה. שאר משפטי התוכנית, שאינם גוף הלולאה, יתחילו בדיוק מתחת לאות f של משפט ה-`for`.
 כעת הכנס את ערכו של `Sum` אל חלון ה-`watch`, והרץ את התוכנית בשלבים. מתי משתנה ערכו של `Sum`?

3. בעזרת הידע הזה תקן את השגיאה שבתוכנית, הדר אותה והרץ אותה עבור אותו הקלט שבסעיף 2.
 מה ממוצע הציונים?

4. מחק את השורה `Sum := 0`.
 (מחיקת שורה: הצב את הסמן על השורה שברצונך למחוק הקש `<ctrl>` ומבלי לשחרר את המקש, הקש `y`).

5. הדר את התוכנית והרץ אותה עבור הקלט שבסעיף 2.
 מה הפלט שנתקבל?

6. הרץ את התוכנית שוב עבור הקלט שבסעיף 2.
 א. מה הפלט שנתקבל?
 ב. האם הוא זהה לפלט שקיבלת בסעיף 5?
 ג. מדוע שינתה מחיקת שורה זו את הפלט?

המשפט `Sum := 0` **אתחל** את ערכו התחילי של המשתנה `Sum` בכל הרצה להיות 0. לכן נכנה משפט מסוג זה **משפט אתחול**.

7. החזר את משפט האתחול `Sum := 0` למקומו, הדר והרץ שוב עבור הקלט שבסעיף 2 מהו הפלט שנתקבל?

האם הוא זהה לפלט שנתקבל בסעיף 3 ולפלט שנתקבל בסעיף 5?

מדוע?

סיכום לקחים

1. בתוכנית עם צובר, חשוב לאתחל את הצובר.
2. הקפדה על עימוד נכון של התוכנית עוזרת לאתר שגיאות בקלות, והופכת תוכנית לקלה יותר להבנה.

משימה 3

יישם את פתרון שאלה 7.11 במעבדה. לנוחותך העתקנו אותה שוב:

קלוט מספר N וקלוט מחרוזת של תווים, אשר אורכה הוא N .

הצג כפלט את מספר התווים שהם אותיות גדולות ב-ABC.

רמז: זכור שהאותיות ה-ABC הגדולות מסודרות בקבוצת התווים של המחשב בסדר

עולה מ-A ל-Z. אותיות ה-abc הקטנות מסודרות אף הן, בנפרד מהגדולות, בסדר

עולה מ-a ל-z.

משימה 4

מטרה: לימוד נקודות עצירה, תרגול לולאת while.

התבונן שוב בבעיה 3, שבפרק 7 לנוחיותך העתקנו אותה שוב:

בעיה 3

בתכנית "הנוסע המתמיד" של חברת תעופה, ניתן לצבור מרחקי טיסה. נוסע אשר צובר למעלה מ-30,000 קילומטרים, זוכה בכרטיס טיסה חינם לארצות הברית. נתונה רשימת מרחקי טיסה של נוסע, אשר זכה בכרטיס טיסה חינם, ויש לנכות מרשימה זו את המרחקים שבתחילת הרשימה המסתכמים בלמעלה מ-30,000.

פתח ויישם אלגוריתם אשר הקלט שלו הוא רשימת מרחקי הטיסה אשר צבר הנוסע, והפלט שלו הוא מספר המרחקים (העוקבים) בתחילת הרשימה אשר סכומם עולה על 30,000.

למשל, עבור הקלט 1000 6000 8000 10000 7000 1500 5000 יהיה הפלט 5.

לפניך ישום הפתרון של בעיה 3.

```
program DistanceSum (input , output);
{
    קלט: רשימת מרחקי טיסה
    פלט: מספר המרחקים, מתחילת הרשימה, שסכומם עולה על 30,000
}
var
    dist : integer      {מרחק טיסה} ;
    Sum : integer ; {סכום מרחקים מצטבר} ;
```

```

Counter : integer ; (מונה מרחקים שהצטברו) ;
begin
  Sum := 0 ;
  Counter := 0 ;
  writeln ('הקש את רשימת המרחקים') ;
  while Sum <= 30000 do
    begin
      read( Dist) ;
      Sum := Sum + Dist ;
      Counter := Counter + 1
    end;
  writeln ( counter , ' : הוא 30,000 עולה על '
end;

```

1. טען את הקובץ distance.pas והדר את התוכנית.

2. הרץ את התוכנית עבור 5000 1500 7000 10000 8000 6000 1000
מהו הפלט שנתקבל?

ברצוננו לעקוב אחר ערכו של Sum במהלך ביצוע התוכנית. אפשרות אחת היא להריץ את התוכנית בשלבים, אולם אפשרות זו ארוכה ומייגעת. ניתן להריץ תוכנית ולעצרה בנקודות שונות במהלך ההרצה. לשם צפייה במצב ביצוע של התוכנית בנקודה מסוימת, יש להגדיר נקודה זו כנקודת עצירה:

3. הצב את הסמן על השורה while Sum <= 30000

4. בחר בתת-תפריט break/watch

בחר בהוראה Toggle Breakpoint

5. שים לב ששורה זו מודגשת על המסך.

הרץ את התוכנית עבור 5000 1500 7000 10000 8000 6000 1000

(כדי להריץ : בחר בתת-תפריט RUN

בחר בהוראה RUN

או

הקש את הקיצור המופיע ליד ההוראה (RUN).

התוכנית תיעצר לפני ביצוע השורה המודגשת.

6. היכן נעצרת התוכנית:

מדוע? כי הגדרנו משפט זה כנקודת עצירה

7. הכנס את המשתנה Sum לחלון הצפייה.

מה ערכו של Sum?

8. הכנס את המשתנה Counter לחלון הצפייה.

מה ערכו של Counter?

9. הכנס את הביטוי $Sum \leq 30000$ לחלון הצפייה.

מה ערכו של הביטוי?

10. המשך את הרצת התוכנית

(בחר : בחר בתת-תפריט RUN

בחר בהוראה RUN

או הקש את הקיצור שליד ההוראה (RUN)

התוכנית תרוץ עד שתתקל שוב בנקודת עצירה, או (אם אין בדרך נקודת עצירה)

עד סוף התוכנית.

שים לב שבתוכנית הזו מוגדרת רק נקודת עצירה אחת ולכן בצוע התוכנית יעצר בכל פעם שהמשפט הבא לביצוע הוא נקודת העצירה שסימנת.

11. השלם את ערכם של המשתנים כעת:

Counter: Sum:

12. המשך את הרצת התוכנית עד סופה (בעזרת RUN) וענה:

- מהו הפלט התוכנית?
- מהו ערכו של Sum עם תום ביצוע הלולאה?
- התבונן בחלון הצפיה על הביטוי $Sum \leq 30000$ מה ערכו של הביטוי?

13. מדוע נסתיים ביצוע הלולאה?

14. הצב את הסמן על נקודת העצירה.

בחר בתת-תפריט break/watch

בחר בהוראה Toggle Breakpoint

שים לב ששורה זו כבר אינה מודגשת על המסך. כלומר ביטלנו את נקודת העצירה. כעת אם נריץ את התוכנית לא נעצר באף שלב.

סיכום

1. הגדרת נקודת עצירה נועדה כדי לצפות בתוכנית בזמן ההרצה.

א. הגדרת נקודת עצירה מתבצעת באופן הבא:

1. הצב הסמן על השורה בה ברצונך שהתוכנית תיעצר.

2. בחר בתת-תפריט break/watch

בחר בהוראה Toggle Breakpoint

2. הקש את הקיצור המופיע ליד ההוראה Toggle Breakpoint.

ב. ביטול נקודת עצירה מתבצע באופן הבא:

1. הצב הסמן על נקודת העצירה שברצונך לבטל

2. בחר בתת - תפריט break/watch

בחר בהוראה Toggle Breakpoint

או

2. הקש את הקיצור המופיע ליד ההוראה Toggle Breakpoint.

3. שים לב שבחירת RUN תמשיך את ביצוע התוכנית עד לנקודת העצירה הבאה.

אם ברצונך להפסיק את הרצת התוכנית הדר אותה (או הקש ctrl F2 בו זמנית).

לביצוע הידור: בחר בתת - תפריט COMPILE

בחר בהוראה COMPILE

או

הקש את הקיצור המופיע ליד ההוראה COMPILE.

משימה 5

מטרה: תרגול לולאת while.

התבונן בשאלה 7.27 מספר הלימוד. לנוחותך העתקנו אותה שוב:

שני תלמידים המשחקים זה נגד זה מותחים בתחילת המשחק קו באורך N ס"מ ($N > 1$). השחקנים מחליפים תורים לסירוגין. כל שחקן מקצר את הקו לחצי מאורכו. השחקן אשר מקצר בתורו את הקו לאורך של פחות מ-1 ס"מ מנצח במשחק. למשל, אם אורכו התחילי של הקו הוא 8.0 ס"מ, אזי השחקן הראשון יקצר את הקו ל-4.0, השחקן השני יקצר את הקו ל-2.0 ס"מ, השחקן הראשון יקצר את הקו ל-1.0 ס"מ והשחקן השני יקצר את הקו ל-0.5 ס"מ וינצח במשחק. פתח וישם אלגוריתם אשר הקלט שלו הוא אורכו התחילי של הקו, והפלט שלו הוא השחקן המנצח במשחק (כלומר הפלט הוא ההודעה "הראשון מנצח" או ההודעה "השני מנצח").

1. הקלד את הישום (התוכנית) לפתרון בעיה זו ממחברתך.

2. הדר את תוכניתך עד שלא יהיו בה שגיאות תחביר.

3. אם צריך, נפה את השגיאות מתוכניתך בעזרת מנפה השגיאות: לרשותך כלים כמו

TRACE, WATCH, BREAKPOINT. הקפד על עימוד תוכניתך, כדי שתוכניתך

תהיה בהירה.

המנצח	אורך הקו
	16.0
	4.0
	10.9
	15

שאלת רשות

2. השלם: הראשון ינצח כאשר:

השני ינצח כאשר:

משימה 6

מטרה: העמקת ההבנה בלולאת while, התבוננות בלולאה אינסופית, הפסקת הרצה על-ידי המשתמש.

התבונן שוב בבעיה 5 שבפרק 7. לנוחותך העתקנו אותה לכאן.

בעיה 5

נתונה התוכנית הבאה:

```
program Odds (input , output);
var
  Limit      :   integer;
  OddNum     :   integer;
begin
  writeln('הקש מספר שלם חיובי ');
  readln (Limit) ;
  writeln('המספרים האי-זוגיים הקטנים מן המספר הנתון הם ');
  OddNum := 1 ;
  while OddNum < > Limit do
  begin
    write(OddNum);
    OddNum := OddNum + 2
  end
end.
```

1. א. טען את הקובץ .odd.pas

ב. הדר את התוכנית.

2. מהי מטרת התוכנית?

3. הרץ את התוכנית עבור הקלט 9.

מה פלט התוכנית?

4. הרץ את התוכנית עבור הקלט 8.

תאר את מה שמוצג על המסך

5. הפסק את הרצת התוכנית:

לחץ ביד אחת על מקש <ctrl> ומבלי לשחרר הקש ביד השנייה על המקש break
על המסך תופיע הודעה: user break, הקש esc.

6. מדוע לא נעצרה התוכנית מעצמה?

7. לולאה שביצועה אינו מסתיים נקראת **לולאה אין סופית**. שנה את תנאי הכניסה
ללולאה כך שיתאים גם למספרים זוגיים.

8. כעת, הדור והרץ את התוכנית עבור הקלט 8.

מהו פלט התוכנית?

סיכום לקחים

1. הפסקת הרצה על ידי המשתמש מתבצעת על-ידי הקשת ctrl והקשת break בו
זמנית.

משימה 7

מטרה: תרגול לולאה עם זקיף.

לפניך שאלה 7.40 מספר הלימוד. לנוחותך העתקנו אותה לכאן.

בבחירות לוועד ישנם שלושה מועמדים.

מועמד מספר 1, מועמד מספר 2, ומועמד מספר 3.

פתח וישם אלגוריתם, אשר הקלט שלו הוא רשימת קולות (כל קול הוא אחד

מהמספרים 1, 2 או 3) אשר בסופה המספר 0, והפלט שלו הוא אחוז הקולות בהם

זכה מועמד מספר 2.

1. הקלד את הישום (תוכנית) לפתרון שאלה 7.40.

2. הדך את תוכניתך עד שלא יהיו בה שגיאות תחביר.

3. הרץ את תוכניתך עבור קלטים שתבחר, על מנת לוודא נכונות. אם נתקלת בבעיה,

השתמש במנפה השגיאות, כדי לתקן את תוכניתך.

תשובות

התבונן בפתרון שבמדריך למורה של "יסודות I".

פרק 8

משימה 1

מטרה: הבהרת המושג יעילות.

התבונן בשאלה 8.6 שבספר הלימוד.

לנורותך העתקנו אותה שוב.

יש לפתח אלגוריתם אשר הקלט שלו הוא מספר שלם חיובי N , והפלט שלו הוא כל המספרים (השלמים והחיוביים) אשר קטנים מ- N והשורש שלהם הוא מספר שלם. למשל, עבור הקלט 50 יהיה הפלט: 1 4 9 16 25 36 49.

מטרתנו היא לפתח אלגוריתם יעיל:

1. האלגוריתם הבא, אשר כולל משתנים מטיפוס שלם הוא פיתרון אפשרי:

קלוט מספר שלם חיובי N

בצע עבור I מ-1 עד N

אם שורש I הוא מספר שלם אזי

הצג כפלט את ערכו של I

א. ישם את האלגוריתם הנתון לתוכנית פסקל.

ב. הרץ את התוכנית עבור $N = 50$.

וודא שאכן נתקבל הפלט הנכון.

ג. כעת הרחב את התוכנית כך שתמדוד את יעילותה:

1. הצהר על משתנה נוסף בשם counter, בו תמנה את מספר הפעמים

שהלולאה התבצעה.

2. אפס את counter (קבע את ערכו התחילי של counter כאפס),

בתחילת התוכנית.

3. בתחילת גוף הלולאה, הגדל את ערכו של counter ב-1.

4. הדפס את counter בסוף התוכנית.

ד. הרץ את התוכנית עבור $N = 50$.

מהו ערכו של counter?

זמן הביצוע (מספר הפעמים שהתבצעה הלולאה) עבור קלט זה הוא ערכו של counter, כיוון ש counter מנה את מספר הפעמים שהלולאה התבצעה.

ה. מהו זמן הביצוע עבור N כלשהו?

ו. שמור את תוכניתך, כיוון שתזדקק לה בהמשך המשימה.

שמירת קובץ:

בחר בתת-תפריט file

בחר בהוראה save

אם תתבקש, הקש את שם הכוון ואת שם הקובץ שברצונך לשמור.

2. לפניך אלגוריתם יעיל יותר מהאלגוריתם הנתון:

א. השלם את החסר:

קלוט מספר שלם חיובי N

בצע עבור I מ-1 עד $\text{trunc}(\sqrt{n})$

הצג כפלט עת ערכו של $I \cdot I$

ב. ישם את האלגוריתם הנייל לתוכנית פסקל.

ג. הרץ את התוכנית עבור $N = 50$.

וודא שאכן נתקבל הפלט הנכון.

ד. הוסף לתוכנית מונה בשם counter2 שימנה את מספר הפעמים שנתבצעה הלולאה.

(הוסף באופן דומה לביצוע סעיף ג באלגוריתם הקודם).

ה. הרץ את התוכנית עבור הקלט $N = 450$.

מהו ערכו של counter2?

מהו זמן הביצוע (מספר הפעמים שהתבצעה הלולאה) עבור קלט זה?

מה יהיה זמן הביצוע (מספר הפעמים שהתבצעה הלולאה) עבור קלט $N = 800$?

ז. מהו זמן הביצוע עבור N כלשהו?

ח. כעת נמדוד בעזרת שעון או סטופר את משך הביצוע של כל אחד מן האלגוריתמים. הקלט עבור שניהם הוא 20,000. התחל למדוד זמן מרגע הקשת enter אחרי הקלדת הקלט.

כמה זמן (בשניות) נדרש לתוכנית השניה כדי להתבצע? _____

טען את תוכניתך הראשונה הדר והרץ.

כמה זמן (בשניות) נדרש לתוכניתך הראשונה כדי להתבצע _____

מסקנות

1. האם התוכניות שקולות ?

(תוכניות שקולות הן תוכניות שעבור אותו קלט מיצרות פלט זהה).

2. האלגוריתם השני יעיל יותר כי א. הוא שקול לראשון

פרק 9

משימה 1

מטרה: מעקב אחר מהלך זימון וביצוע של פונקציה בתוכנית, תחום קיום של משתנים מקומיים לעומת משתני החלק הראשי, פרמטרים. ההבדלים בין F7 ל F8.

התבונן בבעיה 1 שבפרק 9. לנוחותך העתקנו אותה לכאן שוב.

בעיה 1

קבוצת אורחים מגיעה למלון לבילוי סופשבוע. האורחים מגיעים בשתי טיסות - טיסה ביום חמישי בערב וטיסה ביום שישי בבוקר, ויש להביאם במוניות משדה התעופה למלון. פתח וישם אלגוריתם אשר הקלט שלו הוא מספר האורחים המגיעים ביום חמישי ומספר האורחים המגיעים ביום שישי בבוקר, והפלט שלו הוא מספר המוניות שיש לשלוח לשדה התעופה ביום שישי כדי להביא את האורחים. במונית ישנם 7 מושבים לנוסעים מלבד הנהג.

לפניך ישום הפתרון לבעיה 1

```
program Guesttaxis (input,output);
```

```
var
```

```
Thguest, FrGuest : integer ;
```

```
ThTaxis, FrTaxis : integer ;
```

```
function TaxisOrder (Passengers : integer) : integer ;
```

{טענת כניסה: Passengers מקבל מספר שלם חיובי המבטא מספר נוסעים}

{טענת יציאה: הפונקציה מחזירה את מספר המוניות הדרושות להסעת הנוסעים}

var

NumOfTaxis : integer ;

LeftOut : integer ;

begin

NumOfTaxis := Passengers **div** 7 ;

LeftOut := Passengers **mod** 7 ;

if LeftOut > 0 **then**

NumOfTaxis := NumOfTaxis + 1 ;

TaxisOrder := NumOfTaxis ;

end;

begin (החלק הראשי)

writeln ('הקש את מספר האורחים המגיעים ביום חמישי') ;

readln (ThGuest) ;

writeln ('הקש את מספר האורחים המגיעים ביום שישי') ;

readln (FrGuest) ;

ThTaxis := TaxisOrder (ThGuest) ;

FrTaxis := TaxisOrder (FrGuest) ;

writeln ('יש להזמין ליום ,ThTaxis, 'מוניות, וליום שישי, FrTaxis, 'מוניות')
(חמישי)

end.

1. טען את הקובץ TAXISOR.PAS.

2. הכנס לחלון ה-WATCH את המשתנים הבאים:

Passengers, NumOfTaxis, ThGuest, ThTaxis

להכנסת משתנה לחלון ה-WATCH:

בחר בתת תפריט WATCH/BREAK

בחר בהוראה AddWatch

הקש את שם המשתנה הרצוי
או הקש את הקיצור המופיע ליד ההוראה AddWatch
הקש את שם המשתנה הרצוי

3. הרץ את התוכנית שורה אחר שורה עד שהמשפט:
`ThTaxi := TaxiOrder (ThGuests)` יהיה מודגש.
הנח כי ביום חמישי מגיעים 60 אנשים וביום שישי מגיעים 70 אנשים.
הרצת שורה אחת:

בחר בתת תפריט RUN
בחר בהוראה TraceInto
או הקש את הקיצור המופיע ליד ההוראה TraceInto

4. העתק את ערכי המשתנים הבאים מחלון ה-WATCH.

ThGuest :

ThTaxi :

5. מהן ההודעות המופיעות ליד המשתנים הבאים:

Passengers :

NumOfTaxi :

הסיבה לכך שכיוון שכרגע מתבצע החלק הראשי של התוכנית, אין המחשב מכיר
את המשתנים המקומיים של הפונקציה ואת הפרמטרים שלה.

6. נמשיך את ההרצה בשלבים: הרץ את המשפט המסומן.
הרצת שורה אחת:

בחר בתת תפריט RUN
בחר בהוראה TraceInto
או הקש את הקיצור המופיע ליד ההוראה TraceInto

מהו המשפט המסומן עכשיו?

אם כן, מהו המשפט הבא לביצוע?

התבונן בחלון ה-watch. מה מופיע כעת ליד המשתנה Passengers?

Passengers :

האם זהו הערך לו ציפית ?

ערכו של איזה משתנה הועבר ל Passengers?

התבונן בחלון ה-watch. מה מופיע כעת ליד המשתנה NumOfTaxis?

NumOfTaxis :

מדוע נעלמה ההודעה unknown identifier ?

7. המשך את ההרצה שורה אחר שורה. עצור כאשר השורה המודגשת

היא end; (המהווה את סוף הפוקציה) וענה:

מה ערכם של המשתנים הבאים:

Passengers :

NumOfTaxis :

8. הרץ שורה זו בלבד וענה:

מהו המשפט המסומן עכשיו?

מדוע?

אם כן, מהו המשפט הבא לביצוע?

התבונן בחלון ה-watch.

מה מופיע ליד המשתנים הבאים?

Passengers :

NumOfTaxis :

משימה 2

משימה זו מבוססת על שאלה 9.19 מספר הלימוד.

1. יש לפתח פונקציה בשם CountDups אשר מקבלת כפרמטרים שלוש אותיות ומחזירה כפלט את מספר האותיות אשר שוות לאות המתקבלת ראשונה. למשל עבור הזימון CountDups('a', 'f', 'a') תחזיר הפונקציה את הערך 2.

2. כתוב תוכנית המשתמשת בפונקציה זו אשר הקלט שלה הוא מחרוזת תווים באורך 6 והפלט שלה הוא מספר הפעמים שהתו 'd' מופיע במחרוזת זו.

3. הרץ עבור הקלט andtad

וודא שקיבלת את הפלט הנכון.

זכור, אם נתקלת בקשיים במהלך עבודתך, השתמש במנפה השגיאות כדי לתקן את עצמך (watch, traceinto, stepover).

משימה 3

1. הרץ את שאלה 9.22 מספר הלימוד, לנוחותך העתקנו אותה שוב.
גירסה חדשה של המשחק בול-קליעה, אשר פותחה על-ידי מחבר ספר הלימוד.

המחשב בוחר מספר שלם בין 0 ל-99, השחקן שמשחק עם המחשב מנחש מספר שוב ושוב עד אשר הניחוש הוא המספר שבחר המחשב. עבור כל ניחוש שמשחק השחקן, עונה לו המחשב במספר שהוא "מרחק בספרות" בין הניחוש לבין בחירת המחשב.

"מרחק בספרות" בין שני מספרים הוא הסכום של המרחק בספרות היחידות והמרחק בספרות העשרות. למשל, עבור בחירת המחשב 23 והניחוש 81 יהיה "מרחק בספרות" 8, כיוון שהמרחק בספרות העשרות הוא $\text{abs}(2 - 8) = 6$ והמרחק בספרות היחידות הוא $\text{abs}(3 - 1) = 2$. המשחק יסתיים כאשר תשובת המחשב לניחוש היא 0.

2. שחק מול המחשב ונסה לגלות את בחירת המחשב תוך מספר נמוך ככל הניתן של ניחושים. כמה ניחושים נדרשים, לכל היותר, לגילוי בחירת המחשב?

פרק 10

משימה 1

מטרה: מעקב אחר משתנה מטיפוס מערך במהלך ביצוע תוכנית.

התבונן שוב בבעיה 1, שבספר הלימוד. לנוחותך העתקנו אותה לכאן.

בעיה 1

פתח וישם אלגוריתם שהקלט שלו הוא רשימת גבהי תלמידים בכיתה בת 40 תלמידים והפלט שלו הוא רשימת הגבהים שמעל הממוצע.

לפניך ישום הפתרון של בעיה 1.

שים לב, שלצורך התרגול במעבדה צומצם מספר התלמידים בכיתה לחמישה.

```
program HeightsAboveAvg (input,output);
```

```
{קלט: רשימת גבהים של 5 תלמידי הכיתה}
```

```
{פלט: רשימת הגבהים שמעל הממוצע}
```

```
var
```

```
height : array[1 .. 5] of real ;
```

```
Avg, Sum : real ;
```

```
I : integer ;
```

```
begin
```

```
Sum := 0 ;
```

```
writeln ('הקש את רשימת הגבהים של 5 תלמידי הכיתה:');
```

```
for I := 1 to 5 do
```

```
  begin
```

```
    read (Height[I]);
```

```
    Sum := Sum + Height[I];
```

```

end;
Avg := Sum/5 ;
for I := 1 to 5 do
  if Height[I] > Avg then
    write ( Height[I]:6:2 ) ;
end.

```

1. טען את הקובץ HEIGHT.PAS והדר את התוכנית.

2. הכנס לחלון ה-WATCH את המשתנים: Sum, Height[I], I, Height[2].

3. הרץ את התוכנית בשלבים עבור הקלט: 1.80 1.56 1.92 1.60 1.76.

שים לב שערכו של height[I] משתנה בכל ביצוע של הלולאה, ערכו של Height[2] משתנה רק עם קליטת נתון לתוכו ונשאר קבוע. מדוע?

עצור את ההרצה לאחר שכל הנתונים נקלטו. (לאחר סיום בצוע הלולאה הראשונה) וענה:

א. מהו ערכו של Height?

ב. כיצד נדע מהו ערכו של האיבר הרביעי ב-Height מחלון ה-WATCH?

4. המשך את הרצת התוכנית בשלבים וענה תוך כדי כך על השאלות הבאות:

א. מהו ממוצע גובהי התלמידים? (רמז: הכנס משתנה אל חלון ה-watch)

ב. כאשר $I = 3$, מה ערכו של Height[I]?

5. מהו פלט התוכנית?

6. שנה את משפט לולאת ה for ל- `for I := 1 to 7 do` הדר והרץ כאשר שני הנתונים הם 1.40 1.63.

המהדר מודיע על שגיאה. העתק את הודעתו.

הסיבה לכך היא שניסינו לפנות לאיבר השישי והשביעי במערך, בעוד במערך קיימים חמישה איברים בלבד.

החזר את משפט ה-for המקורי לתוכנית: `for I := 1 to 5 do`.

7. אם היינו משנים את משפט ה for ל- `for I := 1 to 4 do`, האם היה המהדר מודיע על שגיאה? הרץ ובדוק.

סיכום לקחים

1. במהלך ביצוע התוכנית, אם הכנסת משתנה מטיפוס מערך לחלון ה-watch, תוכל לצפות בכל איברי המערך, מהראשון לאחרון, לפי הסדר משמאל לימין. למשל Height.

2. אם ברצונך לצפות רק באיבר אחד מהמערך, הכנס לחלון ה-watch את שם המערך עם מציין האיבר, למשל Height[I] או Height[2].

3. לא ניתן לגלוש במערך מעבר לתחום הגדרתו, כלומר לא ניתן לפנות לאיבר במערך מחוץ לתחום המוגדר.

משימה 2

הרץ את שאלה 10.20 שבספר הלימוד, לנוחותך, העתקנו אותה לכאן:

1. פתח וישם אלגוריתם אשר הקלט שלו הוא מחרוזת של 20 תווים והפלט שלו הוא

מספר הזוגות של תווים עוקבים במחרוזת אשר שוים לזוג האחרון.

למשל עבור הקלט ABCABACADCBABDDAAB

יהיה הפלט 5 כיוון שישנם 5 זוגות תווים שהם AB.

שים לב

עבור מחרוזת קלט של 20 תווים זהים יהיה הפלט 19 כיוון שהתווים הראשון-

השני, שוים לשני האחרונים, כך גם השני-שלישי, השלישי-רביעי וכו'.

2. הרץ את תוכניתך עבור קלטים שונים כדי לוודא את נכונותה.

אם נתקלת בקשיים במהלך בדיקת תוכניתך, השתמש במנפה השגיאות

(trace, watch, breakpoint וכו') כדי לתקנה.

מלא את הטבלה בהתאם לקלטים שבחרת:

קלט	פלט
AAAAABBBBBCCCCDDDBC	2

3. כמה דוגמאות קלט תצטרך, על מנת לבדוק את כל המקרים?

משימה 3

במשימה זו נחקור את הפונקציה random.

הפונקציה random מקבלת כפרמטר מספר חיובי שלם ומגרילה מספר חיובי שלם הקטן מהפרמטר שלה. לכל מספר בתחום יש סיכוי שווה להיות מוגרל. מטרת האלגוריתם הבא היא לבדוק האם טענה זו נכונה. אנו נבדקה עבור הגרלה של 2 מספרים. לשם כך נגריל הרבה מספרים בתחום בין 0 או 1, ונבדוק האם מספר הפעמים שהוגרל 1 קרוב למספר הפעמים שהוגרל 0.

1. האלגוריתם הבא מגריל 30,000 מספרים שלמים בין 0 ל-1 לתוך מערך, ולאחר מכן מחשב את מספר הפעמים שהוגרל המספר 1 ומספר הפעמים שהוגרל המספר 0.

בצע עבור I, מ-1 עד 30,000.

חשב random(2) והשם אותו ב A[I]

אתחל את Counterof1 ל-0

בצע עבור I, $A[I] = 1$ אזי

הגדל ב-1 את Counterof1

הצג כפלט Counterof1.

2. ישם אלגוריתם זה על-ידי תוכנית בשפת פסקל. הדר והרץ.

מהו הפלט שנתקבל?

3. לטורבו פסקל קיימת פונקציה memavail הבודקת בזמן ריצה של תוכנית כמה

מקום (ב-BYTES) נשאר פנוי בזיכרון.

הוסף בסוף תוכניתך את השורה writeln (memavail)

הדר והרץ. מהו הפלט שנתקבל?

Counterof1 :

memavail :

4. הקטן את המערך מ 30,000 איברים ל 15,000 איברים. שנה את תוכניתך בהתאם.

שים לב שגבולות הלולאה חייבים להשתנות, אחרת תהיה גלישה מן המערך.

5. הדר והרץ את תוכניתך.

מהו הפלט שנתקבל?

Counterof1 :

memavail :

6. האם התפנה מקום בזיכרון?

מדוע?

7. התבונן באלגוריתם הבא:

אתחל את Counterof1 ל- 0

בצע עבור I, מ 1 עד 30,000

חשב random(2) והשם Num

אם $Num = 1$ אזי

הגדל ב-1 את Counterof1

הצג כפלט את Counterof1.

9. אלגוריתם זה משיג את אותה מטרה, ללא שימוש במערך.

ישם אותו על ידי תוכנית פסקל. הוסף את המשפט `writeln (memavail)` בסוף

התוכנית

הדר והרץ.

10. מהו הפלט שנתקבל?

Counterof1 :

memavail :

11. האם התפנה מקום בזיכרון?

מדוע?

12. האם אפשר לאמר שפונקציה random מגרילה את ההמספר 1 בהסתברות שווה

למספר 0? מדוע?

סיכום לקחים

זיכרון המחשב מוגבל. ככל שהמערך גדול יותר, הוא תופס יותר מקום בזיכרון.

נספח פסקל

שאלה 1:

קבע עבור כל אחת ממחרוזות התווים הבאות, האם היא מזהה בפסקל. אם אינו מזהה, נמק מדוע.

א. total	ב. Total	ג. 25	ד. abcdef
ה. A+B	ו. V2	ז. x11k3m8	ח. 3total
ט. student grade	י. q15	יא. c	יב. meter/sec/sec

הופעת פסיק, נקודה או כל תו אחר שאינו ספרה, הסימן + או הסימן -, בערך מטיפוס שלם אינה חוקית בפסקל.

דוגמאות

- ★ 1,500 אינו ערך חוקי מטיפוס שלם בפסקל, משום שמופיע בו פסיק.
- ★ 5.8 אינו ערך חוקי מטיפוס שלם בפסקל, משום שמופיעה בו נקודה.

שים לב! ♥ כי למרות שבסימון המתמטי המקובל, פסיק יכול להופיע במספר שלם, פסיק אינו יכול להופיע בערך חוקי מטיפוס שלם בפסקל.

שאלה 2:

קבע עבור כל אחד מהערכים הבאים, האם הוא ערך חוקי מטיפוס שלם בפסקל. אם אינו חוקי, הסבר מדוע.

א. +0	ב. -5	ג. +5.3	ד. -5.0
ה. +9.0	ו. -0	ז. +123456	ח. 00
ט. 18.0	י. -00	יא. +12345	יב. 1,125

כל נסיון לכתוב ערך ממשי עם פסיק, או ללא ציון שבר אינו חוקי בפסקל.

דוגמאות

- ★ 1,500.7 אינו ערך חוקי מטיפוס ממשי בפסקל משום שמופיע בו פסיק.
- ★ 6 אינו ערך חוקי מטיפוס ממשי בפסקל משום שחסרה בו נקודה עשרונית וסדרת ספרות המייצגת את השבר של המספר.
- ★ 36 אינו ערך חוקי מטיפוס ממשי בפסקל משום שחסרות בו ספרות (אחת לפחות) אחרי הנקודה העשרונית.

שאלה 3:

קבע עבור כל אחד מהערכים הבאים, האם הוא ערך חוקי מטיפוס ממשי בפסקל. אם אינו חוקי, נמק מדוע.

א. +0	ב. -1.9	ג. +5.3	ד. -5.0
ה. 9,000.45	ו. 456.	ז. -12.34	ח. 0.

שאלה 4:

קבע עבור כל אחד מהערכים הבאים, האם הוא ערך חוקי בפסקל. אם הוא חוקי, קבע את טיפוסו (שלם או ממשי). אם אינו חוקי, נמק מדוע.

א. 314,570	ב. 103.3	ג. +9.0	ד. -400
ה. 002	ו. 10000.300	ז. .456	ח. 8/2
ט. 18.	י. -88	יא. 456.	יב. +17,500
יג. 6/4.	יד. +72.0	טו. '2'	טז. 000

שאלה 5:

יצג כל אחד מן המספרים הבאים כערך מטיפוס ממשי בפסקל:

א. 1,500	ב. 1,500.5	ג. 1500
----------	------------	---------

שאלה 6:

קבע עבור כל אחת מהצהרות המשתנים הבאות, האם היא חוקית בפסקל. אם אינה חוקית, נמק מדוע.

- א. `var Count, Amount: integer;`
- ב. `var Count: integer;`
- ג. `var Amount: integer;`
- ד. `var Count, Amount: integer;`
- ה. `var Count: integer`
- ו. `Amount: integer;`

שאלה 7:

כתוב הצהרת משתנים אשר תפקידיהם לייצג ערכי יום, חודש, שנה, והמספר הממוצע של ימים בחודש במהלך השנה. הקפד על מתן שמות משמעותיים למשתנים.

שאלה 8:

השלם את הצהרת המשתנים הבאה:

`var` _____ (שומר תוצאת קפיצה לרוחק)
 _____ (שומר מספר סידורי של קופץ)

שאלה 9:

כתוב הצהרת משתנים אשר תפקידיהם לייצג ערכי גיל, גובה ומשקל של בן-אדם.

שאלה 10:

האם הצהרות המשתנים הבאות שקולות?

`var Day, Month: integer;` `var Month: integer;`
`Avg: real;` `Avg: real;`
 `Day: integer;`

שאלה 11:

נתונה הצהרת המשתנים

```
var Temp1, Temp2, Temp3: integer;  
    Temp4, Temp5: real;
```

כתוב משפטי השמה לביצוע הפעולות הבאות.

- איפוס המשתנה Temp5.
- הגדלת ערך המשתנה Temp1 ב-1.
- השמת ערך המשתנה Temp2 במשתנה Temp3.
- הוספת ערך המשתנה Temp4 למשתנה Temp5.

שאלה 12:

הנח כי בתכנית מופיעה הצהרת המשתנים

```
var Total: integer;  
    Avg: real;
```

לפניך רשימת מספרים חוקיים בפסקל.

20.300	05	+45	13.3
-7.4	5	-88	45

התאם לכל אחד מהמשתנים את המספרים אשר אפשר לשים בו.

שאלה 13:

הפוך כל אחד מזוגות משפטי ההשמה הבאים למשפט השמה יחיד.

- $X := 2 * Y;$
- $X := X - 5$
- $A := X + Y;$
- $A := B - 3$

ידוע לנו כי המבנה הכללי של משפט השמה הוא **ביטוי** := **משתנה**, כאשר טיפוס המשתנה וטיפוס ערך הביטוי זהים.

ביטוי חשבוני (arithmetic expression) בפסקל הוא ערך מספרי, משתנה או צירוף של ערכים מספריים, משתנים ופעולות חשבון. ביטויים חשבוניים מוכרים לנו היטב ממתמטיקה.

דוגמאות

★ 5, x, 3x+5 הם ביטויים חשבוניים מוכרים ממתמטיקה.

כל ביטוי בפסקל, בפרט ביטוי חשבוני, מאופיין על-ידי טיפוס הביטוי. אם ביטוי חשבוני הוא ערך מספרי, אזי טיפוסו מוגדר היטב על-פי דרך הצגת הערך. אם ביטוי חשבוני הוא משתנה, אזי טיפוסו מוגדר היטב על-פי הצהרת המשתנה. אם ביטוי חשבוני מורכב מצירוף ערכים מספריים, משתנים ופעולות חשבון, אזי טיפוס הביטוי הוא כטיפוס הערכים המספריים והמשתנים המופיעים בו.

דוגמאות

נניח כי המשתנים A, B הם מטיפוס שלם, והמשתנים C, D הם מטיפוס ממשי.

★ טיפוס הביטוי	A+B	הוא שלם.
★ טיפוס הביטוי	C*D	הוא ממשי.
★ טיפוס הביטוי	2-7	הוא שלם.
★ טיפוס הביטוי	3.7+2.3	הוא ממשי.
★ טיפוס הביטוי	3.5-D	הוא ממשי.
★ טיפוס הביטוי	B*12	הוא שלם.

שים ♥ כי פעולת כפל מסומלת בפסקל רק באמצעות *, ולא כפי שמקובל לפעמים במתמטיקה: באמצעות נקודה או ללא אף סמל.

דוגמאות

★ הביטוי המתמטי	3*X	יכתב בפסקל כך: 3*X
★ הביטוי המתמטי	3X	יכתב בפסקל כך: 3*X
★ הביטוי המתמטי	3.5*X	יכתב בפסקל כך: 3.5*X

בפסקל מותרים ביטויים חשבוניים המורכבים מביטויים מטיפוס ממשי ומביטויים מטיפוס שלם. טיפוסו של ביטוי המורכב מחיבור, חיסור או כפל של ביטויים חשבוניים מטיפוסים שונים הוא ממשי.

דוגמאות

נניח כי המשתנה A הוא מטיפוס שלם והמשתנה B הוא מטיפוס ממשי.

★ טיפוס הביטוי	A+B	הוא ממשי.
★ טיפוס הביטוי	A*B	הוא ממשי.
★ טיפוס הביטוי	2-B	הוא ממשי.
★ טיפוס הביטוי	3.7+A	הוא ממשי.
★ טיפוס הביטוי	2-5.3	הוא ממשי.
★ טיפוס הביטוי	3.7*4	הוא ממשי.

סמל הפעולה בפסקל של חילוק ממשי הוא /. תוצאת הפעולה היא כתוצאת פעולת החילוק / במתמטיקה.

דוגמאות

★	3.5/2.5=1.4
★	-3.5/2.5=-1.4

פעולת חילוק ממשי בפסקל יכולה להתבצע בין שני ביטויים מטיפוס ממשי, בין שני ביטויים מטיפוס שלם או בין שני ביטויים שאחד ממשי ואחד שלם. ביטוי המורכב מפעולת חילוק ממשי הוא מטיפוס ממשי.

דוגמאות

נניח כי המשתנים A, B הם מטיפוס שלם, והמשתנים C, D הם מטיפוס ממשי.

★ טיפוס הביטוי	C/D	הוא ממשי.
★ טיפוס הביטוי	A/B	הוא ממשי.
★ טיפוס הביטוי	2/7	הוא ממשי.
★ טיפוס הביטוי	3.7/2	הוא ממשי.
★ טיפוס הביטוי	3.5/D	הוא ממשי.

★ טיפוס הביטוי $2/A$ הוא ממשי.
 ★ טיפוס הביטוי B/C הוא ממשי.
 סמלי הפעולה בפסקל של חילוק שלם הם **div** ו-**mod**. תוצאת הפעולה **div** היא מנת החילוק, ותוצאת הפעולה **mod** היא שארית החילוק.

דוגמאות

$$\begin{array}{rcl} 7 \text{ div } 3 & = & 2 \text{ H} \\ 7 \text{ mod } 3 & = & 1 \\ 2 \text{ div } 3 & = & 0 \text{ H} \\ 2 \text{ mod } 3 & = & 2 \end{array}$$

חילוק שלם בפסקל יכול להתבצע רק בין שני ביטויים מטיפוס שלם. ביטוי המורכב מפעולת חילוק שלם בין שני ביטויים מטיפוס שלם הוא מטיפוס שלם.

דוגמאות

נניח כי המשתנים A ו-B הם מטיפוס שלם.

★ טיפוס הביטוי $A \text{ div } B$ הוא שלם.
 ★ טיפוס הביטוי $A \text{ mod } B$ הוא שלם.
 ★ טיפוס הביטוי $A \text{ div } 25$ הוא שלם.
 ★ טיפוס הביטוי $7 \text{ mod } B$ הוא שלם.
 ★ טיפוס הביטוי $10 \text{ div } 5$ הוא שלם.
 ★ טיפוס הביטוי $5 \text{ mod } 4$ הוא שלם.

שאלה 14:

הנח כי בתכנית מופיעה הצהרת המשתנים הבאה:

```
var Total: integer;  
    Avg: real;
```

קבע עבור כל אחד מהביטויים החשבוניים הבאים, האם הוא חוקי בפסקל. אם הוא חוקי, קבע את טיפוסו.

Avg*Total	ב.	2*Total	א.
5*Avg+Total	ד.	3-1.2	ג.
4/2+5*7	ו.	136 mod Total	ה.
2*Total/Total	ח.	Total-5*Avg	ז.
3 mod Total	י.	Avg div 3	ט.
4 div 2.0	יב.	Avg-1	יא.

שאלה 15:

חשב את ערכו של כל ביטוי חוקי המופיע בשאלה 14. הנח כי ערכו של Total הוא 2, וערכו של Avg הוא 1.2.

div ו-**mod** הן מלים שמורות בפסקל. כאשר מלים אלה מופיעות בתכנית פסקל הן צריכות להיות מוקפות ברווחים.

הלך הולב... מדוע כשמופיעות המלים **div** ו-**mod** בתכנית פסקל הן מוקפות ברווחים?

יש להקיף את המלים **div** ו-**mod** ברווחים כדי שהמהדר יצליח לזהותן בתוך ביטוי חשבוני כלשהו. המהדר לא מבחין כמובין בכך שהמלים האלה מודגשות. הדגשת המלים השמורות בתכנית נועדה לקורא אותה, ולא למהדר.

דוגמה

★ **mod 9** a הוא שם משתנה. הביטוי **mod 9** a אינו שם משתנה, כי מופיעים בו רווחים, ולכן הכוונה היא לשארית החילוק של המשתנה a במספר 9.

בחומרי הלמידה שלנו, יופעלו האופרטורים **div** ו-**mod** רק על ערכים אי-שליליים.

שים ♥ כי בפסקל אין סמל פעולה להעלאה בחזקה כלשהי. אנו נשתמש בפעולת כפל לביטוי העלאה בחזקה. למשל: לחישוב X^3 נכתוב $X * X * X$.

לא תמיד אפשר לייצג בפסקל ביטויים חשבוניים בדרך המקובלת במתמטיקה. טבלה 3.1 מדגימה הצגת כמה ביטויים חשבוניים הנכתבים בצורה שונה במתמטיקה ובפסקל.

ביטוי חשבוני במתמטיקה	ביטוי חשבוני בפסקל
$2 \cdot 3$	$2 * 3$
$B^2 - 4AC$	$B * B - 4 * A * C$
$X \cdot Y + 2$	$X * Y + 2$
$\frac{1}{X}$	$1 / X$
$A + \frac{B}{C}$	$A + B / C$

טבלה 3.1 - ביטויים חשבוניים במתמטיקה ובפסקל

שאלה 16:

המר את הביטויים החשבוניים הבאים, הכתובים בדרך המקובלת במתמטיקה, לביטויים חשבוניים בפסקל.

- $3X$
- $3X + Y$
- $X + Y$
- $Y^2 + 5$
- $\frac{X}{Y}$

שאלה 17:

הנח כי בתכנית מופיעה הצהרת המשתנים

```
var Number: integer;
```

וערכו של המשתנה Number הוא 2. חשב את ערכו של כל אחד מהביטויים החשבוניים הבאים.

24 mod 4	ב.	24 div 4	א.
36 mod 5	ד.	36 div 5	ג.
6 mod 9	ו.	6 div 9	ה.
Number mod Number	ח.	(Number+1) div 2	ז.
3 mod Number	י.	25 div Number	ט.

שאלה 18:

נתונה הצהרת המשתנים

```
var A, B, C, D: integer;
```

כתוב מהם ערכי המשתנים A, B, C ו-D לאחר ביצוע קטע התכנית הבא.

```
A := 27;
B := 343;
D := (A+B) div 10;
A := D mod 10;
B := B+A;
C := B div A;
C := (B+C-30) mod D
```

שאלה 19:

הנח כי בתכנית מופיעה הצהרת המשתנים הבאה:

```
var Total: integer;
    Sum: real;
```

קבע עבור כל אחד מהביטויים החשבוניים הבאים, האם הוא חוקי בפסקל. אם הוא חוקי, קבע את טיפוסו.

Sum div Total	ב.	Total/Total	א.
Total-3	ד.	Total-3.0	ג.
7*Sum	ו.	Sum+ Total	ה.
Total mod Total	ח.	Sum/Total	ז.
3 mod Total	י.	Total div 4+Sum	ט.
Total div 2.0	יב.	(5+Total)*4.0	יא.

שאלה 20:

המר את הביטויים החשבוניים הבאים, הכתובים בדרך המקובלת במתמטיקה, לביטויים חשבוניים בפסקל.

א. YX

ב. $X-Y$

$\frac{X+2}{Y}$

ג. 5

ד. $5(X+Y)$

ה. $\frac{X}{Y}+6$

שאלה 21:

הפוך כל אחד מזוגות משפטי ההשמה הבאים למשפט השמה יחיד.

א. $X := Y+2;$

$X := X*2;$

ב. $X := 2*X;$

$X := 5+X \text{ div } 2;$

ג. $A := 4*A-2;$

$A := 5+A \text{ div } 2;$

נרחיב עתה בהצגת השימוש במשפטי הקלט read ו- readln .
נתייחס אל לוח המקשים כאל אמצעי הקלט.

דוגמאות

נניח כי בתכנית מופיעה הצהרת המשתנים

var A, C: integer;

B: real;

$\text{read}(A)$ ★

ביצוע המשפט גורם למחשב להפסיק את ביצוע התכנית עד לקבלת נתון אחד מלוח המקשים. בדרך-כלל, מודיע המחשב שהוא מחכה לנתון על-ידי היבהוב סמן. הנתון שיש להקיש בדוגמה הזאת הוא מספר מטיפוס שלם, כיוון שטיפוס המשתנה A הוא שלם. — אם הקלט הוא

123

אזי הערך 123 יושם במשתנה A.

$\text{read}(A, B)$ ★

ביצוע המשפט גורם למחשב להפסיק את ביצוע התכנית עד לקבלת שני נתונים מלוח המקשים. שני הנתונים שיש להקיש בדוגמה הזאת הם מספר מטיפוס שלם, כיוון שהמשתנה A הוא מטיפוס שלם, ומספר מטיפוס ממשי, כיוון שהמשתנה B הוא מטיפוס ממשי.
 – אם הקלט הוא

123 4.5
 אזי הערך 123 יושם במשתנה A, והערך 4.5 יושם במשתנה B.
 תוצאה דומה תתקבל אם הקלט הוא
 123 4.5

או אם הקלט מופיע בשתי שורות
 123
 4.5
 (כלומר, אחרי הקשת הנתון 123 מקיש המשתמש <Enter> לסיום שורה)

★ read(A, B, C)
 ביצוע המשפט גורם למחשב להפסיק את ביצוע התכנית עד לקבלת שלושה נתונים מלוח המקשים. הנתון הראשון שיש להקיש בדוגמה הזאת הוא מספר מטיפוס שלם, הנתון השני שיש להקיש הוא מספר מטיפוס ממשי, והנתון השלישי הוא מספר מטיפוס שלם.

– אם הקלט הוא
 123 4.5 -89
 אזי הערך 123 יושם במשתנה A, הערך 4.5 יושם במשתנה B, והערך -89 יושם במשתנה C.

תוצאה זהה תתקבל אם הקלט הוא
 123 4.5
 -89

או
 123
 4.5
 -89

★ read(B, A)
 – אם הקלט הוא
 -0.5 +8
 אזי הערך -0.5 יושם במשתנה B, והערך 8 יושם במשתנה A.
 – אם הקלט הוא

2 8
 אזי הערך 2.0 יושם במשתנה B, והערך 8 יושם במשתנה A (המחשב יבצע המרה של המספר מטיפוס שלם 2 למספר מטיפוס ממשי 2.0).

★ read(A);
 read(B, C)
 ביצוע שני המשפטים גורם למחשב להפסיק את ביצוע התכנית עד לקבלת נתון מטיפוס שלם. עם קבלת הנתון הוא מושם במשתנה A, והביצוע מופסק עד לקבלת שני

נתונים נוספים מלוח המקשים. תוצאת ביצוע שני המשפטים שקולה לתוצאת ביצוע המשפט

```
read(A, B, C)
```

ולתוצאת ביצוע שני המשפטים

```
read(A, B);
```

```
read(C);
```

ולתוצאת ביצוע שלושת המשפטים

```
read(A);
```

```
read(B);
```

```
read(C);
```

```
readln(A);
```

★

```
read(B, C);
```

— אם הקלט הוא

```
123 4.5
```

```
4 -6
```

אזי הערך 123 יושם במשתנה A, הערך 4.0 יושם במשתנה B, והערך -6 יושם במשתנה C.

— אם הקלט הוא

```
123
```

```
4.5 4
```

```
-6
```

אזי הערך 123 יושם במשתנה A, הערך 4.5 יושם במשתנה B, והערך 4 יושם במשתנה C.

תוצאה זהה תתקבל אם הקלט הוא

```
123
```

```
4.5
```

```
4
```

```
-6
```

כללים המנחים אותנו בניהול קלט הם:

★ סדר קליטת הנתונים למחשב הוא משמאל לימין. אם אין מספיק נתונים בשורת הקלט, עובר המחשב לקליטת נתונים נוספים בשורת הקלט הבאה.

★ כאשר הקלט הוא ערכים מספריים צריך רווח אחד לפחות להופיע בין נתוני הקלט, כדי לדעת היכן מתחיל והיכן מסתיים כל נתון.

★ מספר הנתונים שבקלט צריך להיות לפחות כמספר המשתנים שברשימת המשתנים.

★ סדר השמת הנתונים במשתנים שברשימת המשתנים הוא משמאל לימין.

★ טיפוס הנתון וטיפוס המשתנה שבו הוא מושם, על-פי הוראת קלט, צריכים להיות זהים. לכלל זה יש יוצא מן הכלל: ניתן להשים נתון מטיפוס שלם במשתנה מטיפוס ממשי. המחשב יבצע את המרת הנתון לערך מטיפוס ממשי בעת קליטתו.

סדר הצגת פריטי הפלט על צג הוא משמאל לימין, לפי הופעתם ברשימת הפלט של משפט write או של משפט writeln. אם אין מספיק מקום בשורת הפלט, מציג המחשב את שאר פריטי הפלט בשורת הפלט הבאה. למשתנה המופיע ברשימת הפלט במשפט write או במשפט writeln צריך להיות ערך לפני ביצוע המשפט. אם למשתנה X אין ערך, אזי תוצאת ביצוע המשפט write(X) תלויה במחשב בו מבצעים את המשפט. במחשבים רבים, ערך המשתנה X הוא הערך שנשאר בתא המתאים בזיכרון המחשב מתכנית קודמת שבוצעה בו.

דוגמה 10.1... ראינו שבמשפט פלט ניתן להציג ערכי משתנים ומחרוזות של תווים. מהו תפקיד חשוב של שימוש במחרוזות תווים על אמצעי פלט? מחרוזות תווים המוצגות על אמצעי פלט משמשות להסבר משמעות המידע המוצג עליו.

דוגמה

★ נניח כי בסיום ביצוע תכנית ערכיהם של המשתנים Ones, Fives, Tens הם 1, 2, 3,

בהתאמה. ניתן היה לפלוט את המידע הזה באמצעות המשפט

```
write(Tens, Fives, Ones)
```

ולקבל על המסך את הפלט

```
3      1      2
```

אך באמצעות המשפטי writeln הבאים:

```
writeln ('(כמויות המטבעות בעודף הן: ');
writeln ('Tens, ' מטבעות של 10 אגורות');
writeln ('Fives, ' מטבעות של 5 אגורות');
writeln ('Ones, ' מטבעות של אגורה ');
```

יתקבל הפלט הבא, שהוא ברור יותר:

כמויות המטבעות בעודף הן:

3 מטבעות של 10 אגורות

1 מטבעות של 5 אגורות

2 מטבעות של אגורה

כזכור, ביצוע משפט read או משפט readln גורם למחשב להפסיק את ביצוע התכנית עד לקבלת נתונים מלוח המקשים. רצוי כי בתכנית תופיע הודעה למשתמש מתי להקיש את הנתונים. ההודעה תנתן על הצג באמצעות מחרוזות תווים.

דוגמה

```
writeln('הקש מחיר משקה וסכום ששולם:');
```

```
readln(Price, Payment);
```

ביצוע המשפט

```
writeln('הקש מחיר משקה וסכום ששולם:');
```

יגרום להצגה על המסך של ההודעות

הקש מחיר משקה וסכום ששולם:

לפני בקשת הנתון

```
readln(Price, Payment);
```

המשפט `writeln` יכול להופיע בתכנית גם ללא רשימת פלט. ביצוע משפט זה גורם למחשב לעבור לשורת הפלט הבאה.

דוגמאות

★ אם מתבצעים המשפטים

```
writeln('שורה ראשונה');
```

```
writeln('שורה שנייה');
```

על הצג יופיע

שורה ראשונה

שורה שנייה

אם מתבצעים המשפטים

```
writeln('שורה ראשונה');
```

```
writeln;
```

```
writeln('שורה שנייה');
```

על הצג יופיע

שורה ראשונה

שורה שנייה

★ ביצוע המשפט

```
writeln(A,B);
```

שקול לביצוע שני המשפטים

```
write(A,B);
```

```
writeln
```

שאלה 24:

הנח כי המשתנה `Money` מטיפוס שלם וערכו 99. מהו פלט קטע התכנית הבא?

```
write('שקלים חדשים', Money);
```

```
writeln('המחיר הוא');
```

```
writeln;
```

```
write('תודה ולהתראות!')
```

לכל פריט המופיע ברשימת הפלט של משפט `write` או של משפט `writeln`, ניתן לצרף רוחב שדה, המציין את מספר העמודות שבהן יוצג הפריט, מימין לשמאל. רוחב שדה של פריט מסוים מסומן על-ידי נקודתיים (:). וביטוי חשבוני מטיפוס שלם לציון מספר העמודות.

★ נניח כי המשתנה Year מטיפוס שלם וערכו 1977. אם מתבצעים המשפטים:

```
writeln ('בשנת', Year:4, 'חל מהפך פוליטי בישראל');
writeln ('5:בשנת', Year:5, '21:חל מהפך פוליטי בישראל');
writeln ('10:בשנת', Year:10, '25:חל מהפך פוליטי בישראל');
writeln ('10:בשנת', Year+15:10, '25:חל מהפך פוליטי בישראל');
```

יופיע על הצג:

```
בשנת 1977 חל מהפך פוליטי בישראל
בשנת 1977 חל מהפך פוליטי בישראל
בשנת 1977 חל מהפך פוליטי בישראל
בשנת 1992 חל מהפך פוליטי בישראל
```

כאשר רוחב השדה גדול ממספר העמודות שתופס הפריט, מודפסים רווחים לפניו; כלומר, לשמאלו. ציון רוחב השדה של פריטי הפלט מאפשר למתכנת לשלוט בעריכת הפלט על הצג.

מספרים מטיפוס ממשי אינם מוצגים על המסך בדרך שלה היינו מצפים.

דוגמה

★ נניח כי המשתנה X מטיפוס ממשי וערכו 2178.56. אם מתבצע המשפט

```
write(X);
```

על הצג יופיע

```
2.17856000000000E03
```

אשר פרושו

```
2.17856000000000*103
```

הצגת מספר מטיפוס ממשי, נעשית באמצעות שני ביטויים מטיפוס שלם - רוחב שדה ורוחב שבר. רוחב שבר מציין את מספר הספרות אחרי הנקודה העשרונית שיופיעו על הצג.

דוגמאות

★ נניח כי המשתנה X מטיפוס ממשי וערכו 2178.56.

אם מתבצעים המשפטים

```
writeln(X:8:3);
writeln(X:8:2);
writeln(X:12:2);
writeln(X:12:3);
```

על הצג יופיע

```
2178.560
2178.56
2178.56
2178.560
```

writeln(42/4:10:1, 42/4:10:2, 42/4:10:3)

על הצג יופיע

10.5 10.50 10.500

שאלה 25:

הנח כי המשתנה Number מטיפוס שלם וערכו 123. מה יופיע על הצג אם יתבצעו המשפטים הבאים?

```
writeln('סוף המספר', Number:5, 'תחילת המספר');
writeln('15:סוף המספר', Number:5, '15:תחילת המספר');
```

שאלה 26:

הנח כי המשתנה N מטיפוס שלם וערכו 5. מה יופיע על הצג אם יתבצע המשפט הבא?

```
write('N=:N, N+1:N);
```

שאלה 27:

הנח כי המשתנים Key ו- Limit מטיפוס שלם וערכיהם 3 ו- 5, בהתאמה. מה יופיע על הצג אם יתבצע המשפט הבא?

```
writeln('Key+Limit=', Key:1, '+', Limit:1, '=', Key+Limit:1);
```

שאלה 28:

הנח כי המשתנה N מטיפוס שלם וערכו 7. מה יופיע על הצג אם יתבצע המשפט הבא?

```
writeln(2:2, '=', N+3:2, '?');
writeln('HA!':N, 'HA!':N-1, 'HA!':N-2, 'HA!':4);
```

תשובה 28 (מופיעה רק במדריך למורה):

```
2=10?
```

```
HA! HA! HA! HA!
```

**שאלה 29:**

נניח כי בתכנית מופיעה הצהרת המשתנים

```
var X, Y: real;
    Z, W: integer;
```

ונניח כי ערכי המשתנים X, Y, Z, W הם 89.5, -6.05, 91, -8091, בהתאמה. כתוב קטעי תכניות להצגת שורות הפלט הבאות.

א. 89.500 Z=91

W= -8091

ב. הנתונים הם:

89.5, 91

-6.05, -8091

בין החלקים השונים של תכנית פסקל או אף בתוכם מופיעות הערות.

הערה בתכנית פסקל יכולה להופיע באחת משתי הדרכים הבאות:

{ הערה }

או

(* הערה *)

ניתן לכתוב הערה בכל מקום בתכנית. הערה יכולה להתחיל ולהסתיים באותה שורה, או להשתרע על-פני מספר שורות.

דוגמה

★ { חישוב מועד נחיתה, בשעות ודקות }

הערה איננה הוראת פעולה למחשב. בזמן הידור תכנית, מתעלם המהדר מן ההערות המופיעות בה; כלומר, אינו מתייחס לתווים הנמצאים בין הסוגריים.

כאמור, הערות מיועדות למתכנת או למשתמש בתכנית, ולא למחשב. הן עוזרות בתיעוד (documentation) תכנית, הכולל, בין השאר, תאור משימת התכנית ותפקידי משתנים.

אם שוכחים לסגור הערה, באמצעות { או (*), המהדר לא מבחין בכך ולא מודיע על שגיאת תחביר, אך תהליך הידור עלול להתבצע בצורה שונה מן הרצוי.

דוגמה

★ { חיבור תוספת למשכורת העובד }

Salary := Salary+Bonus;

{ הורדת מיסים ממשכורת העובד }

Salary := Salary-Tax;

המהדר מתייחס לתו { הסוגר את ההערה השנייה כסוגר את ההערה הראשונה, שממנה הושמט תו זה. כל שלוש השורות הראשונות נחשבות הערה, ולכן, המשפט Salary := Salary+Bonus לא יתבצע, והתוספת לא תחובר למשכורת... ניתן לשאול מדוע המהדר לא מודיע על התו {, הפותח את ההערה השנייה. המהדר מזווג כל תו { עם תו } ומתעלם מכל התווים המופיעים ביניהם.
