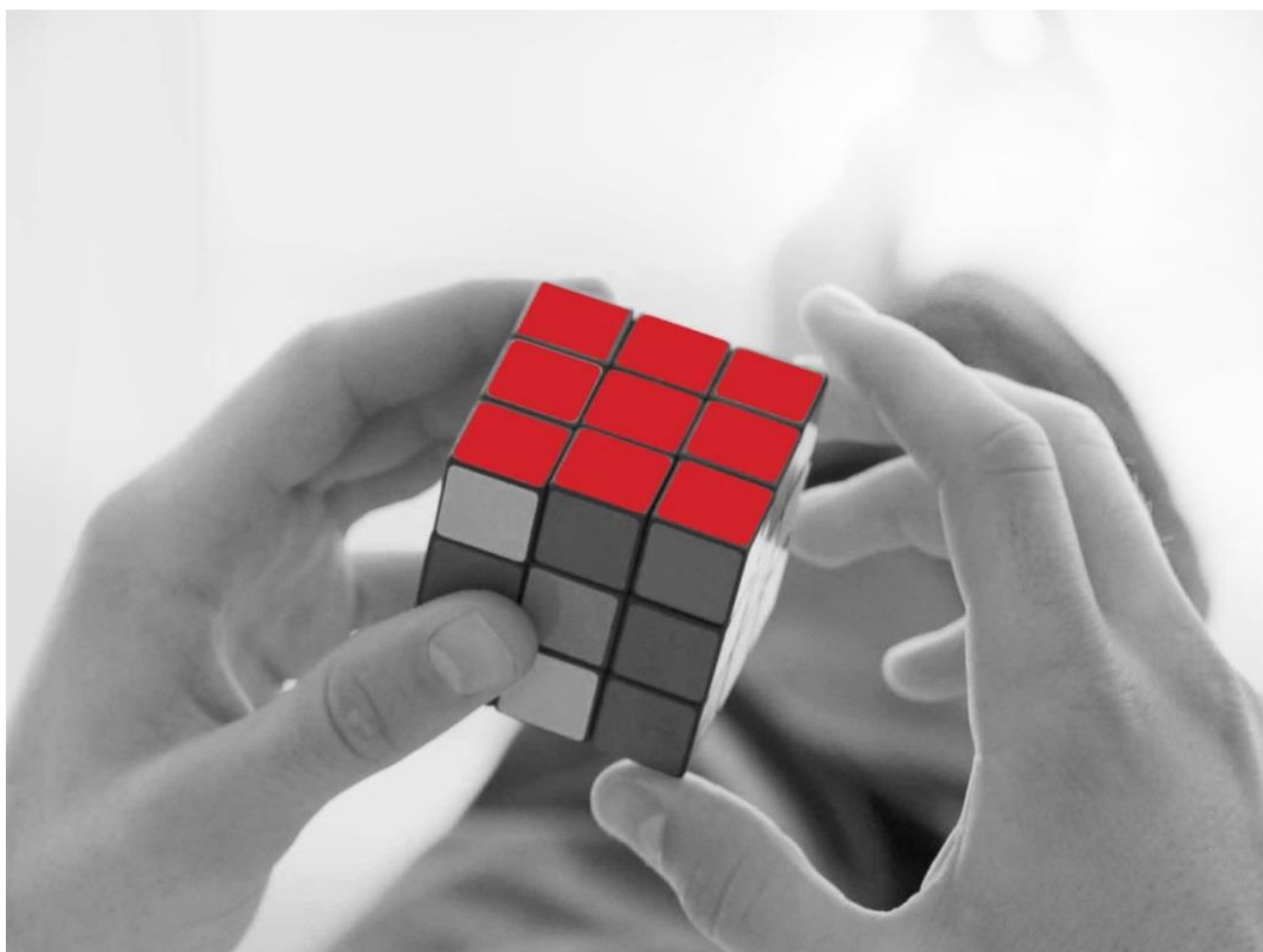


מבוא לתכנות



תוכן עניינים

5	1. התחלה
5	1.1 אלגוריתם
6	1.2 מודל הפקיד
8	1.3 תרשים זרימה
9	2. פעולות בסיסיות
9	2.1 קלט
10	2.2 פלט
12	2.3 אופרטורים
13	2.4 השמה
15	תרגילים
16	3. תנאים
16	3.1 תנאי פשוט
18	3.2 תנאי מקונן
19	3.3 תנאי מורכב
21	4. לולאות
22	4.1 לולאת תנאי
24	4.2 הרצה יבשה בטבלה
27	4.3 קיבול לולאת תנאי
30	4.4 לולאת אינדקס
34	4.5 סיום מוקדם של איטרציה ושל לולאה
36	4.6 קיבול לולאת אינדקס
39	5. פונקציות
39	5.1 פונקציה
40	5.2 פרמטרים
42	5.3 סיום מוקדם של תרשים
44	6. מערכים
44	6.1 מהו מערך
44	6.2 שימוש במערך
47	6.3 מיון מערך
47	6.4 חיפוש במערך
48	6.5 מהו מערך דו-ממדי (מטריצה)
50	7. סיום
56	8. נספחים
56	נספח א הצמתים בתרשים הזרימה
59	נספח ב האופרטורים בתרשים הזרימה
60	נספח ג מונחון

1. התחלה

מטרת חוברת זו הינה הבנת המושגים הבסיסיים במודול.

1.1 אלגוריתם



אפשר להגדיר אלגוריתם כך:
סידרת הוראות בסיסיות לביצוע משימה מורכבת.

על אלגוריתם להיות חד משמעי וברור.

נבחן את 2 הדרישות הללו על דוגמא מחיינו.

אלגוריתם למשלוח פיצה:

1. קח את הכתובת למשלוח.
2. קח את הפיצה מהחנות.
3. סע לבית הלקוח (הכתובת למשלוח).
4. מסור את הפיצה.
5. אסוף את הכסף.
6. חזור לחנות.

למי מיועד האלגוריתם הזה?

כפי שניתן להניח, האלגוריתם הנ"ל הינו אלגוריתם לשליח פיצה. במקרה זה, ה"משימה המורכבת" היא משלוח של פיצה. בכדי שהשליח יוכל להשלים את המשימה בהצלחה, הוא מפשט אותה למספר שלבים מוגדרים, החל מקבלת הכתובת, לקיחת הפיצה ועד לחזרה לחנות עם הכסף. כפי שניתן לראות מדוגמא זאת, האלגוריתם ברור וחד משמעי שכן הוא מחולק לשלבים כאשר ההוראות בכל שלב הן בסיסיות.

האם אותו אלגוריתם יתאים למוכר במכולת?

לא. הסיבה לכך טמונה בעובדה, שמוכר במכולת אינו מכיר את כל ההוראות הבסיסיות, אותן מכיר שליח פיצה. במקרה של מוכר במכולת, כאשר הוא יגיע לשורה של "סע לבית הלקוח" הוא ייתקע, שכן אין הוא יודע איך מבצעים את הפקודה "סע".

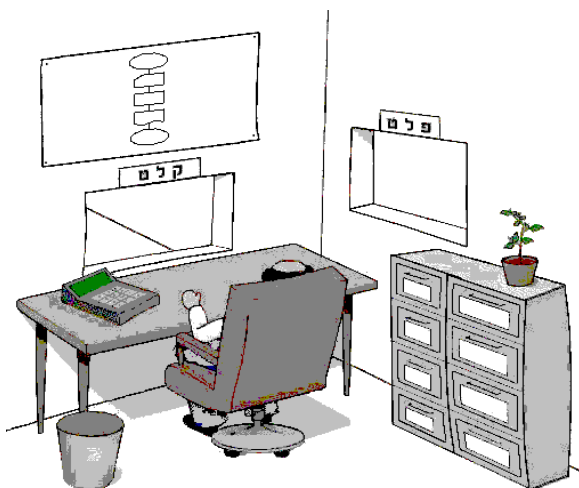
כעת, נחזור לעולם המחשבים. כידוע, בניגוד לבני אדם, המחשב אינו בעל אינטליגנציה ואין הוא מסוגל לשער מה כוונתנו. לכן, עלינו להיות חד משמעיים כשאנחנו כותבים את האלגוריתם.

אם כך, מדוע עלינו גם להיות ברורים? לרוב, אין מתכנת בודד עובד על תוכנה שלמה, אם כי הוא עובד בצוות. חובה עלינו להיות ברורים בקוד, על מנת לאפשר לחברינו לצוות להבין מה אנחנו עושים. לעיתים קרובות, הם יצטרכו להמשיך או להתחבר לעבודה שלנו.

במודול זה, נלמד לפתח ולנסח אלגוריתמים. כמו כן, כחלק מהכרת עולם האלגוריתמים נלמד גם מושגים שיהיו רלוונטיים בהמשך.

1.2 מודל הפקיד

חשוב על המחשב בתור פקיד ועל התוכניתן בתור מנהל העבודה שלו.



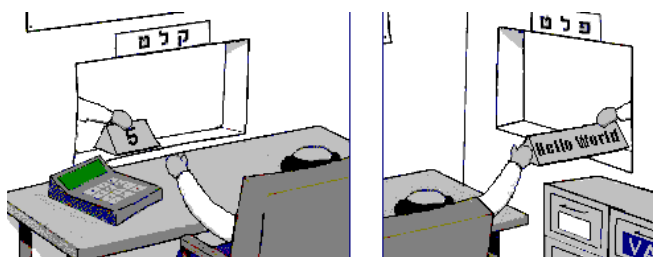
הפקיד

הפקיד יודע למלא אחר הוראות בסיסיות. הוא לא מבין את המשמעות של מה שהוא עושה ולכן גם לא מפעיל שיקול דעת. תפקידך, מנהל העבודה, הוא לנסח הוראות עבור הפקיד כך שיבצע נכון את המשימה בכל מצב.

משרדו של הפקיד

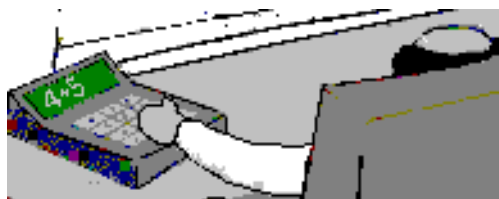
הפקיד נעזר בצידוד שבמשרדו:

- מחשבון – הפקיד מבצע חישובים פשוטים במחשבון.
- ארון עם תאים – על מנת לשמור ערך, הפקיד מאחסן אותו בתא (המונח המקובל לתא בזיכרון הוא משתנה).
- חלונות – הקשר של הפקיד עם החוץ.
- חלון קלט – הפקיד שולף את הנתונים עמם הוא עובד מתור הקלט.
- חלון פלט – הפקיד מדפיס את תוצאות עבודתו החוצה דרך חלון הפלט.
- לוח הוראות – הפקיד מבצע אך ורק ובדיוק את ההוראות שעל הלוח.



הנתונים עליהם עובד הפקיד נמצאים בתור הקלט. בכל פעם שהפקיד מגיע להוראת קלט, הוא מוציא את הנתון הראשון בתור (משמאל למעלה) ומאחסן את ערכו. את הוראת הקלט נכיר בפרק 2.1 קלט.

התוצר של עבודת הפקיד הוא מה שיוציא לפלט (מימין למעלה). את הוראת הפלט נכיר בפרק 2.2 פלט.



את כל החישובים מבצע הפקיד במחשבון. הפעולות החשבוניות שניתן לבצע מפורטות בפרק 2.3 אופרטורים



הפקיד לא זוכר נתונים שנקלטו או תוצאות של חישובים.
הוא מאחסן כל ערך בתא. בכל תא שמור ערך אחד
ברגע נתון.

לכל תא יש **מזהה** (שם ייחודי) כדי שהפקיד יפנה לתא
המתאים כשיזדקק לערך השמור. על השם להיות **בעל**
משמעות מתאימה לערך שהוא מכיל.

השפה בה כותבים את ההוראות לפקיד היא **תרשים**
זרימה.

הבהרה על התאים (מגירות בזיכרון)

נסה לחשוב מה קרה כששמעת את שם המרצה בפעם הראשונה. ברגע שבו שמעת את שמו של המדריך, ניסית לזכור אותו. למעשה, אחסנת את המידע בזיכרון. ניתן לומר כי, פתחת במוח מגירה ששמה "שם המדריך" ובתוכה שמת את הנתון - "יהודה". כעת, הנתון שמור וגם נדע כיצד לגשת אל המידע. כדי לקרוא למדריך, כל מה שצריך לעשות הוא למצוא את המגירה "שם המדריך" ולשלוף את הערך משם.

בדוגמא הנ"ל השתמשנו בדימוי של מגירה לתא בזיכרון. בנוסף, לכל מגירה ניתן שם ייחודי אשר איפשר לנו לגשת למידע השמור בתוכה ו/או לשמור בתוכה מידע נוסף. למעשה, אותו שם ייחודי נקרא **מזהה** שכן תפקיד לזהות את התא באופן חד-חד-ערכי.



1.3 תרשים זרימה

תרשים זרימה הוא שרטוט המתאר אלגוריתם.



צמתים

תרשים הזרימה מורכב מצמתים המחוברים ביניהם בקו. מחוץ לצמתים ולקווים רושמים הערות. ישנם 3 סוגי צמתים. לכל סוג יש משמעות וכיוון זרימה קבועים.

1. צומת start וצומת stop



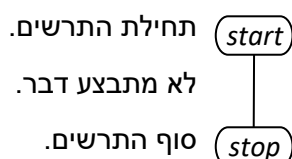
2. צומת start הוא סגלגל (אליפסה) ומופיע בו הכיתוב start. משמעות: תחילת האלגוריתם. כיוון זרימה: הצומת הבא לביצוע נמצא בהמשך הקו היורד.

3. צומת stop הוא סגלגל ומופיע בו הכיתוב stop. משמעות: סוף האלגוריתם. כיוון זרימה: אין צמתים נוספים.

בכל תרשים זרימה יש צומת start אחד ויחיד וצומת stop אחד ויחיד. בפרק 4 יוצג תרשים זרימה, בו זוג צמתים אחר תופס את מקומם של start ו-stop.



האלגוריתם המתואר בתרשים הבא אינו מבצע דבר:



2. פעולות בסיסיות

2.1 קלט

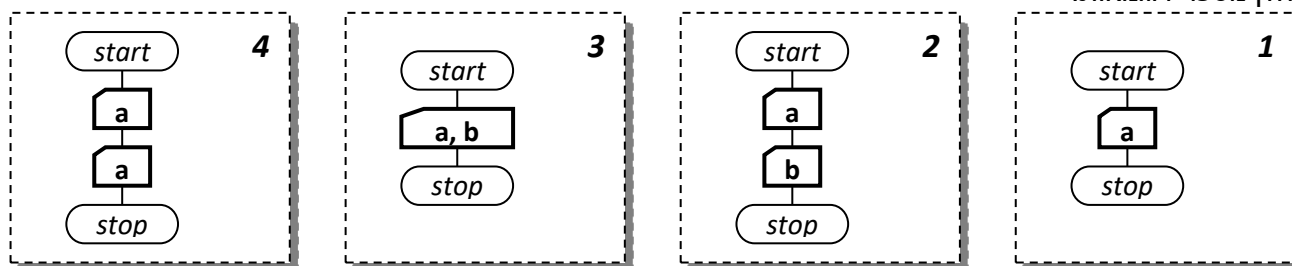


קבלת קלט הוא תהליך, אשר במהלכו המחשב יקבל נתונים חיצוניים (משמע, בכל אמצעי שאינו המחשב. הדוגמאות הבולטות לאמצעי מסוג זה הן מקלדת, עכבר, קורא כרטיסים וכו'), לצורך עיבודם באמצעות תוכנית מחשב. כיוון שיכול להיווצר מצב בו מתקבלים מספר נתונים, נהוג לומר כי הנתונים ממתינים לעיבוד **בתור הקלט**. הנתונים עצמם נקראים **קלט** ולרוב הם יישמרו בתא בזיכרון (משתנה).

תחביר צומת קלט

בתוך צומת קלט רושמים את המזהה של המשתנה בו יש לאחסן את הנתון. אם המזהה מופיע לראשונה בתרשים, יוצר בזיכרון משתנה חדש והוא יישמש בכדי לשמור את הנתונים.

להלן מספר דוגמאות:



A מזהה משתנה חדש. **קלט** נתון ושומר במשתנה A.

תרשים מספר 1:

A מזהה משתנה חדש. קלט נתון ושומר במשתנה A.
B מזהה משתנה חדש. קלט נתון ושומר במשתנה B.

תרשים מספר 2:

ניתן לקצר ולרשום מספר הוראות קלט בצומת קלט בודד. למעשה, תרשים 3 שקול לתרשים 2, שכן מבצע את אותה המשימה באותה היעילות בדיוק.

תרשים מספר 3:

A מזהה משתנה חדש. קלט נתון ושומר במשתנה A. קלט נתון ושומר במשתנה A במקום הנתון הקודם (המושג המקובל לפעולה זו הוא "דריסה", שכן אנו מוחקים את הערך הקודם ושומרים במקומו ערך חדש. פעולה זו נפוצה מאוד בתכנות).

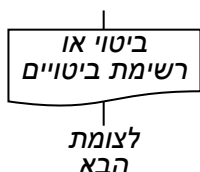
תרשים מספר 4:

2.2 פלט

פלט הוא התוצר של הרצת אלגוריתם. ברוב המקרים תוצאת הפלט תשתנה, בהתאם לקלט שהתקבל במהלך האלגוריתם. לרוב, ובמיוחד לאורך המודול הנוכחי, נתייחס לפלט כהדפסה אל המסך.

תחביר צומת פלט

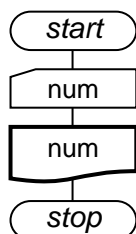
בתוך צומת פלט רושמים **ביטוי** או רשימה של ביטויים.



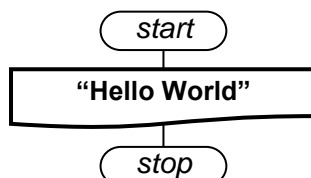
ביטוי יכול להיות:

- קבוע מספרי (ערך שאינו משתנה).
- טקסט בתוך מרכאות.
- מזהה (שם) של משתנה.
- חישוב (יוסבר בפירוק בפרק אופרטורים).

מזהה (שם) של משתנה



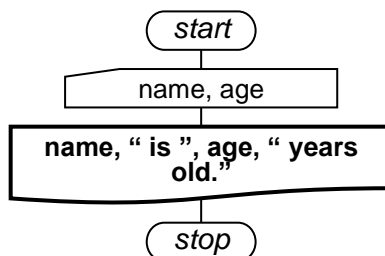
טקסט



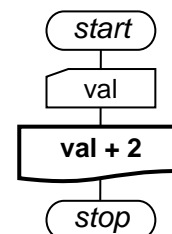
קבוע מספרי



רשימה של ביטויים



חישוב



טיפ:

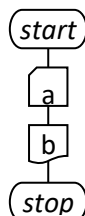
בכל פתרון יש לפחות צומת פלט אחד!



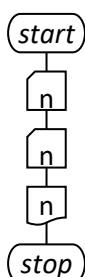
לידיעתך: הסימן כוכבית (*) בתחילת התרגיל – פתרון בחוברת "אסטרטגיה ופתרונות".



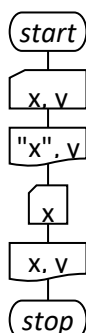
1. מה לא תקין בתרשים? הצע תיקון.



2. מה המשימה שמבצע התרשים הבא?



3. * קלוט שני ערכים והצג אותם.
4. * קלוט שני ערכים והצג אותם בסדר הפוך מסדר קליטתם.
5. * קלוט שני ערכים והצג את השני שנקלט.
6. * קלוט שני ערכים והצג את הראשון שנקלט.
7. * קלוט ערך והצג אותו שלוש פעמים.
8. מה יהיה הפלט עבור התרשים הבא, אם בתור הקלט מתקבלים הנתונים הבאים:
→ 15, 32, 1, 9



2.3 אופרטורים

חישוב

ביטוי חשבוני מורכב מאופרטורים ומאופרנדים (operator) ואופרנט (operand) או אופרנד.

- אופרטור הוא סימן המייצג פעולה.
- אופרנט הוא ערך שעליו מבוצעת הפעולה.

הבהרה: בביטוי $1+2$

הסימן '+' הוא אופרטור והוא מבצע פעולה על האופרנדים (המספרים).

- לכל אופרטור יש קדימות: כשמעריכים ביטוי מתחילים מהאופרטורים בעלי הקדימות הגבוהה ביותר (1), ויורדים בהדרגה. כדי לשנות את סדר הפעולות, משתמשים בסוגריים.

האופרטורים החשבוניים

a ו- b הם ביטויים מספריים.

קדימות	אופרטור	משמעות
1	(a) סוגריים.	a (משנה את סדר החישוב).
	$ a $ ערך מוחלט.	אם a שלילי $-a$, אחרת a .
	$+a$ פלוס.	a .
	$-a$ מינוס.	הופך את a משלילי לחיובי, ולהפך.
2	$a * b$ כפל.	a פי b .
	a / b חילוק.	a חלקי b .
	$a \% b$ מודולו.	השארית מחלוקה של a ב- b (שלמים בלבד).
3	$a + b$ חיבור.	a ועוד b .
	$a - b$ חיסור.	a פחות b .

2.4 השמה

לעתים נרצה שיתבצעו חישובי ביניים שלא יופיעו בפלט. בתכנות לחישובים, אין משמעות אם לא שומרים את תוצאתם (עד עתה הדפסנו את התוצאה, אך לא יכולנו להמשיך את התוכנית עם התוצאה של החישובים. לכן, יש צורך בהשמה =).

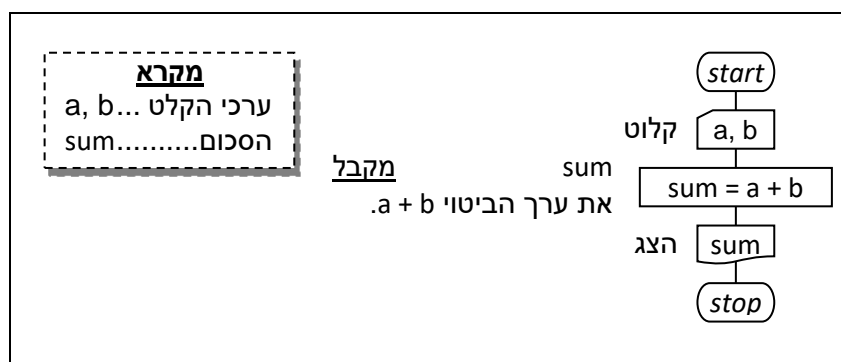
תחביר צומת השמה

ביטוי = תא

לצומת
הבא

ביטוי יכול להיות קבוע מספרי, מלל בתוך מרכאות, מזהה של משתנה או חישוב. את ערך הביטוי מעתיקים לתוך המשתנה.

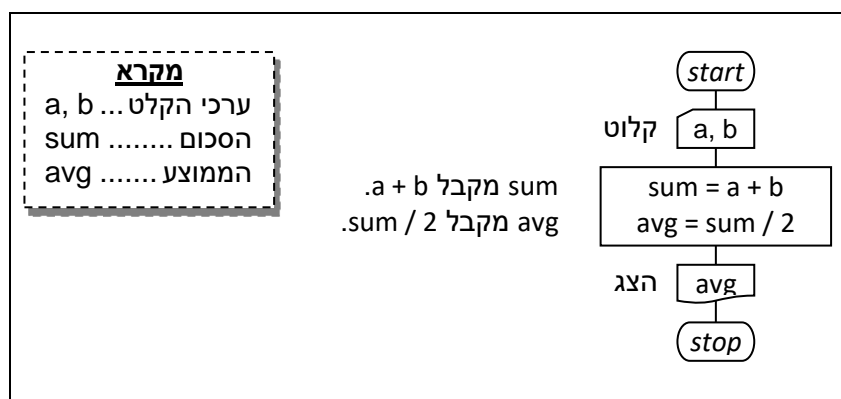
דוגמאות:



קליטת שני מספרים והצגת סכומם

ניתן לרכז מספר השמות בצומת השמה יחיד.

ההשמות תתבצענה אחת אחרי השנייה, מלמעלה למטה.



קליטת שני מספרים והצגת הממוצע

מהו מקרא?

השמה פותחת לנו אפשרות שמירת נתונים במשתנים נוספים מעבר לנתונים שבתור הקלט. על אלה נכריז בצד של התרשים עם תיאור קצר מה תפקידם לאחסן. שימו לב, אין חובה לציין את המשתנים של הקלט (a,b).

מודולו

$A \% B$ (A מודולו B) הוא השארית מחלוקה של A ב-B.

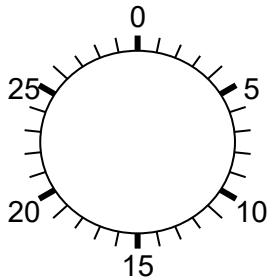
$$100 \% 30 = 10$$

כלומר, 30 נכנס 3 פעמים בשלמות ב-100 ($100 / 30 = 3\frac{1}{3}$).

אבל 3 פעמים 30 הם 90, ונותרנו עם שארית (מודולו) 10.

המודולו שימושי כדי:

- לדעת אם שלם אחד מתחלק בשני ללא שארית (כלומר השארית היא 0). בדוגמא למעלה, 100 לא מתחלק ב-30 ללא שארית.
- לתאר ציר מספרים "מעגלי": מ-0 עד שלם מסוים וחוזר ל-0. לכל שלם מודולו 30 תהיה התוצאה בין 0 לבין 29. בשרטוט שלפניך ציר מעגלי עם שנתות מ-0 עד 29. כדי לחשב $30 \% 100$, בעזרת השרטוט, יש להתחיל משנת 0, ולהתקדם 100 שנתות עם כיוון השעון. כשנסיים נעמוד על שנת 10.



טיפ:

חלקו את המספר חלוקה רגילה אך עם שברים ולא חילוק עשרוני. השאירו את המספרים כמו שהם, ללא צמצום.



$$14 \% 4 \rightarrow \frac{14}{4} = 3\frac{2}{4} \rightarrow 2$$

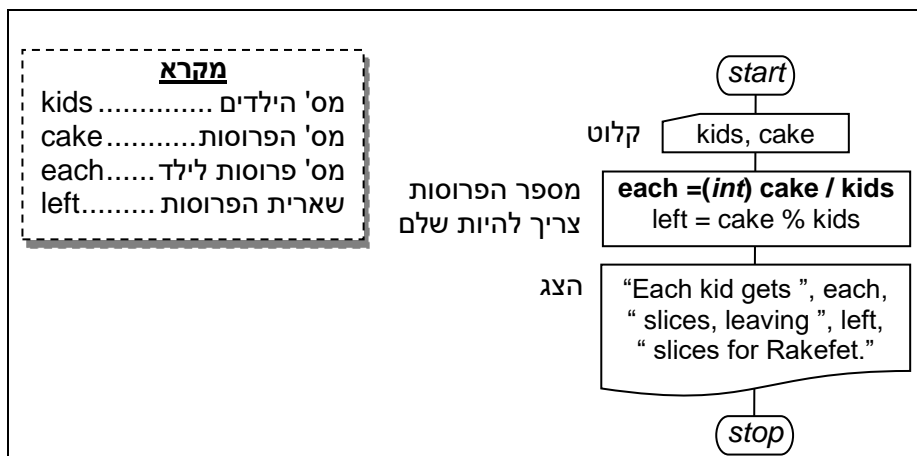
השמה שלמה

האופרטור (int) מקצץ את חלק השבר מביטוי ומשאיר את החלק השלם.

לדוגמא:

קליטת מספר הילדים בגן רקפת ומספר פרוסות העוגה.

הצגת מספר הפרוסות שאפשר לתת לכל ילד ומספר הפרוסות שישארו לגננת.



תרגילים



1. * קלוט שלושה ערכים. הצג את סכומם ואת הממוצע שלהם.
 2. קלוט אורך ורוחב של חדר מלבני. הצג את שטח החדר ואת היקפו.
 3. * קלוט קוטר ועומק של סיר והצג את הקיבולת שלו.
(שטח עיגול הוא πR^2 . $\pi \approx 3.14$, ו-R הוא בדיוק $\frac{1}{2}$ מהקוטר).
 4. קלוט אורך של סרט קולנוע בדקות והצג את אורך הסרט בשעות ודקות.
לדוגמא: קלט – 88, פלט – 1 hour(s) and 28 minute(s).
 5. קלוט מספר, מובטח כי הוא בן 4 ספרות לפחות. מצא מהי ספרתו הימנית ביותר והדפס אותה.
 6. קלוט מספר, מובטח כי הוא בן 4 ספרות לפחות. מצא מהי ספרתו השניה מימין והדפס אותה.
 7. קלוט נתון תלת ספרתי והדפס את ספרת המאות (מובטח כי אינה 0).
 8. קלוט נתון דו ספרתי והדפס את סכום ספרותיו.
 9. בתור הקלט שלם בין 10 לבין 99.
* הפוך את סדר הספרות והצג את המספר החדש.
לדוגמא: קלט – 61, פלט – 16.
- ⊗ לא הבנתי את השאלה!
כשנתקלים בניסוח שונה, כדאי להפוך אותו למשפט "קלוט... והצג...".
להלן ניסוח אחר של השאלה:
קלוט שלם בין 10 לבין 99,
הפוך את סדר הספרות והצג את המספר החדש.
10. * קלוט מספר והצג את השלם הזוגי הגדול ממנו והקרוב ביותר אליו.
לדוגמא: קלט – 4.1, פלט – 6.

3. תנאים

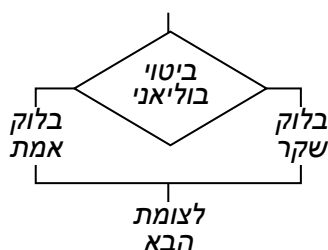
3.1 תנאי פשוט

לעתים, נרצה להתנות ביצוע בלוק (קטע תרשים) בתנאי מסוים.

תחביר צומת תנאי

ביטוי בוליאני הוא ביטוי שערכו נכון יכול להיות או אמת או שקר

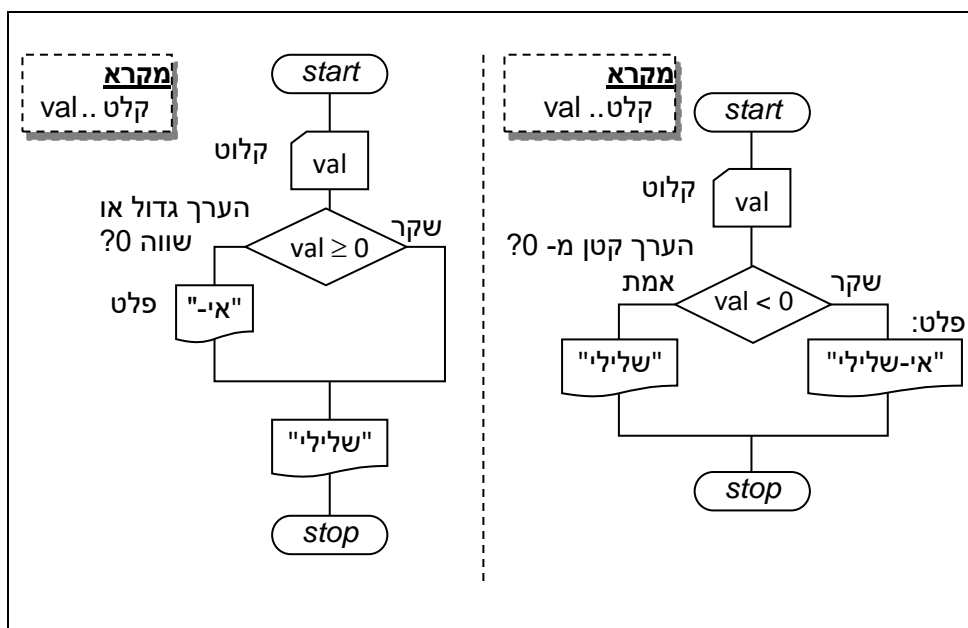
אם ברגע הבדיקה הביטוי אמת – יתבצע הבלוק השמאלי בלבד.
אחרת (הביטוי שקר) – יתבצע הבלוק הימני בלבד.



דוגמא:

קליטת ערך והצגת "שלילי"
או "אי-שלילי" בהתאם
לערכו.

שני התרשימים הבאים
שקולים:



ישנם שישה אופרטורים המרכיבים ביטוי בוליאני משני ביטויים.

בטבלה הבאה a ו- b הם שני ביטויים:

קדימות	אופרטור	אמת כאשר...
4	$a = b$ שווה	לשני הביטויים אותו ערך.
	$a > b$ גדול מ-	ל- a ערך גבוה מ- b .
	$a < b$ קטן מ-	ל- a ערך נמוך מ- b .
	$a \neq b$ שונה מ-	לשני הביטויים אין את אותו ערך.
	$a \leq b$ קטן/שווה	ל- a ערך נמוך מ- b או שווה לו.
	$a \geq b$ גדול/שווה	ל- a ערך גבוה מ- b או שווה לו.

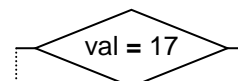
צומת תנאי מפצל תרשים לשניים: בלוק אמת משמאל ובלוק שקר מימין.

- בסוף שני הבלוקים התרשים מתאחד, כלומר לכל תנאי יש תמיד כניסה אחת בדיוק ויציאה אחת בדיוק, כאשר בכל תרשים יש רק צומת *stop* אחד.
- כאשר משתמשים בביטוי בוליאני רק אחד מהבלוקים יכול להתבצע, משמע, לא קיים מצב שבו ביטוי אחד יהיה אמת ושקר באותו הזמן.
- יש להבדיל בין שני התפקידים של סימן השווה (=).

בתפקידו הראשון סימן השווה מתפקד כאופרטור ההשמה (כאשר הוא בצומת השמה) והוא ישנה את ערך התא הרשום משמאל לערך הביטוי מימין.

val = 17

בתפקידו השני סימן השווה מתפקד כאופרטור השוואה (כאשר הוא בצומת תנאי) והוא לא ישנה ערך של תא. אופרטור השוואה הוא סימטרי. בדוגמא מימין אפשר היה לכתוב אותה המידה $17 = \text{val}$.



תרגילים

1. קלוט שני ערכים והצג "Growing..." אם השני גדול מהראשון בלבד.
2. קלוט שני ערכים והצג את הערך הגדול ביותר (בשוויון הראה את אחד מהם).
3. *קלוט שלם והצג "Even" אם הוא זוגי ו-"Odd" אם הוא לא.
4. קלוט שני שלמים. הצג האם הראשון מתחלק בשני ללא שארית ובנוסף, האם השני מתחלק בראשון ללא שארית.
5. *קלוט שני ערכים והצג קודם את הערך הקטן ואת הערך הגדול אחריו (בשוויון הצג את שניהם).



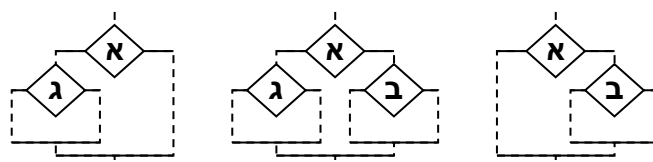
תרגיל אתגר:

6. * בעל בית תוכנה החליט להעלות את המשכורת של כל תכניתן ב-10%, בתנאי שלאחר העלאה כזו הסכום לא יהיה גבוה מ-6,000 ש"ח.
- אם הסכום אכן יהיה גבוה מ-6,000 ש"ח, יקבל אותו תכניתן העלאה של 5% בלבד. שרטט תרשים עבור המשימה הבאה: קלוט שם של תכניתן ואת משכורתו הנוכחית. הצג את המשכורת של התכניתן לאחר ההעלאה.

3.2 תנאי מקונן

תנאי מפצל את התרשים לשני בלוקים. אפשר להמשיך ולפצל כל בלוק ע"י תנאי נוסף. תנאי בתוך בלוק של תנאי אחר נקרא **תנאי מקונן**.

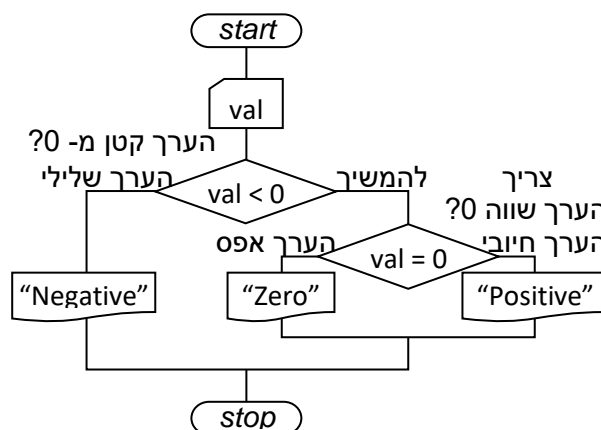
תחביר צומת תנאי



תרגילים

1. קלוט ערך והצג "Negative", "Zero", או "Positive" בהתאם לערכו, עם תנאי מקונן והשווה עם

מקרא
val
קלט



הפתרון למטה.

2. קלוט שלושה ערכים.

הצג "Increasing..." אם השני גדול מהראשון והשלישי גדול מהשני בלבד.

3. *קלוט שלושה ערכים והצג את הערך הגדול ביותר (במקרה של שוויון של אחד מהם).

4. בתור הקלט שלם בין 1 לבין 9,999. הצג את השלם ואת מספר ספרותיו.

5. * אגף מס הכנסה קבע מדרגות מס באופן הבא:

על כל שקל מ- 23,000 הש"ח הראשונים – שיעור המס הוא 10%.

על כל שקל מ- 23,000 הש"ח הבאים – שיעור המס הוא 20%.

על כל שקל מ- 74,000 הש"ח הבאים – שיעור המס הוא 30%.

על כל שקל מ- 100,000 הש"ח הבאים – שיעור המס הוא 40%.

על כל שקל נוסף – שיעור המס הוא 50%.

שרטט תרשים עבור המשימה הבאה:

קלוט שם של עצמאי ואת הכנסתו.

הצג את הסכום שעל העצמאי לשלם למס הכנסה (קרא היטב את הגדרות המס).

3.3 תנאי מורכב

ביטוי בוליאני מורכב

ישנם שני אופרטורים המרכיבים ביטוי בוליאני שלישי משני ביטויים בוליאניים.
בטבלה הבאה a ו- b הם שני ביטויים בוליאניים:

קדימות	אופרטור	אמת כאשר...
5	$a \&\& b$ וגם	שני הביטויים הם אמת.
6	$a b$ או	לפחות אחד הביטויים אמת.

פעולת **וגם** $\&\&$ קודמת לפעולת **או** $||$.

כשמרכיבים מספר ביטויים בוליאניים, חלק עם האופרטור $\&\&$ וחלק עם האופרטור $||$, מומלץ להשתמש בסוגריים כדי למנוע טעויות ובשביל הבהירות.

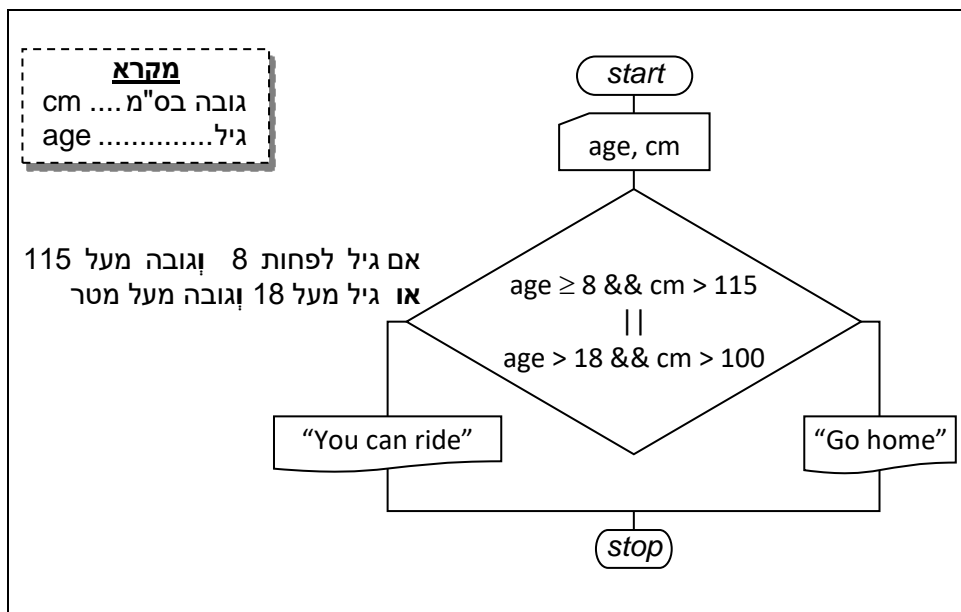


1. קלוט נתון. הדפס האם הנתון מתחלק ב-3 או ב-5 בשלמות.



לדוגמא:

מותר לעלות לרכבת ההרים
מגיל 8 עד 18 אם גובהך
מעל 115 ס"מ.
אחרי גיל 18 מותר לעלות
גם אם גובהך רק מעל 100
ס"מ.
קלוט גיל וגובה בס"מ של
איש והצג האם מותר לאיש
לעלות למתקן.



פתור את התרגיל שבדוגמא עם תנאים מקוננים פעמיים:
עם תנאי מורכב ופעם ללא. מצא את ההבדלים.



תרגילים



1. * קלוט שלושה ערכים והצג אותם מהקטן לגדול.

2. מנהל ביה"ס "יסוד" החליט שלא יופיעו בתעודות הציונים המספריים, אלא הערכה מילולית לפי המפתח הבא:

פחות מ- 55 \leftarrow בלתי מספיק.

55 עד 64 \leftarrow מספיק.

65 עד 74 \leftarrow כמעט טוב.

75 עד 84 \leftarrow טוב.

85 עד 94 \leftarrow טוב מאד.

95 ומעלה \leftarrow מצוין.

קלוט ציון של תלמיד והצג את ההערכה המילולית המתאימה.

3. * במערכת המשוואות הבאה A עד F הם מקדמים ו-x ו-y הם נעלמים:

$$A \times x + B \times y = C$$

$$D \times x + E \times y = F$$

ניתן לחשב את x ואת y ע"י נוסחאות העזר הבאות:

$$x = \frac{C \times E - B \times F}{A \times E - B \times D}$$

$$y = \frac{A \times F - C \times D}{A \times E - B \times D}$$

קלוט את המקדמים A עד F והצג את x ואת y לפי נוסחאות העזר.

המנע מחלוקה ב- 0 בנוסחאות העזר!

אם המכנה 0, אין פתרון. במקרה כזה הצג "Equation has no solution".

4. * שנה מעוברת היא שנה שמתחלקת ב- 4 ללא שארית,

וגם לא מתחלקת ב- 100, אלא אם היא מתחלקת ב- 400.

לדוגמא: 2002 לא מתחלקת ב- 4 \leftarrow 2002 רגילה.

2004 מתחלקת ב- 4 ולא ב- 100 \leftarrow 2004 מעוברת.

1900 מתחלקת ב- 4 וב- 100 ולא ב- 400 \leftarrow 1900 רגילה.

2000 מתחלקת ב- 4 וב- 100 וב- 400 \leftarrow 2000 מעוברת.

קלוט שנה והצג האם היא שנה מעוברת.

5. מספר הימים בכל חודש הוא קבוע, למעט פברואר (2):

חודש	1	2	3	4	5	6	7	8	9	10	11	12
ימים	31	28/9	31	30	31	30	31	31	30	31	30	31

בשנה רגילה יש בפברואר 28 ימים, ובשנה מעוברת – 29.

קלוט את מספר החודש ואת השנה והצג את מספר הימים באותו החודש.

מותר להיעזר בפתרון שאלה 4.

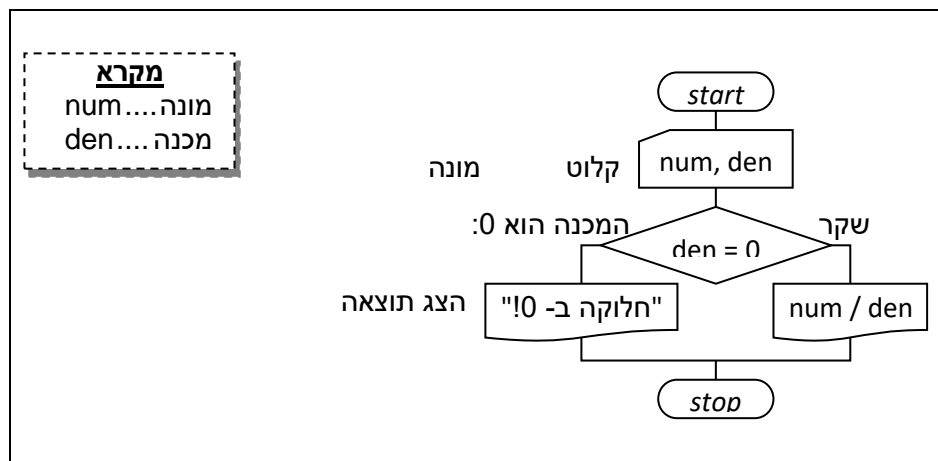
4. לולאות

כללי

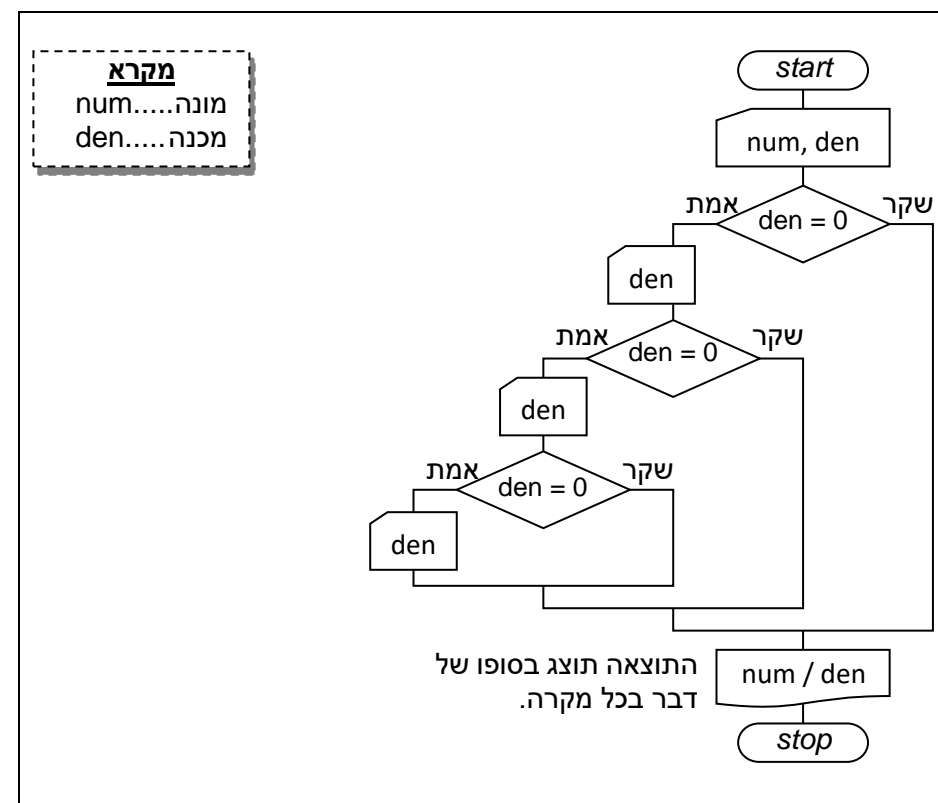
נתונה בעיה:

קלוט מונה ומכנה והצג את התוצאה (מונה חלקי מכנה). המנע מחלוקה ב-0.

הפתרון הבא הוא נכון:



לעיתים, נרצה שפעולה מסוימת תתבצע בתוכנית שלנו מספר פעמים, לדוגמא, עד שתנאי מסוים מתקיים. נסבך את הבעיה:



קלוט מונה ומכנה.

כל עוד המכנה לא תקין קלוט מכנה חדש. כאשר המכנה תקין הצג את התוצאה.

הדרישה הפעם היא שתוצג תוצאה בכל מקרה. אבל מספר הפעמים שנוכל לקלוט מכנה חדש מוגבל!

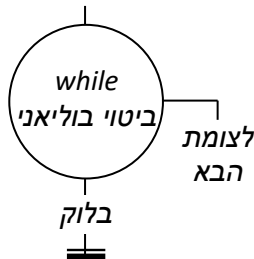
התרשים המסורבל הבא מתמודד, רק אם יוכנס מכנה לא תקין פחות מ-4 פעמים:

תרגיל זה מדגים את העיקרון של הלולאות בצורה הטובה ביותר. אין אנו יודעים כמה פעמים המשתמש יכניס את הערך 0 לחלוקה. אולי זה יהיה פעמיים, אולי זה יהיה 200 פעמים. לולאות מאפשרות לנו את החופש לבצע קטע מסוים שוב ושוב עד אשר תנאי מסוים מולא.

4.1 לולאת תנאי

לולאת תנאי מאפשרת ביצוע חוזר של בלוק.

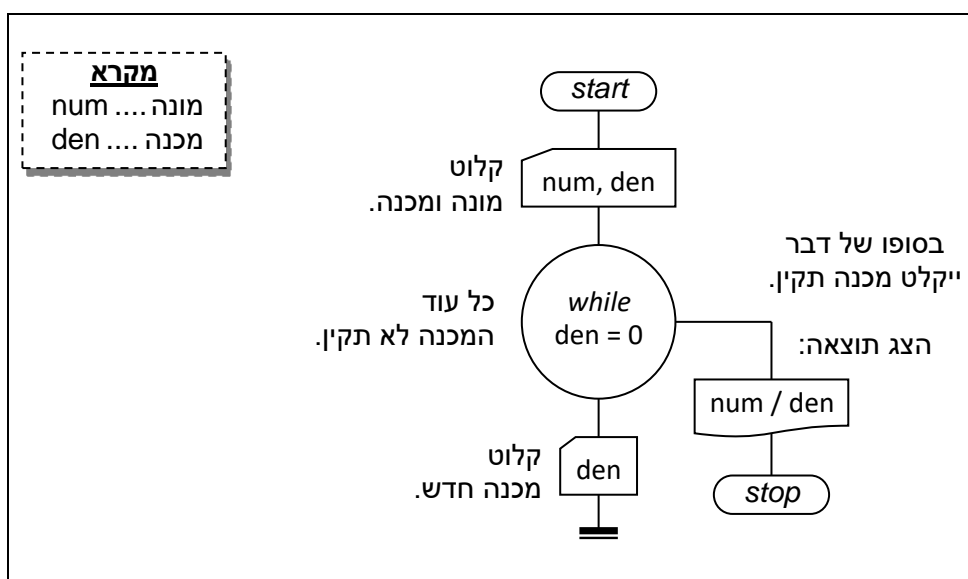
תחביר צומת לולאת תנאי



מעקב אחר לולאת תנאי

כשמגיעים לצומת לולאת תנאי בודקים האם הביטוי הבוליאני אמת. אם כן, מבצעים את הבלוק. לאחר ביצוע הבלוק חוזרים לצומת הלולאה. כך ממשיכים עד שהביטוי הבוליאני שקר – והלולאה מסתיימת. כאשר הלולאה מסתיימת התוכנית ממשיכה. כלומר הקוד ממשיך ימינה מהעיגול לצומת הבאה וכך הלאה עד שהוא מגיע לצומת ה- stop.

ביצוע יחיד של בלוק הלולאה נקרא איטרציה.



קלט מונה ומכנה.
כל עוד המכנה לא תקין
קלט מכנה חדש.
כאשר המכנה תקין הצג את התוצאה.

נעקוב אחר ביצוע התרשים עבור תור הקלט: 18, 0, 0, 6 →

נעשה שימוש בטבלת הרצה יבשה.

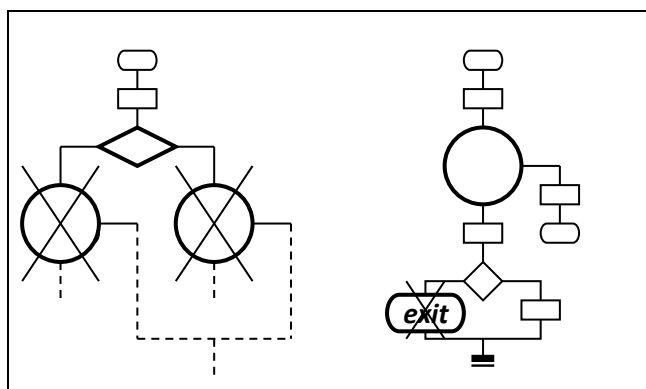
משמעות	num	Den	הדפסה למסך
קלוט מספר ראשון מתור הקלט	18		
קלוט מספר שני	18	0	
בדוק האם $den=0$, כן. היכנס ללולאה			
קלוט מספר מתור הקלט לתוך - den	18	0	
בדוק האם $den=0$, כן. היכנס ללולאה			
קלוט מספר מתור הקלט לתוך - den	18	6	
בדוק האם $den=0$, לא. צא מהלולאה.			
הצג פלט: תוצאת חישוב של 18 חלקי 6 (num/den)			3

שאלות:

1. * כמה פעמים בדקנו את תנאי הלולאה ?
2. * כמה פעמים ביצענו את בלוק הלולאה ?
3. * כמה איטרציות היו ?
4. * תן דוגמא לרשימת קלט עבורה לא תתבצע אף איטרציה.



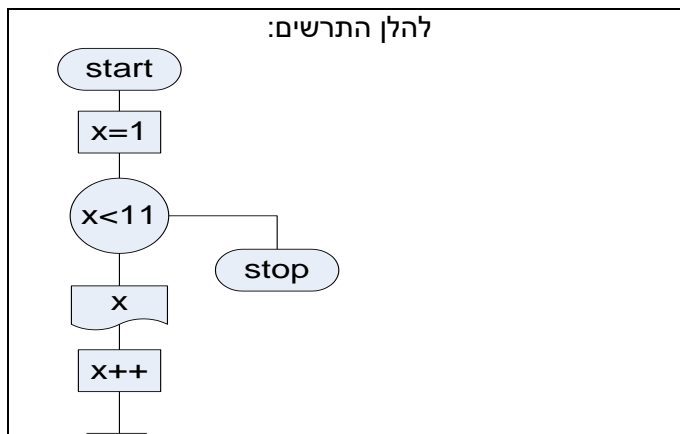
- צומת לולאה גורם לתרשים "לשבור" ימינה.
- צומת ה-*stop* יופיע מימין לצומת הלולאה ולא בדיוק מתחתיו.
- בסוף כל איטרציה חוזרים ישר לצומת הלולאה לבדוק את הביטוי הבוליאני.



- אסור לסיים תרשים עם צומת *exit* בתוך בלוק לולאה.
- ברוב המקרים אין לקונן צומת לולאה בבלוק של צומת תנאי. יש לנצל את התנאי של הלולאה כדי להימנע ממצב כזה.
- ישנם מקרים בהם לא ניתן להימנע ממצב כזה ובמקרים אלו ניתן לעשות זאת:

4.2 הרצה יבשה בטבלה

לולאות מביאות איתן גמישות רבה לתוכנית, ואיתה מתחיל קושי במעקב אחרי התקדמות התוכנית "בעל פה". במהרה תגלו כי כל תוכנית מצריכה מחשבה מקדימה. המחשב הוא כלי נפלא – הוא יודע לעשות את מה שאומרים לו במהירות מדהימה. עם זאת, הוא יעשה כל מה שנאמר לו ולא דבר אחד יותר מזה. אין הוא יודע "לעגל פינות" או לשער למה התכוונו (היזכרו בהגדרת האלגוריתם). הרצה יבשה היא מעקב אחרי התקדמות התוכנית שלנו צעד אחר צעד. באיטיות רבה ותוך שליטה מלאה בתוצאה.



תרגיל לדוגמא:
צרו לולאה שמדפיסה את המספרים בין 1 ל-10 (כולל).
לצורך כך, נעזר במשתנה עזר x ונאתחל אותו לספרה 1.
הלולאה תדפיס אותו ותעלה את המספר ב-1.
כך הלאה, עד שנגיע למספר 11. זה יהיה תנאי היציאה שלנו מן הלולאה.

את הריצה היבשה נתאר בצורה הבא:
נרשום את כל המשתנים שקיימים בלולאה (במקרה הזה x).
מתחתיו נרשום צעד צעד את התקדמות התוכנית.
במקרה של כתיבה על המסך, מומלץ גם לצייר ריבוע שמייצג את המסך, ובכל פעם שמתבצעת הורשה של הדפסה למסך, נבצע אותה על הנייר.
להלן התרשים בכמה צעדים:

משמעות	x
אתחל את x ל-1	1
בדוק האם $x < 11$? כן. הדפס את x.	
קדם את x ב-1	2
בדוק האם $x < 11$? כן. הדפס את x.	
קדם את x ב-1	3
בדוק האם $x < 11$? כן. הדפס את x.	
... נדלג כמה שלבים. כעת $x=9$	9...
בדוק האם $x < 11$? כן. הדפס את x.	
קדם את x ב-1	10
בדוק האם $x < 11$? כן. הדפס את x.	
קדם את x ב-1	11
בדוק האם $x < 11$? לא. צא מהלולאה.	

מסך המחשב

12345678910

מומלץ, גם כשמתחילים לתכנת על המחשב, לא לגעת במקשי המקלדת לפני שציירו ו"הרצנו" על הנייר את התוכנית ויש לנו מושג לאיזה כיוון אנחנו עומדים לתכנת.
אם תור הקלט לא הוגדר, ולרוב הוא לא מוגדר מראש, עליכם להגדיר אילו נתונים אתם רוצים להריץ על האלגוריתם שלכם.
התחילו עם משהו פשוט ובנאלי. אחרי שבדקתם ונראה שזה עובד, נסו להכשיל את האלגוריתם שבניתם. קחו מקרי קצה (שווה ל-0, מספר שלילי, מספר גדול מאוד או מספר קטן מאוד וכו...) בהתאם לתוכנית.



1. * קלוט מספר טבעי (שלם חיובי) top .
הצג את כל המספרים הטבעיים מ-1 עד top (כולל).
2. קלוט שני שלמים והצג את כל השלמים ביניהם (כולל) בסדר עולה.
לא מובטח שהנתון השני גדול מהראשון.
3. קלוט מספר טבעי n .
הצג את כל המספרים הזוגיים מ-0 עד n (כולל). לא מובטח ש- n זוגי.
4. * קלוט שני מספרים טבעיים max ו- den .
הצג את כל המספרים הטבעיים עד max (כולל) המתחלקים ב- den .
לא מובטח ש- max עצמו מתחלק ב- den .

עיבוד נתונים

5. * קלוט מספרים אי-שליליים עד הזקיף 99- והצג את סכומם.
6. בתור הקלט כמות לא ידועה של מספרים חיוביים שבסופם המספר 0.
הצג את ממוצע כל המספרים לא כולל ה-0 (יתכן שהנתון הראשון הוא 0).
7. קלוט מספרים עד שתקלוט מספר שלילי או 0.
הצג את הערך הגבוה ביותר שנקלט (יתכן שהנתון הראשון אינו חיובי).
8. * קלוט מספרים עד שתקלוט מספר שלילי או 0.
הצג את הערך החיובי הנמוך ביותר (יתכן שהנתון הראשון אינו חיובי).
9. קלוט 100 מספרים.
הצג את המספר הסיידורי של הערך הגבוה ביותר.

ניתוח שלם

10. קלוט שלם והצג את הספרה השמאלית ביותר שלו.
11. * קלוט שלם והצג את מספר הספרות שלו.
12. קלוט שלם והצג את סכום הספרות שלו.
13. * קלוט שלם X ושלם חד ספרתי dig .
הצג כמה פעמים הספרה dig מופיעה ב- X .
14. * קלוט שלם והצג את ספרותיו בסדר הפוך (עבור הקלט 1304 יוצג 4031. עבור 123- יוצג -321).
15. פולינדרום הוא מספר סימטרי, כלומר ערכו זהה גם כשרושמים את ספרותיו בסדר הפוך.
קלוט שלם, והצג "פולינדרום" או "לא פולינדרום" בהתאם.

חשבון

16. קלוט שני שלמים אי-שליליים.
הצג את מכפלתם ללא שימוש בפועל הכפל (*).
17. * קלוט שני שלמים אי-שליליים.
הצג את הראשון בחזקת השני ללא שימוש בפועל החזקה (**).
18. קלוט שני שלמים אי-שליליים.
הצג את מנתם השלמה ואת השארית ללא שימוש בפועלי החילוק (/) והמודולו (%).
19. עצרת של מספר טבעי n היא מכפלת כל השלמים בין 1 לבין n (כולל).
למשל: העצרת של 5 היא $1 \times 2 \times 3 \times 4 \times 5 = 120$
העצרת של 2 היא $1 \times 2 = 2$
קלוט מספר טבעי n והצג את העצרת שלו.
20. * קלוט מספר. הצג את השורש הריבועי שלו בדיוק של 1% ללא שימוש בפועל החזקה (**).
דיוק של 1% – אם מחפשים לדוגמא את השורש של 1000, מספיק למצוא שורש של מספר בין 990 לבין 1010.

סדרת פיבונצ'י

- סדרת פיבונצ'י מוגדרת באופן הבא:
האיבר הראשון שווה 1.
האיבר השני שווה הוא גם 1.
כל שאר האיברים שווים לסכום שני האיברים הקודמים.
כלומר, הסדרה מתחילה כך: $1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, \dots$
21. * קלוט מספר טבעי $index$ והצג את האיבר ה- $index$ בסדרה.
(מובטח ש- $index > 2$)
22. קלוט מספר טבעי $value$.
הצג את סידרת פיבונצ'י עד האיבר הראשון הגדול מ- $value$ (כולל).

מתמטיקה

23. * קלוט מספר טבעי והצג את כל המחלקים השלמים שלו.
למשל: המחלקים של 60 הם 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60.
המחלקים של 61 הם 1, 61.
24. * קלוט שני מספרים טבעיים והצג את המחלק המשותף הגדול ביותר.
למשל: המחלק הגדול ביותר של 60 ושל 72 הוא: 12.
המחלק הגדול ביותר של 64 ושל 81 הוא: 1.
25. * קלוט מספר והצג את פירוקו לגורמים ראשוניים.
למשל: 60 מתפרק ל- $2 \times 2 \times 3 \times 5$.
59 מתפרק ל- 1×59 (59 עצמו הוא ראשוני).

מספר לולאות

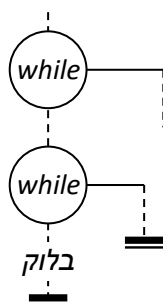
26. * קלוט שני שלמים.
הצג את כל השלמים שביניהם (כולל) מהראשון שנקלט ועד השני שנקלט.
27. קלוט שני שלמים.
הצג את כל השלמים שביניהם (כולל) מהקטן לגדול ובחזרה לקטן.
28. בתור הקלט רשימת טמפרטורות היומיות של יולי (30 ימים) 2000 ו- 2001 (סה"כ 60 מספרים).
חשב את הטמפרטורה הממוצעת ביולי 2000.
הצג את הימים ביולי 2001 בהם החום היה גבוה מהממוצע של יולי 2000.

4.3 קינון לולאת תנאי

ניתן לקונן לולאת תנאי בבלוק של לולאה אחרת. אפשרות זו מקנה גמישות רבה.

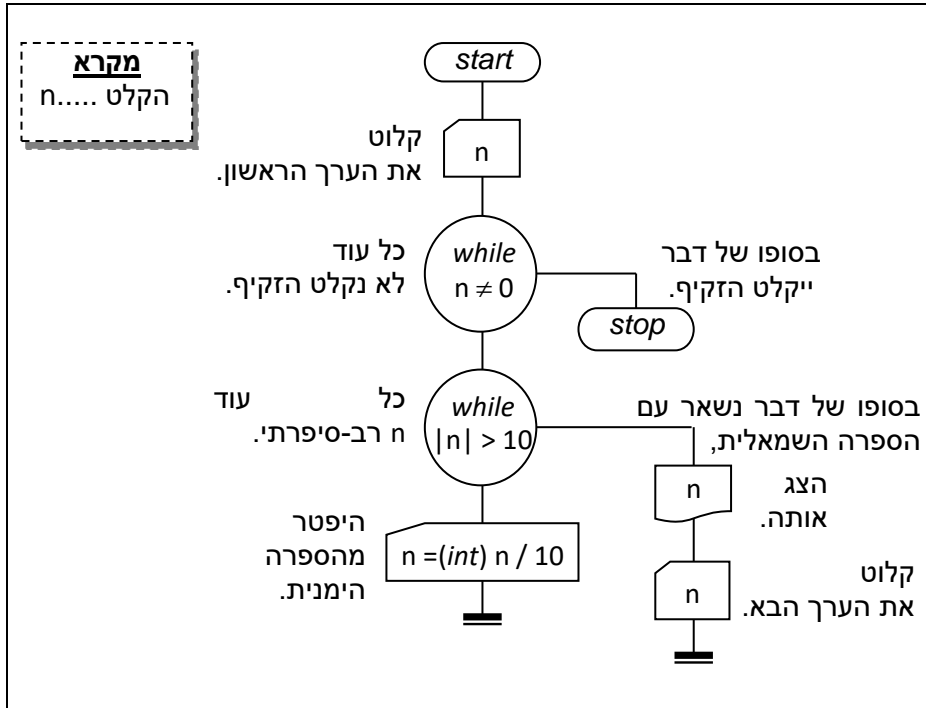
למדנו כי לולאה אחת עושה פעולה מסוימת שוב ושוב. אם הפעולה שלה היא בעצם לולאה אחרת, היא תבצע את הלולאה הפנימית שוב ושוב. כל פעם שהלולאה הפנימית מתבצעת קוראת פעולה נוספת שוב ושוב. נחבר את שני הדברים הללו – קיבלנו לולאה שמבצעת לולאה נוספת מספר פעמים. מספר הפעמים שהקוד יכנס ללולאה הפנימית הוא מספר האיטרציות של הלולאה הפנימית **כפול** מספר האיטרציות של הלולאה החיצונית.

תחביר



- כל בעיה שניתן לפתור עם לולאה מקוננת, ניתן לפתור גם עם לולאה בודדת. היתרון של לולאה מקוננת: בהירות.
- ברוב המקרים אנו לא נקונן יותר מלולאה אחת בתוך השנייה (סה"כ 2 לולאות), אולם לעיתים קיים צורך כזה. את הפתרון לצורך ביותר משני לולאות מקוננות תלמדו בפרק "פונקציות".





קלט שלמים עד שייקלט 0.
הצג את הספרה השמאלית
ביותר של כל ערך.

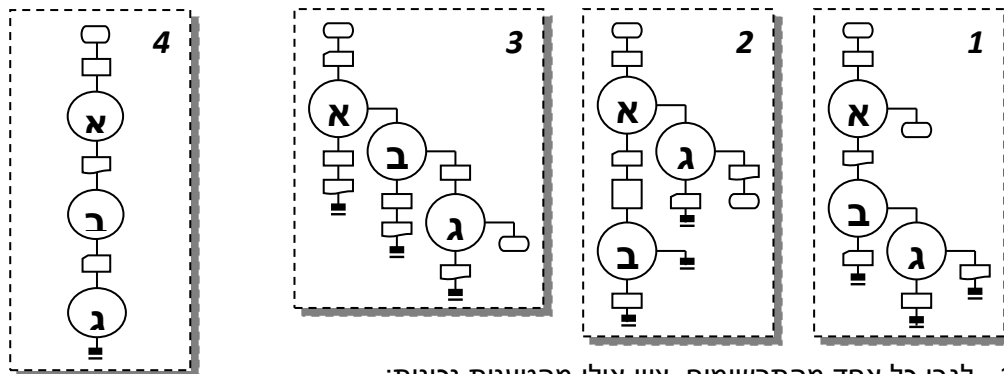
תרגום האלגוריתם לעברית –
קלט מספר.
כל עוד המספר שונה 0, בדוק
אם הערך המוחלט שלו גדול
מ-10.

אם כן, המשיך בחילוק שלמים
במספר 10.

אם לא, כלומר n כרגע מכיל
מספר בעל ספרה בודדה, הצג
את n כפלט.
קלט n חדש וחזור לבדיקה
האם n שונה מ-0.

המשך התוכנית לא יוצא מהלולאה הפנימית. הלולאה הפנימית היא חלק מהתהליך הפנימי של הלולאה החיצונית בעוד
שהלולאה החיצונית היא היחידה שיכולה להוביל להמשך התוכנית. כמובן שבמקרה זה המשך התוכנית הוא צומת stop.

תרגילים



1. לגבי כל אחד מהתרשימים, ציין אילו מהטענות נכונות:

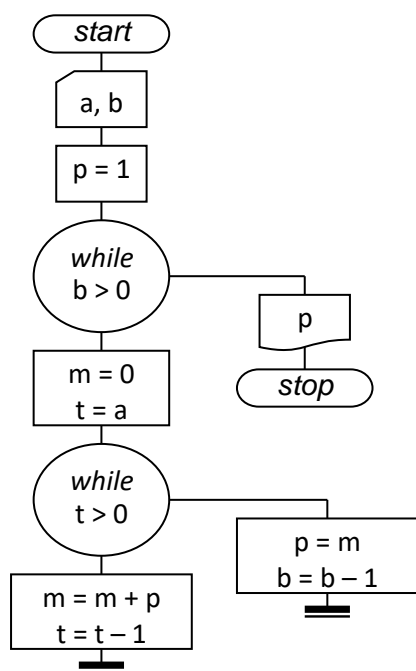
א, ב, ג הן לולאות כאשר לכל לולאה מוגדרות כמות איטרציות(חזרות) הבאות:

א: 3

ב: 2

ג: 2

כמה פעמים יתבצע קטע הקוד שתחת כל לולאה?



*.2 מה יהיה הפלט של התרשים הבא עבור הקלט $\rightarrow 3, 4$
מה יהיה פלט התרשים עבור הקלט $4, 3 \rightarrow$?

3. קלוט שלמים עד שייקלט ערך שלילי.

עבור כל שלם חיובי הצג את סכום ספרותיו.

*.4 קלוט שלם והצג את ספרותיו ממוינות בסדר עולה.

למשל: קלט – 5001, פלט – 0015.

קלט – 48444, פלט – 44448.

*.5 קלוט שלמים עד שייקלט מספר ראשוני.

הצג את כמות השלמים הלא-ראשוניים שנקלטו.

6. קלוט שלמים עד שייקלט 0.

עבור כל שלם הצג האם הוא איבר בסדרת פיבונצ'י.

$\rightarrow 1, 1, 2, 3, 5, 8, 13, 21, \dots$

7. בתור הקלט רשימת דרגות החום היומיות לשנת 1984.

הנתון 999- הוא זקיף המסמן את סוף החודש.

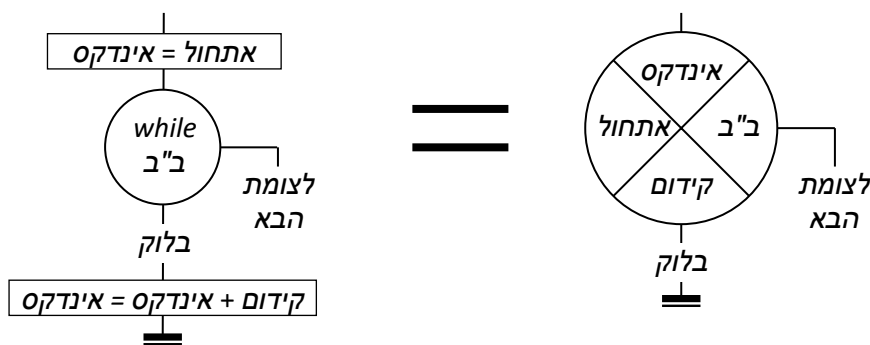
הצג את מספרי החודש החם ביותר והחודש הקר ביותר בשנת 1984.

4.4 לולאת אינדקס

לולאת אינדקס היא שכלול של לולאת תנאי. בנוסף לבדיקת התנאי, היא גם מאתחלת משתנה (תא בזיכרון המחשב) (אינדקס) לפני הבדיקה הראשונה ומקדמת אותו לפני כל בדיקה נוספת. נוח ומקובל להשתמש בלולאת אינדקס כאשר מספר האיטרציות (חזרות) ידוע.

תחביר צומת לולאת אינדקס

השווה בין התרשימים השקולים (לולאת אינדקס מימין ולולאת תנאי משמאל):



אינדקס | המשתנה שיאותחל ויקודם ע"י הלולאה.

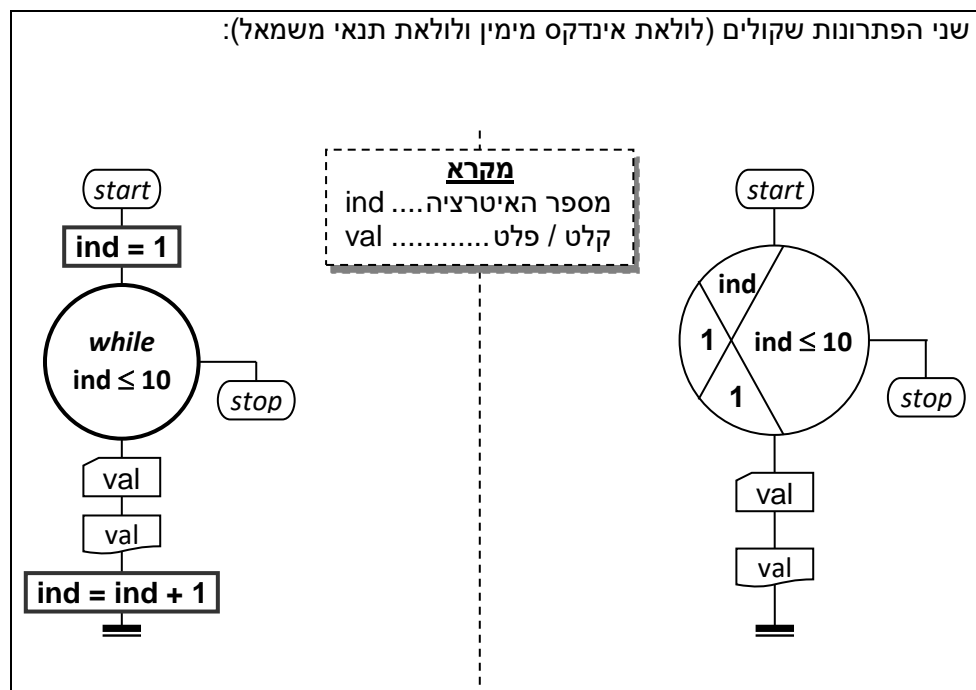
ב"ב | ביטוי בוליאני שהוא התנאי לביצוע איטרציה.

אתחול | ערך שהאינדקס יקבל לפני בדיקת התנאי בפעם הראשונה.

קידום | ערך בו יגדל האינדקס לפני בדיקת התנאי החל מהפעם השנייה.

שני הפתרונות שקולים (לולאת אינדקס מימין ולולאת תנאי משמאל):

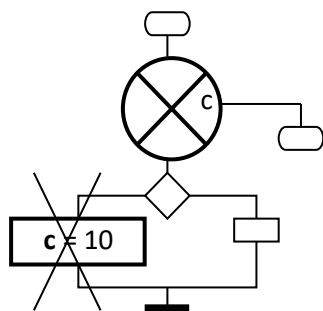
קלוט 10 מספרים והצג אותם.





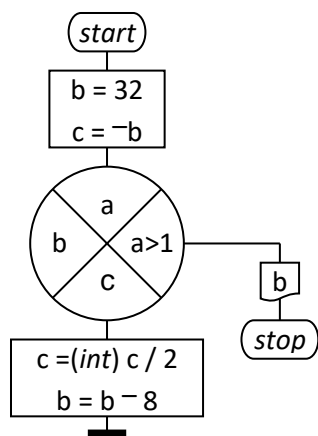
1. קלוט 100 נתונים, סכם אותם והצג את התוצאה.
2. בקלט מספר לא ידוע של נתונים. הנתון הראשון חיובי ושלם ומסמל את מספר הנתונים שאחריו. חשב את ממוצע נתונים אלו (ללא הראשון).
3. * קלוט 100 נתונים והדפס את הגדול מביניהם.
4. * הדפס את סידרת המספרים מ-1 ועד 100 (1, 2, 3, 100).

- כל הכללים וההגבלות לגבי לולאת תנאי חלים גם על לולאת אינדקס.
- אם משתמשים בלולאת תנאי ויוצרים אינדקס, יש לשקול להשתמש בלולאת אינדקס – לזה היא מיועדת.
- אסור לשנות אינדקס של לולאת אינדקס בבילוק הלולאה.



תרגילים

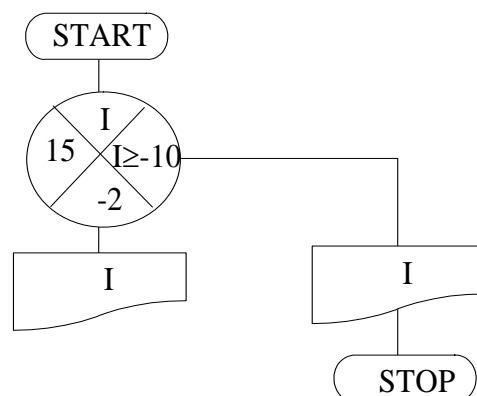
1. מה יהיה הפלט של התרשים?



2. קלוט שני שלמים. הצג את כל השלמים שביניהם (כולל) מהראשון שנקלט ועד השני שנקלט.
3. * קלוט נתון למשתנה N. חשב והדפס את N עצרת לפי הנוסחה: $S! = 1 * 2 * 3 * \dots * N$
4. * קלוט נתון למשתנה N. חשב והדפס את סכום המספרים מ-1 ועד N אשר מתחלקים ב-3 (לדוגמא: אם הנתון היה 7 אזי התוצאה תהיה 9 כי 3 ו-6 מתחלקים ל-3).

5. * קלוט 99 נתונים, סכם כל נתון שלישי (כלומר תצטרך לסכם את הנתון השלישי, שלישי, תשיעי...) הדפס את התוצאה.
6. * קלוט נתון למשתנה N. חשב את סכום הסדרה: $2^0 + 2^1 + 2^2 + \dots + 2^N$ והדפס את התוצאה. לדוגמא אם הנתון שקראת היה 3 אזי התוצאה שתודפס היא: 15
($2^0 + 2^1 + 2^2 + 2^3 = 1 + 2 + 4 + 8 = 15$)
7. * קלוט נתון N. הדפס שתי סדרות נפרדות.
א. הדפס את סדרת המספרים מ-1 עד N.
ב. בהמשך להדפסה הקודמת הדפס סידרה נוספת שבה כל מספר מהסדרה הקודמת מוכפל ב-2.
8. * קלוט נתון למשתנה N.
סכם את כל המספרים עד N המתחלקים ב-4 ואת כל המתחלקים ב-7, לסכום אחד. הדפס את התוצאה (כמה לולאות צריך לפתרון השאלה?)
9. * ידוע כי $Y = X^2 + 5$. עליך לחשב ולהדפיס את ערכי Y עבור ערכים שונים של X. קלוט:
א. ערך X התחלתי.
ב. ערך X סופי.
ג. ההפרש בין כל ערך של X בתחום שנקלט (סעיפים א' ו-ב').
לדוגמא: אם נקלוט 2, 13, 5
** מובטח כי הערך ההתחלתי קטן מהערך הסופי וההפרש הוא חיובי.
עליך לחשב את Y עבור סדרת X שמתחילה ב-5 ומתקדמת עד 13 בקפיצות של 2.
($Y = 30, 54, 86, 126, 174$).

10. * מה יודפס ?



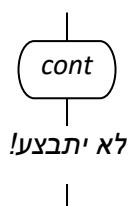
11. * קלוט שני שלמים.
הצג את כל השלמים שביניהם (כולל) מהקטן לגדול ובחזרה לקטן.
12. * קלוט 100 שלמים.
הצג את סכום כל הזוגיים.
13. * קלוט 100 נתונים.
הצג את סכום כל הנתונים שמספרם הסידורי זוגי (השני, הרביעי וכו'...).

-
14. * קלוט נתון למשתנה A. לאחר מכן קרא 100 נתונים והדפס את מספרם הסידורי של אלה ששווים ל - A. (לדוגמא: אם התוצאה שתודפס תהיה 90,23,1 המשמעות היא שהנתון באינדקס 1, באינדקס 23 ואינדקס 90 מתוך 100 הנתונים שנקראו שווים ל - A).
15. * קלוט 100 זוגות של כרטיסים (סה"כ 200 כרטיסים), בכל זוג, הנתון הראשון מציין שם של תלמיד והנתון השני מציין ציון. הדפס את שמות התלמידים אשר הצליחו בבחינה (ציון גבוה מ- 70) וכן את ממוצע הציונים של המצליחים. (כלומר - בחשוב הממוצע אין להתחשב בציוניהם של הנכשלים) שים לב למקרה בו כולם נכשלו.
16. * קלוט 50 נתונים וסכם אותם. לאחר מכן קלוט עוד 1000 נתונים והדפס את מספרם הסידורי (1 - 1000) של אלו השווים לסכום 50 הראשונים.
17. קלוט מספר טבעי count, ועוד count נתונים. הצג את הערך הגדול ביותר בתוך count הנתונים, המספר הסידורי של המופע הראשון של אותו ערך בקלט ואת מספר המופעים הנוספים של אותו הערך.
18. * קלוט מספר טבעי amount, ועוד amount נתונים. הצג את הערך השני בגודלו מבין amount הנתונים ואת המספר הסידורי של המופע האחרון של אותו ערך בקלט.
19. סדרת פיבונצ'י היא הסדרה: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89... → קלוט מספר טבעי n, והצג את האיבר הראשון בסדרה הגדול מ- n.
-

4.5 סיום מוקדם של איטרציה ושל לולאה

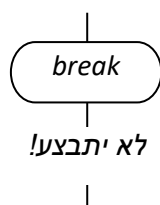
ניתן לדלג מתוך בלוק הלולאה חזרה לתנאי הלולאה.
ניתן לדלג מתוך בלוק הלולאה לצומת הבא מחוץ ללולאה.

תחביר צומת *cont*



צומת *cont* מפנה לצומת של הלולאה הפנימית ביותר בה הוא נמצא. למעשה, צומת *cont* מפסיקה את האיטרציה הנוכחית. לדוגמא, שימוש בצומת *cont* בלולאת אינדקס יגרום לקידום ערכו של המשתנה ולאחר מכן לבדיקה של הביטוי הבוליאני.

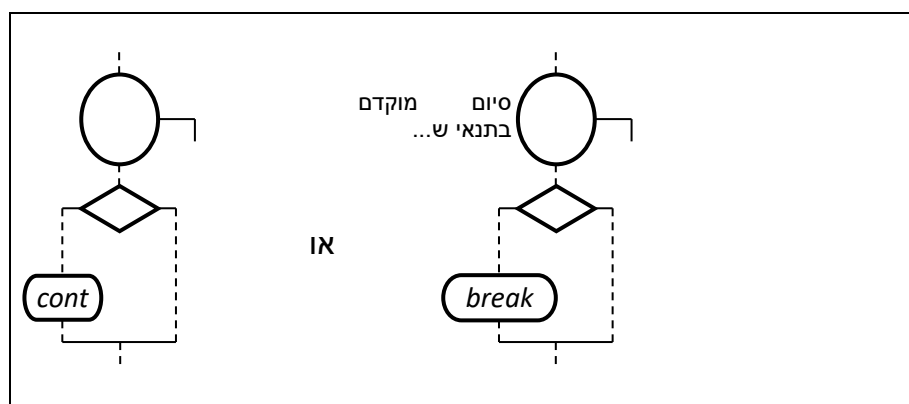
תחביר צומת *break*



צומת *break* מסיים את פעולת הלולאה בה הוא נמצא.
אם זו לולאה מקוננת, הוא יסיים את הלולאה הפנימית, אך ימשיך לבצע את החיצונית.

- מומלץ לא להשתמש ב-*cont* ו-*break*, אם יש אפשרות לגרום ללולאה לסיים את האיטרציות בעצמה.
- מכיוון שפקודות מתחת לצומת *cont* ומתחת לצומת *break* לא יתבצעו, נציב את הצמתים בסוף בלוק מותנה בתוך בלוק של לולאה.

 לדוגמא:





1. * קרא 100 נתונים ובדוק אם הם ממוינים בסדר עולה (כלומר, אם כל נתון הוא גדול או שווה לקודמו). אם כן, הדפס: "ממוין", אחרת, הפסק והדפס: "לא ממוין".

2. נתונים 1,000 כרטיסים. קלוט אותם בזה אחר זה וסכם את סכום הנתונים, כאשר הגעת לנתון השווה לסכום כל קודמיו הפסק והדפס אותו. אם לא מצאת כזה הדפס "לא נמצא".

3. * נתונים 365 כרטיסים. המכילים את הטמפרטורה הממוצעת באזור ת"א בכל יום מימות שנת 1994, ידוע כי הטמפרטורה בת"א לעולם אינה עולה על 40C ואינה יורדת מ-5C. קלוט את הכרטיסים ובדוק אם לא נפל כרטיס שגוי בקלט, במידה שכן הפסק והדפס "נתונים שגויים" במידה שלא הדפס "נתונים תקינים".

4. בעצרת האו"ם 170 מדינות.
בהצבעה במליאה ניתן להצביע ב- 4 אופנים שונים לפי הקודים הבאים:

- 1 - בעד
- 2 - נגד
- 3 - נמנע
- 4 - וטו

קלוט 170 קודים מהצבעה מסוימת, הדפס את מס' הקולות בעד, מס' הקולות נגד, ואת מס' הנמנעים.
במידה שהגעת להצבעת וטו, הדפס רק את המספר הסידורי של המדינה שהטילה וטו, ועצור את התוכנית.

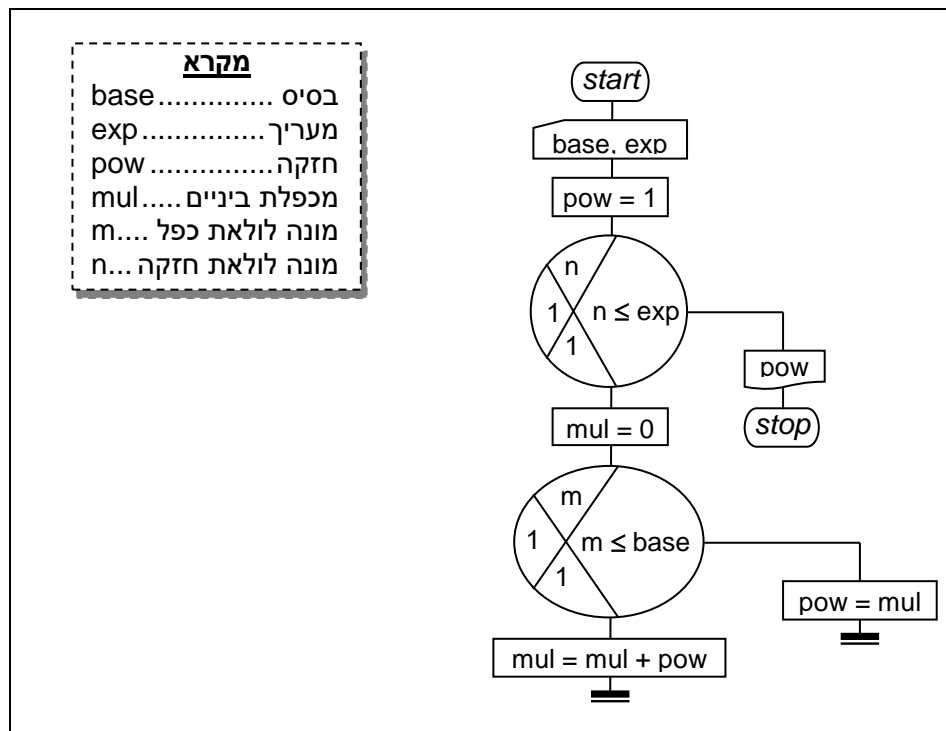
5. * קלוט 1000 נתונים.
סכם לתוך משתנה SUM1 את אלו הקטנים מ-1,000.
סכם לתוך משתנה SUM2 את אלו הקטנים מ-2,000 וגדולים או שווים ל-1,000.
סכם למשתנה SUM3 את השאר.
הדפס את שלושת משתני הסיכום.
אין להשתמש בתנאי בתוך תנאי.

4.6 קינון לולאת אינדקס

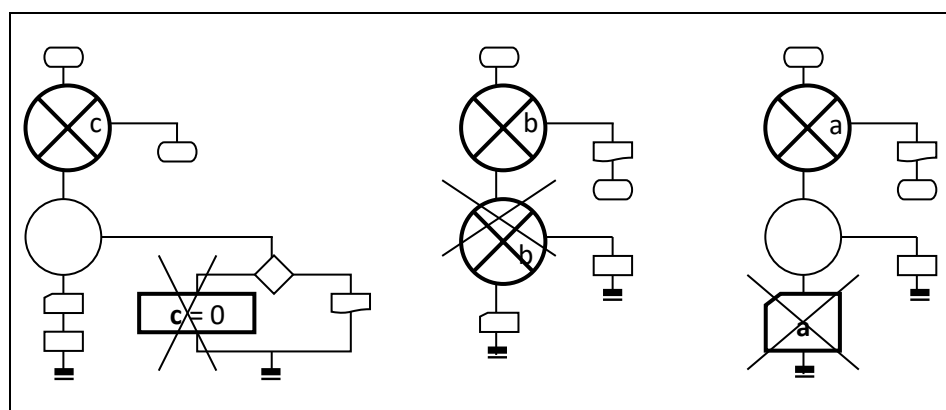
לעיתים, ניתן לכתוב מספר לולאות אחת בתוך השנייה. פעולה זו נקראת **קינון לולאות**. נרצה לבצע קינון לולאות, כאשר נרצה לחזור פעמים על סדרת פעולות שמתבצעת ע"י לולאה. במקרה זה, נכתוב את סדרת הפעולות בלולאה אחת ולאחר מכן נמקם את הלולאה בתוך לולאה שנייה.

לדוגמא:

קלוט שני מספרים טבעיים
(חיוביים שלמים).
הצג את הראשון בחזקת
השני ללא שימוש בפועלים *
ו- **.



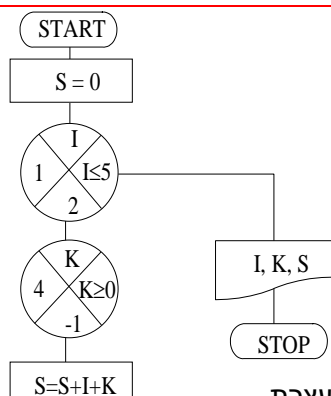
להזכירך, אסור לשנות
אינדקס של לולאת אינדקס
בבלוק הלולאה.



תרגילים



1. הדפס ריצה יבשה של התרשים הבא:



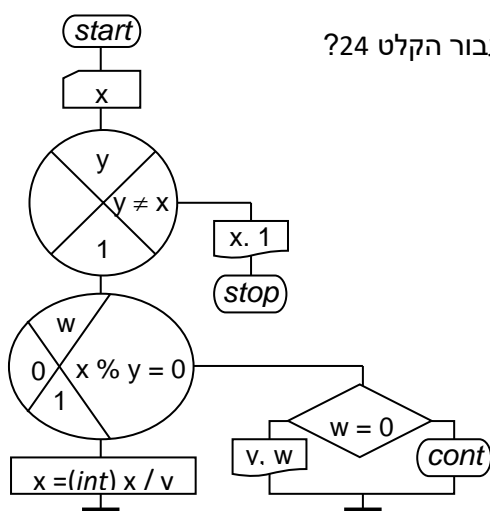
2. * קלוט 100 נתונים והדפס עבור כל אחד מהם את העצרת.

3. * בבי"ס מסוים יש 100 תלמידים ולכל תלמיד 10 ציונים. הנתונים מסודרים בקלט לפי תלמידים דהיינו - 10 ציוני תלמיד ראשון, אח"כ 10 ציוני תלמיד שני וכו'. הדפס את ממוצע ציוניו של כל תלמיד וכן את ממוצע בי"ס (ממוצע ביה"ס היא ממוצע כל הציונים שנקלטו).

4. * קלוט מספר N. הדפס את כל המספרים מ-1 עד N, ולאחר מכן את כל המספרים מ-2 עד N וכן הלאה, כשבכל פעם מודפס טור מספרים הקטן באיבר אחד מקודמו, עד שיודפס N לבדו.
לדוגמא: N=4 יודפס 1,2,3,4 (פעם ראשונה)
2,3,4 (פעם שניה)
3,4 (פעם שלישית)
4 (פעם רביעית)

5. כמו בשאלה הקודמת רק שהפעם תודפס סדרה כזו:
אם N=4 1,2,3,4 (פעם ראשונה)
1,2,3 (פעם שניה)
1,2 (פעם שלישית)
1 (פעם רביעית)

6. * מה יהיה הפלט של התרשים הבא עבור הקלט 24?
מה יהיה פלט התרשים עבור הקלט 23?



7. קלוט עשרה שלמים. הצג את כל השלמים בין כל זוג שלמים שנקלטו.
למשל: קלט – 9, 12, 8, 8...
פלט – 9, 10, 11, 12, 11, 10, 9, 8, 8...

8. קלוט מספר שורות ומספר עמודות והצג מלבן כוכביות.

למשל: קלט – 3, 7 →

```

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

9. * הדפס את טבלת לוח הכפל בגודל 20 X 20 סדר הדפסה יהיה לפי השורות, כלומר:

```

1,2,3.....20
2,4,6.....40
3,6,9.....60
20,40,60.....400

```

(יש להתעלם מהצורה החיצונית של לוח הכפל ולהתייחס לערכים שצריך להדפיס).

10. קלוט מספר שורות ומספר עמודות והצג את לוח הכפל.

11. * קלוט 1000 נתונים. סכם כל נתון עשירי החל מהנתון הראשון (כלומר: הראשון, האחד עשר, העשרים ואחד וכו'). שים לב: אין לקרוא 991 נתונים או 1001 נתונים.

12. * במפעל 200 עובדים.

בסוף השנה מעוניין המנהל לדעת מי מהעובדים קיבל במהלך השנה את המשכורת הגבוהה ביותר, ובאיזה חודש נתנה משכורת זו. (הנח כי כל המשכורות חיוביות)
קלוט: שם העובד ולאחריו 12 המשכורות החודשיות שנתנו לו במשך השנה כשהן ממוינות לפי חודש.

הדפס: שם עובד שמשכורתו הייתה הגבוהה ביותר, המשכורת הגבוהה ביותר והחודש בו ניתנה אותה המשכורת.

13. קלוט שלם והצג את ספרותיו ממוינות בסדר יורד.

למשל: קלט – 4020, פלט – 0042.

קלט – 73771, פלט – 77731.

14. * קלוט 100 שלמים. הצג את כמות השלמים שספרותיהם ממוינות בסדר יורד.

15. בתור הקלט כמויות משקעים חודשיות לעשור 1991 עד 2000. הצג את השנה הגשומה

בעשור ואת החודש הגשום בעשור. הפלט עשוי להיות:

Rainiest year was 1997

Rainiest month was 12 of 1995

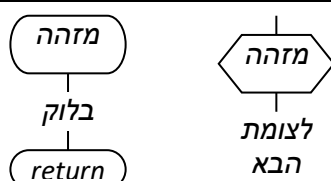
5. פונקציות

5.1 פונקציה

פונקציה היא אלגוריתם המהווה חלק מאלגוריתם אחר. ראו בפונקציה פעולה מוגדרת מראש הניתנת לשימוש שוב ושוב. במהלך המודול, פתרונם תרגילים שהשתמשו בתרגילים קודמים, ע"י העתקת התרגיל הקודם והוספת דברים. עם פונקציות ניתן להגדיר לדוגמא, פתרון של תרגיל בפונקציה ולהשתמש בה אח"כ שוב ושוב. נשתמש בפונקציה כדי:

- לרשום אלגוריתם עם פעולות שנוחות לנו ולא רק עם הפעולות הקיימות.
- לרשום בלוק פעם אחת ולזמן אותו שוב ושוב.

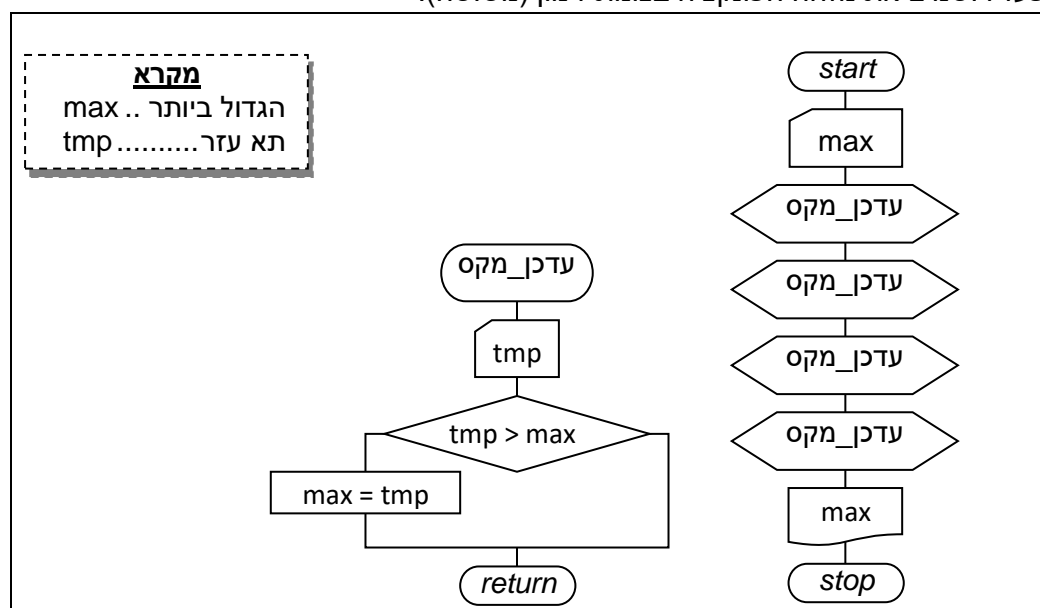
תחביר צומת זימון ותרשים פונקציה



לכל פונקציה מזהה ייחודי.

תרשים פונקציה זהה לתרשים ראשי למעט שני הבדלים:
צומת כותרת – במקום *start* רושמים את מזהה הפונקציה.
צומת return – במקום *stop* רושמים *return*.

קריאה לפונקציה תבצע ע"י שימוש במזהה של הפונקציה בתוך משושה. אותו משושה מהווה העברת אחריות לפונקציה ועצירת התוכנית הראשית, עד שזו תסיים את הפעולות הפנימיות שלה ותחזיר את האחריות לתוכנית הראשית, להמשך האלגוריתם. כשרוצים שהפונקציה תופעל רושמים את מזהה הפונקציה בצומת זימון (משושה).



לדוגמא:
קלוט חמישה מספרים והצג את הגדול מביניהם.

פתרון יכול להיות מורכב מכמה תרשימים. תרשים יחיד הוא הראשי, לו צומת *start* יחיד ממנו מתחילים וצומת *stop* יחיד בו מסיימים. שאר התרשימים הם פונקציות. לכל אחד מהם צומת כותרת אחד וצומת *return* אחד. ניתן לפתור את התרגילים ללא פונקציות, אם כי זה יהיה הופך את התרגיל לארוך ומסורבל. זאת, במיוחד אם יש צורך לבצע קטע קוד מספר פעמים.

שים לב, זהו הפתרון לבעיית הלולאות המקוננות מפרק הלולאות. אם יש צורך ביותר מלולאה מקוננת אחת אנחנו פותרים זאת ע"י קריאה לפונקציה. זה הופך את הקוד לקריא וקל יותר למעקב ע"י ריצה יבשה (נסו לבצע ריצה יבשה ל-3 לולאות מקוננות... מה עם 4? יש הרבה מקום לטעויות). נסו להימנע בינתיים מלולאות מקוננות ואף לקריאה לפונקציה במקום לולאות מקוננות, נושא זה לא קשה במיוחד אך הוא מעבר לטווח הלימוד של מודול זה. יהיה מקום רב להתנסות בכלי זה כאשר תעברו לעבוד מול המחשב.

5.2 פרמטרים

פרמטר של פונקציה הוא משתנה הדרוש לביצועה. פונקציה **המקבלת** פרמטר היא כללית יותר. מנסחים את ההוראות עבור משתנה רשמי כלשהו (**פרמטר פורמלי**) ובעת הקריאה לפונקציה מציינים לאיזה משתנה יש להתייחס בפועל (**פרמטר אקטואלי**).

בקורס זה הפרמטר האקטואלי הוא תמיד התייחסות לתא קיים (by reference). פרמטר אקטואלי בשפת #C, למשל, הוא תא חדש המקבל ערך של ביטוי (by value).



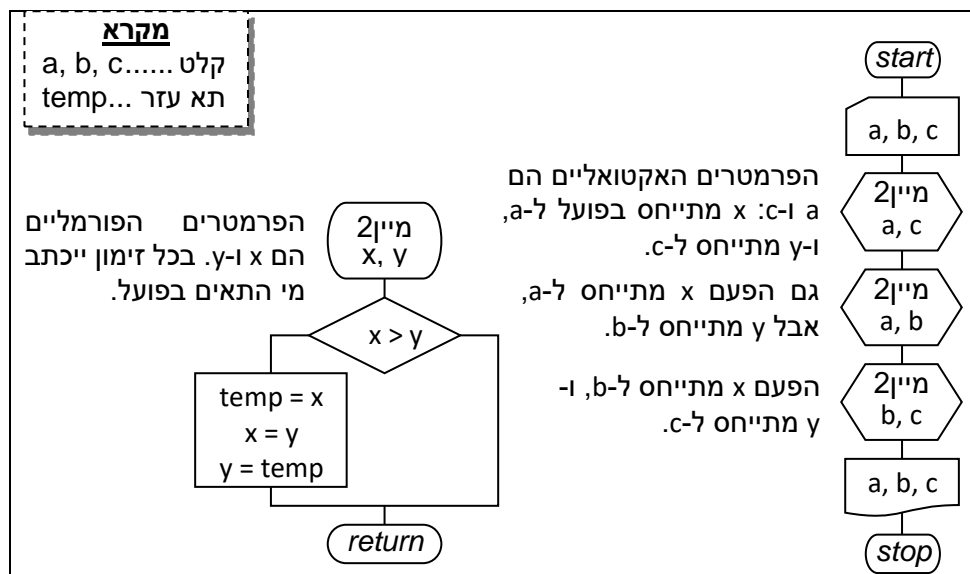
במילים אחרות, הפרמטר הפורמאלי הוא שם משתנה, אשר מכריזים עליו בעת כתיבת הפונקציה לצורכי כתיבתה. למעשה, הסיבה שמכריזים עליו היא, בכדי שבקוד הפונקציה נוכל להתייחס לערכו (שכרגע אינו ידוע) ולבצע עליו פעולות שונות.

הפרמטר האקטואלי, הוא אותו משתנה שנשלח לפונקציה מהתוכנית הראשית. למעשה, ערכו של הפרמטר האקטואלי נקבע במהלך ריצת התוכנית, וכאשר מעבירים אותו לפונקציה, הפרמטר הפורמלי יתייחס לערכו.

פונקציה יכולה לקבל פרמטר אחד או יותר, או לא לקבל פרמטרים כלל. למרות זאת, כאשר הפונקציה הוכרזה עם מספר פרמטרים, יש להעביר אליה את מספר הפרמטרים שהוגדרו בהכרזה, משמע, כלל הפרמטרים.

לדוגמא:

קלוט שלושה מספרים והצג אותם ממוינים בסדר עולה.



תרגילים

- * קלוט זמן (שעות, דקות ושניות).
הצג את הזמן שנקלט בפורמט של שעות דיגיטלי hh:mm:ss.
לדוגמא: קלט – 5, 30, 0 → פלט – 05:30:00.
היעזר בפונקציה עם פרמטר.

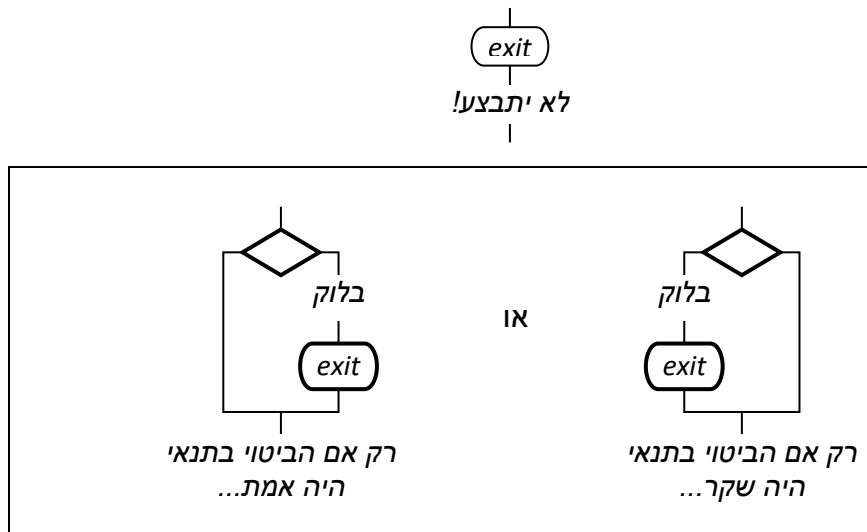


- * קלוט ארבעה נתונים. הצג את מספר הנתונים השליליים.
יתכן שכל הנתונים אי-שליליים או שכולם שליליים.

5.3 סיום מוקדם של תרשים

ניתן לדלג לסוף התרשים המוכיח.

תחביר צומת *exit*



צומת *exit* בתוך פונקציה מפנה
לצומת ה-*return* של אותה
הפונקציה.
צומת *exit* בתרשים הראשי מפנה
לצומת ה-*stop*.
פקודות מתחת לצומת *exit* לא
יתבצעו.

* בתור הקלט חמישה מספרים.

אם הראשון חיובי הצג את החיובי הגדול ביותר, החיובי הקטן ביותר, את סכום כל החיוביים ואת ממוצע כל החיוביים.



שרטט שני פתרונות – אחד עם *exit* ואחד ללא.

[illegible]



1. בתור הקלט תוצאות שלושת משחקיה של מכבי ת"א באליפות אירופה.
כל תוצאה היא זוג שלמים: הראשון הוא מספר הסלים לזכותה של מכבי והשני הוא מספר הסלים לחובתה. ניצחון במשחק מעניק שתי נקודות ותיקו (שוויון) מעניק נקודה.
הצג את מספר הנקודות שצברה מכבי ת"א.

2. קלוט 4 נתונים. הצג את ממוצע הנתונים עד לנתון הראשון השווה 0 (לא כולל).
לדוגמא: עבור רשימת הקלט: 5, 7, 0, 9 →
כל הנתונים יקלטו, אבל יחושב הממוצע של 5 ו-7 בלבד כלומר הפלט יהיה 6.
יש לטפל גם במצבים הבאים: הנתון הראשון הוא 0; אף נתון לא שווה 0.

3. * "בנק אוניברסל" מוכר המחאות נוסעים בדולרים וגובה עמלה באופן הבא:
עבור כל המחאה – 10¢ (ישנם 100¢ ב-\$1);
על כל דולר מה- 500\$ הראשונים – עוד 1¢;
על כל דולר נוסף – עוד 1½¢.
בתור הקלט:
מספר ההמחאות המבוקש
והסכום הכולל של ההמחאות ללא העמלות.
הצג:
את מספר ההמחאות המבוקש,
את העלות הכוללת של ההמחאות ללא העמלות,
את סכום העמלות הכולל
ואת סה"כ לתשלום (עלות + עמלות).

4. עובד במפעלי ישראלקטריק חתום על הסכם השכר הבא:
אם הוותק מעל 10 שנים – תוספת של 10% משכר היסוד.
עבור כל ילד מהרביעי עד השישי (כולל) – תוספת של 400 ש"ח.
עבור כל ילד מהשביעי (כולל) – תוספת של 700 ש"ח.
כל שעה מעבר ל- 160 שעות נחשבת "שעה נוספת".
עבור כל שעה נוספת מ- 16 הראשונות – 80 ש"ח.
עבור כל שעה נוספת מעבר לכך – 120 ש"ח.
קלוט משכורת יסוד, ותק, מספר ילדים ומספר שעות עבודה.
הצג את שכר המגיע לעובד מבלי לשנות את הקלט.

5. חברת המשקאות "האנגובר" קונסת את הספקים עבור נזקים באופן הבא:
אם נשברו שלושה עד חמישה בקבוקים – הקנס מחיר בקבוק.
אם נשברו שישה עד עשרה בקבוקים – הקנס גדל במחיר שני בקבוקים.
אם נשברו מעל עשרה בקבוקים – הקנס גדל במחיר ארבעה בקבוקים.
עבור כל ארגז שאבד – הקנס מחיר עשרה בקבוקים.
קלוט את המחיר לבקבוק ועבור כל אחד מארבעת הספקים:
את מספר הבקבוקים השבורים של הספק ואת מספר הארגזים שאבדו לו.
הצג את הקנס הכולל שעל כל ספק לשלם.

6. מערכים

בפרק זה נלמד את הנושאים הבאים:

- מהו מערך
- שימוש במערך
- שיטות לאתחול מערך
- העברת מערך לפונקציה
- מיון מערך
- חיפוש במערך
- מהו מערך דו - ממדי (מטריצה)
- שימוש במערך דו-ממדי

6.1 מהו מערך

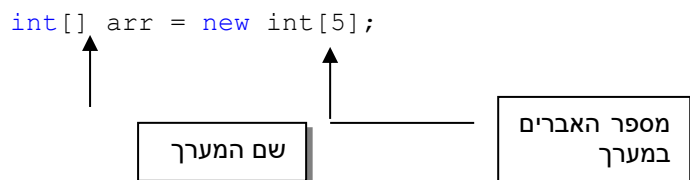
אם נרצה לקלוט מהמשתמש 100 מספרים, ולאחר הקליטה להדפיס אותם בסדר הפוך. באמצעות הכלים שהכרנו עד כה, נצטרך להגדיר 100 משתנים בזיכרון, שיכילו את המספרים. ואם נרצה לקלוט 10000 מספרים, האם נגדיר 10000 תאים בזיכרון? פעולה כזו תהפוך את הקוד למסורבל ובלתי ניתן לתחזוקה. בכדי לפתור בעיות מסוג זה, נשתמש במערך.

מערך – רצף של משתנים בזיכרון.

מערך הינו מספר תאים רציפים בזיכרון. כאשר נגדיר את המערך, נוכל להגדיר בנוסף כמה תאים הוא יכיל. כל תא במערך (שהוא בעצם משתנה לכל דבר), מקבל מספר המציין את מיקומו. מספר זה מכונה אינדקס התא. כאשר נרצה לפנות לתא מסוים במערך, נפנה אליו דרך שם המערך ובנוסף נציין את אינדקס האיבר אליו נרצה לגשת.

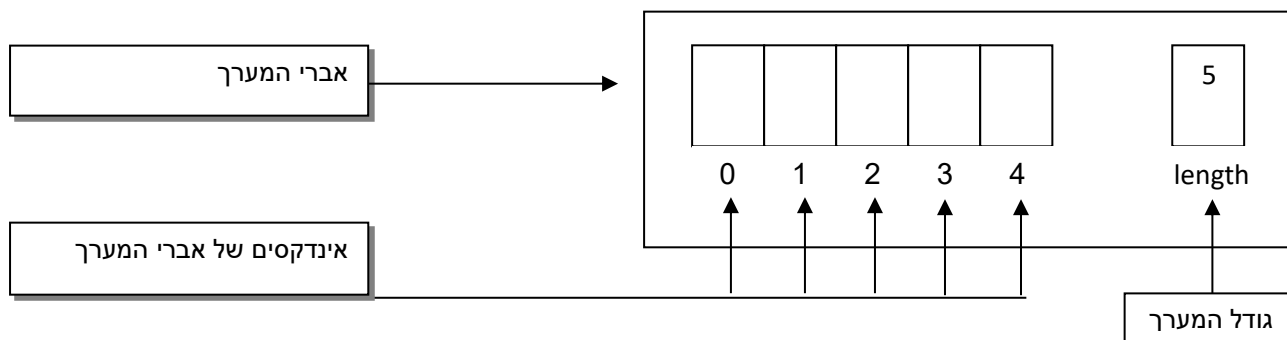
6.2 שימוש במערך

יצירת מערך:



בדוגמא זו הוגדר מערך המכיל 5 תאים.

ציור זיכרון של המערך:



ניתן להבין מהצורה, שהאיבר הראשון במערך מקבל את האינדקס 0, השני את האינדקס 1 וכן הלאה, כאשר האינדקס של האיבר האחרון במערך הוא גודל המערך (length) פחות 1.

ניתן להגדיר בסוגריים, המציינים את גודל המערך, גם שם של משתנה מספרי.

לדוגמא:

```
int size = 10;
int arr = new int[size];
```

בקטע הקוד הנ"ל הוגדר משתנה בשם size המכיל 10, ובאמצעות משתנה זה הוגדר גודל המערך.

בכדי לגשת לאיבר במערך יש לפנות אליו דרך שם המערך ולאחר השם יש לציין את האינדקס של האיבר בתוך סוגריים מרובעים [].

```
arr[0] = 8; //גישה לאיבר הראשון במערך
arr[3] = 13; //גישה לאיבר הרביעי במערך

alert(arr[3]); //הדפסת האיבר הרביעי במערך - יודפס 13
```

גישה לאיבר
מסוים
במערך:

ניתן גם לדעת את גודל המערך באמצעות המשתנה length שהמערך מכיל.

```
alert(arr.length); //הדפסת גודל המערך - יודפס 5
```

הדפסת
גודל
המערך:

חשוב להבין שכאשר ניגש לאיבר מסוים במערך, כל הביטוי המציין את האיבר עצמו מתנהג כמו משתנה בודד.

arr[3] – הינו משתנה אחד בודד מטיפוס Number, ולכן ניתן לבצע על איבר זה כל פעולה שניתן לבצע על משתנה מטיפוס Number, כולל השמה לתוכו, ביצוע פעולות חשבון, קבלת הערך למשתנה אחר, העברה לפונקציה, וכו'.

כאשר נרצה לבצע פעולה מסוימת על כל המערך, נשתמש בד"כ בלולאת מונה, כך שמונה הלולאה יציין בכל איטרציה של הלולאה את אינדקס האיבר במערך.

```
int arr = new int[7];

for (int i = 0; i < arr.length ; i++)
{
    arr[i] = i * 2;
}

String result = "";
for (int i = 0; i < arr.length ; i++)
{
    result += arr[i] + ", ";
}

alert(result);
```

לדוגמא:

0,2,4,6,8,10,12,

פלט:

בדוגמא הנ"ל הוגדר מערך בעל 7 תאים. באמצעות לולאת for, ניתן לרוץ החל מהאיבר ה-0 ועד 1 פחות מגודל המערך (סה"כ 7 איטרציות). כאשר בלולאה הראשונה לכל איבר במערך, יוכנס $i*2$ (מספרים זוגיים החל מ-0). ניתן לראות שבתוך הסוגריים המרובעים, המצינים את אינדקס האיבר, מופיע i. בכל איטרציה של הלולאה i גדל באחד, והדבר יביא לכך שבכל איטרציה נפנה לאיבר הבא במערך. הלולאה השנייה עוברת על כל המערך, ומוסיפה את כל האברים שבמערך, למשתנה מסוג String (בתוספת פסיק). ניתן לראות שהפלט תואם למידע שהוכנס לתוך המערך.

כלל חשוב: כאשר נרוץ בלולאה על מערך, לעולם לא נכתוב במפורש את גודל המערך (כתנאי עצירה לסיום הלולאה), אלא נשתמש במשתנה length של המערך בכדי לדעת מה גודלו. זאת בכדי שהלולאה תתאים לכל גודל של מערך, גם אם נחליט לשנות את גודלו ביום מן הימים.

לאחר יצירת מערך לא ניתן לשנות את גודלו!

6.3 מיון מערך

ישנם מספר אלגוריתמים המיועדים לביצוע מיון של מערך.

- מיון ליניארי
- מיון בועות
- ועוד

שלבים במיון ליניארי:

1. השוואת האיבר הראשון מול כל האיברים שאחריו.
2. במידה שנמצא שאיבר מסוים קטן יותר מהאיבר הראשון – ביצוע של החלפה ביניהם.
3. לאחר סיום ההשוואה מול כל האיברים, בטוח שאיבר הקטן ביותר נמצא במקומו.
4. חזרה לשלב 1 – (השוואת האיבר השני מול כל האיברים שאחריו) וכו', עד שמגיעים לאיבר האחרון.

שלבים במיון בועות:

1. מעבר על כל המערך והשוואה בין שני איברים צמודים (הראשון והשני, השני והשלישי וכו'), "דחיפת" הגדול כלפי מעלה באמצעות החלפה.
2. לאחר סיום מעבר אחד על המערך, בטוח שהאיבר הגדול ביותר נמצא בסוף.
3. חזרה לשלב 1 – ועצירת ההשוואה איבר אחד לפני הסוף וכו'.

6.4 חיפוש במערך

כאשר נרצה לחפש איבר מסוים במערך, יש לרוץ על כל המערך עד שמוצאים את האיבר, או מגיעים לסוף. כך, יודעים שאיבר לא נמצא. במידה שהמערך ממין, ניתן "לחסוך" בחיפושים ולהשתמש בשיטת חיפוש הנקראת "חיפוש בינארי".

שיטת חיפוש בינארי, מזכירה חיפוש של שם בספר טלפונים. במקום לעבור שם אחר שם, עד שנמצא את השם המבוקש, נפתח באמצע הספר ונחליט האם השם המבוקש נמצא לפני או אחרי השם הנוכחי הנמצא בספר, ואז נבחר בחצי הספר שבו השם המבוקש יכול להיות ושוב נחלק אותו באמצע ונחליט האם השם המבוקש נמצא לפני/אחרי השם הנוכחי בספר וכן הלאה עד שנמצא את השם המבוקש.

בשיטת החיפוש הבינארי מחזיקים 2 משתנים:
start – המייצג את התחלת האזור בו האיבר יכול להימצא.
end – המייצג את סוף האזור בו האיבר יכול להימצא.

שלבים בשיטת החיפוש הבינארי:

1. אתחול המשתנים start, end לאיבר הראשון ואחרון במערך (בהתאמה).
2. מציאת האמצע בין start ל-end ובדיקה האם האיבר המבוקש נמצא.
 - א. אם כן – סיום החיפוש.
 - ב. אם לא – בדיקה האם האיבר המבוקש נמצא מימין או משמאל לאמצע.
 1. אם מימין העברת ה-start לאיבר אחד אחרי האמצע.
 2. אם משמאל העברת ה-end לאיבר אחד לפני האמצע.
 - ג. חזרה לשלב 2 – עד אשר ה-start יהיה יותר גדול מה-end.

6.5 מהו מערך דו - ממדי (מטריצה)

עבור תוכנה כלכלית נדרש להחזיק את הטבלה הבאה:

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	total
96													
97													
98													
99													
00													
01													
02													
03													
04													
05													
06													

לכל שנה בעשור האחרון, יוחזקו 12 המדדים החודשיים, ובנוסף יוחזק המדד השנתי. למעשה, אנו צריכים מערך של 10 מערכים בגודל 13 כל אחד. בכל מערך של 13, תוחזק שנה אחת. בתא האחרון יושב המדד השנתי, ובתאים 0-11 יושבים מדדי החודשים.

כדי לפתור בעיה כזו, עם היכולות המוכרות לנו עד עתה, עלינו ליצור 10 מערכים. ישנה דרך יותר פשוטה לפתור בעיה זאת והיא להשתמש במערך דו-ממדי.

מערך דו-ממדי, דומה מאד בתפיסתו למערך חד-ממדי. ההבדל ביניהם הוא, שכאשר ניצור מערך חד-ממדי תיווצר שורה אחת של תאים בזיכרון, וכאשר ניצור מערך דו-ממדי ייווצרו בזיכרון מספר שורות. מערך דו-ממדי מזכיר טבלה עם שורות ועמודות ומכונה גם מטריצה.

תרגילים

מערכים



- קלוט נתונים לתוך מערך בגודל 50. הדפס את ממוצע המספרים במערך וכן את כל המספרים הגדולים מהממוצע.
- קלוט נתונים לתוך מערך בגודל 100 והדפס את ממוצע המספרים הזוגיים בלבד.
- קלוט נתונים לתוך מערך בגודל 60 ובדוק האם קיים תא שערכו שווה לסכום כל התאים האחרים. אם כן, הדפס את מיקומו במערך.
- קלוט נתונים לתוך מערך בגודל 20. החלף את עשרת התאים הראשונים בעשרת התאים האחרונים (תאים 0 ו-10 מתחלפים, תאים 1 ו-11 מתחלפים וכו').

5. קלוט נתונים מספריים למערך בגודל 60 וסדר את הנתונים במערך, כך שכל הזוגיים יהיו בחלקו הימני. הדפס את המערך המקורי ולאחר סידורו מחדש.
6. קלוט נתונים למערך בגודל 20. החלף את סדר התאים מהסוף להתחלה (תאים 0 ו-19 מתחלפים, 1 ו-18 מתחלפים וכו').
7. קלוט נתונים תווים למערך בגודל 50. הדפס כמה הם ספרות, כמה אותיות וכמה אחרים.
8. בדוק האם מערך בגודל 10 מהווה פולינדרום או לא.
9. קלוט נתונים למערך בגודל 10 ולמערך בגודל 50. בדוק האם המערך הקטן נמצא במלואו במערך הגדול. אם כן, הדפס את מספר התא במערך הגדול בו הקטן מתחיל.
10. קלוט נתונים תווים למערך בגודל 50 והדפס כמה פעמים מופיע כל תו במערך.
11. קלוט נתונים למערך בגודל 10 וקוד (+ או -). הזז את המערך שמאלה או ימינה, לפי הקוד, בצורה מעגלית!

מטריצות

1. מלא מטריצה בגודל 10×10 במספרים עוקבים (0-99).
2. בנה מטריצה אשר כולה מלאה באפסים, חוץ מהאלכסונים אשר יהיו 1.
3. קלוט נתונים למטריצה בגודל 10×10 . הדפס את מספר השורה בה סכום האברים הוא הגבוה ביותר.
4. קלוט נתונים למטריצה בגודל 10×10 . אפס את כל התאים המתחלקים בשלמות ב-5 או ב-7.
5. קלוט נתונים למטריצה בגודל 10×10 . בדוק האם המטריצה מהווה ריבוע קסם (סכום כל שורה = סכום כל עמודה = סכום כל אלכסון).
6. קלוט נתונים למטריצה בגודל 10×10 ומטריצה בגודל 40×40 . בדוק האם המטריצה הקטנה מוכלת בגדולה.
7. בכיתה יש 30 תלמידים ולכל תלמיד 10 ציונים. קלוט את נתוני כל התלמידים והדפס מה ממוצע כל תלמיד ומיהו התלמיד המצטיין. כמו כן, מצא מהו הממוצע בכל מקצוע ומהו המקצוע בו מספר הנכשלים הגבוה ביותר.
8. מלא מטריצה בגודל 10×10 בצורה הבאה: המסגרת החיצונית בספרה 1, המסגרת הפנימית יותר בספרה 2 וכך הלאה.

7. סיום

תרגילים מתקדמים

המשימות בתרגילים הבאים מורכבות.
מומלץ לפרק את המשימות לתתי-משימות
ולא לנסות להתמודד מיד עם המשימה הסופית.

1. קלוט מספר שורות ומספר עמודות והצג מלבן סולמיות מלא בטילדות.
למשל: קלט – 4, 6 →



פלט –

~ ~ ~ ~ #
~ ~ ~ ~ #
#

ודא את תקינות הקלט.

2. * קלוט שלם המייצג רוחב והצג את הצורה הבאה: (פלט עבור הקלט 5)

5
5 4
5 4 3
5 4 3 2
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5

3. קלוט שלם המייצג רוחב בסיס והצג את מגדל המשולשים הבא: (פלט עבור הקלט 5)

1
2 2
2
3 3 3
3 3
3
4 4 4 4
4 4 4
4 4
4
5 5 5 5 5
5 5 5 5
5 5 5
5 5
5

4. קלוט זמן התחלה (שעה, דקה ושניה) וזמן סיום. הצג את כל השניות מזמן ההתחלה ועד זמן הסיום בפורמט של שעות דיגיטלי – hh:mm:ss.

לדוגמא: קלט – 23, 59, 58, 0, 0, 2 →

פלט – 23:59:58

23:59:59

00:00:00

00:00:01

00:00:02

5. * קלוט זמן התחלה (שעה, דקה ושניה) וזמן סיום. הצג את הזמן שחלף

מזמן ההתחלה ועד זמן הסיום בפורמט של שעות דיגיטלי – hh:mm:ss.

לדוגמא: קלט – 23, 59, 58, 0, 0, 2 →

פלט – 00:00:04

6. קלוט מספרים טבעיים (שלמים הגדולים מ-0) עד שייקלט זקיף שלילי.

הצג את המחלק המשותף הגבוה ביותר של כל הטבעיים.

לדוגמא: קלט – 1, -1, 120, 48, 60, 24 → פלט – 12.

קלט – 1, -1, 118, 48, 60, 24 → פלט – 2.

7. * הנוסחה הבאה פועלת על שני מספרים x ו- n :

$$f(x,n) = -\frac{x^2}{1!} + \frac{x^4}{2!} - \frac{x^6}{3!} + \frac{x^8}{4!} \dots \frac{x^{2n}}{n!}$$

למשל, נציב $x = 4$ ו- $n = 3$:

$$+\frac{4^4}{2!} - \frac{4^6}{3!} = -\frac{16}{1} + \frac{256}{2} - \frac{4096}{6} = -570 \frac{2}{3}$$

תזכורת: n עצרת היא מכפלת השלמים מ-1 עד n : $n! = 1 \times 2 \times \dots \times n$

למשל, 5 עצרת היא 120 כי $5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$

קלוט ערכים עבור x ו- n , והצג את תוצאת הנוסחה.

8. מספר מושלם הוא שלם הגדול מ-1, השווה לסכום כל מחלקיו (לא כולל את

עצמו). למשל 6 הוא מספר מושלם כי $1 + 2 + 3 = 6$.

קלוט שלם n , והצג את n המספרים מושלמים הראשונים.

לדוגמא: קלט – 2.

פלט – 6, 28 →

9. מספר זהות ישראלי מורכב ממספר בן עד שמונה ספרות, ומספרת ביקורת.

ניתן לחשב את ספרת הביקורת באופן הבא:

1	2	3	4	5	6	7	8	מ"ז של ישראל ארצי הוא 12345678-?
---	---	---	---	---	---	---	---	-------------------------------------

1	4	3	8	5	12	7	16	כופלים ב-2 כל ספרה שניה (משמאל לימין).
1	4	3	8	5	3	7	7	מחברים את הספרות.
$1 + 4 + 3 + 8 + 5 + 3 + 7 + 7 = 38$								מסכמים את התוצאות.
$100 - 38 = 62$								מחסירים מ-100 את הסכום.
מ"ז של ישראל ארצי הוא 12345678-2								ומבודדים את ספרת האחדות.

קלוט שם ומספר זהות של אדם (שלם אחד).
הצג האם מספר הזהות נכון או שגוי.

10. * המכולת "סופר מרק" הייתה פתוחה מידי יום למעט שבתות, מיום ב' ה- 1/1/1996 עד יום ב' ה- 31/12/2001.
בתור הקלט מספר הלקוחות היומי (למעט שבתות) מ-1996 עד 2001.
הצג את החודש העמוס ביותר בשש השנים הנ"ל ואת מספר הלקוחות שנכנסו למכולת באותו החודש. הפלט עשוי להיות:
11640 customers entered Super Marky during 12 of 1999.

11. קלוט את התאריך של היום (יום, חודש ושנה).
לאחר מכן קלוט שם ותאריך לידה של אנשים עד שתקלוט שם ריק("").
לכל איש הצג את הגיל בשנים, ו-"Happy Birthday" אם הוא חוגג היום.

12. * קלוט תאריך לידה של אדם מסוים (יום, חודש ושנה) ואת מספר היום בשבוע באותו תאריך.
(7 → 1). הצג את כל הגילאים להם יחגוג האדם את יום ההולדת באותו היום בשבוע עד 120.

13. בתור הקלט גובהם של כל אחד משחקני נבחרת הכדורסל באופן הבא:
הנתון הראשון הוא שיטת המדידה: "Metric" או "Imperial", או זקיף: "Done".
המשמעות של שני הנתונים הבאים תלויה בנתון הראשון.
"Metric" – מטרים וסנטימטרים.
"Imperial" – פיט ואינצ'ים.
הצג את גובהו של כל שחקן נבחרת ואת הגובה הממוצע של הנבחרת בשתי שיטות המדידה.
(אינץ' = 2.54 ס"מ, פוט = 12 אינץ' = 30.48 ס"מ)

14. * מקשי המחשבון שולחים לתור הקלט את מספר המקש שהוקש:
0→9 מקשי הספרות.
10 נקודה עשרונית.
11 חיבור.
12 חיסור.
13 כפל.
14 חילוק.
15 שווה (זקיף).
קלוט הקשות עד הקשה על שווה. הצג את התוצאה.
לדוגמא: קלט – 2, 5, 12, 5, 13, 4, 0, 15 →
פלט – 800 (כי $25 - 5 = 20$, $20 \times 40 = 800$).

מבחן לדוגמא

הנחיות

ענה על כל השאלות. משקל כל השאלות זהה.

משך המבחן: 3 שעות (180 דקות)

חומר עזר: אישי ודומם

שים לב:

- אין צורך לוודא את תקינות הקלט.

שאלה 1

בתור הקלט כמות לא ידועה של מספרים טבעיים (שלמים הגדולים מ-0).

הנתון האחרון (הזקיף) הוא (-1).

הצג את שלושת המספרים עם הכי הרבה ספרות זוגיות.

לדוגמא, עבור הקלט: -1, 505, 66, 17775, 45, 888, 7250 →

יהיה הפלט: 66, 7250, 888

שים לב:

- אין חשיבות לסדר הצגת המספרים.
- אין צורך לוודא את תקינות הקלט.
- חובה** להעיר: איך יטפל האלגוריתם במקרי קצה?

שאלה 2

עקב גל עליה בלתי צפוי מסין, משרד הפנים הוסיף ספרה עשירית למספר הזהות של כל אזרח. ספרת העשרות קובעת באיזה מקום תתווסף הספרה, וספרת האחדות קובעת איזו ספרה תתוסף.

לדוגמא, במספר הזהות 302714549 ספרת העשרות היא 4.

לכן יש הוסיף את הספרה בין 4549 לבין 30271.

הספרה שיש להוסיף היא ספרת האחדות: 9.

כלומר מספר הזהות החדש הוא 3027194549.

קלט שם של אזרח ואת מספר הזהות שלו (כשלם אחד).

הצג שם האזרח ואת מספר הזהות החדש.

שאלה 3

בעצרת האו"ם 170 מדינות.

בהצבעה שולחת כל מדינה בתורה קוד לתור הקלט לפי בחירתה:

1. בעד.

2. נגד.

3. נמנע.

חמש המדינות הראשונות יכולות לבחור גם:

4. וטו.

החלטה מתקבלת בתנאים הבאים:

- אף אחת מחמש המדינות הראשונות לא הטילה וטו.
- מספר המדינות המתנגדות קטן ממספר התומכות.
- המדינות התומכות מהוות לפחות $\frac{1}{4}$ מהמדינות החברות באו"ם.

בתור הקלט קודי הצבעה של 170 המדינות לפי סדר.

אם אחת מחמש המדינות הראשונות הטילה וטו – הצג את מספרה.

אחרת – הצג האם ההצעה נתקבלה או נדחתה.

שאלה 4

קלוט מספר של מסגרות (עד 10).

הצג ריבוע ממסגרות של ספרות.

לדוגמא: עבור הקלט 6 יהיה הפלט:

```

1 1 1 1 1 1 1 1 1 1 1
1 2 2 2 2 2 2 2 2 2 1
1 2 3 3 3 3 3 3 3 2 1
1 2 3 4 4 4 4 4 3 2 1
1 2 3 4 5 5 5 4 3 2 1
1 2 3 4 5 6 5 4 3 2 1
1 2 3 4 5 5 5 4 3 2 1
1 2 3 4 4 4 4 4 3 2 1
1 2 3 3 3 3 3 3 3 2 1
1 2 2 2 2 2 2 2 2 2 1
1 1 1 1 1 1 1 1 1 1 1

```

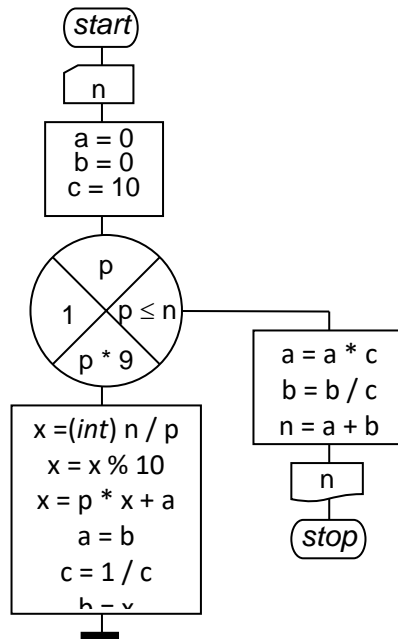
שים לב:

- חובה לוודא את תקינות הקלט.

שאלה 5

נתון התרשים הבא:

1. מה יהיה הפלט עבור הקלט 1234? עקוב בעזרת טבלת מעקב.



2. מה יהיה הפלט עבור הקלט 9?

3. תן דוגמא לקלט שלם בן 6 ספרות עבורו הפלט יהיה זהה לו.

8. נספחים

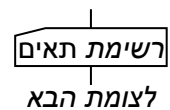
נספח א הצמתים בתרשים הזרימה

לכל הצמתים בהם לא צוין אחרת, קו היוצא מהבסיס מוביל לצומת הבא.

פעולות

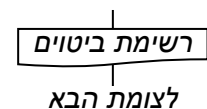
קלט

תחביר: מלבן כתום קדקוד שבו רשימת תאים (לפחות תא אחד).
משמעות: עבור כל תא: הוצא נתון מתור הקלט ושמור את ערכו במשתנה.
הערה: אם אין משתנה בשם זה, הוא נוצר. אם יש, ערכו נדרס ע"י הערך החדש.



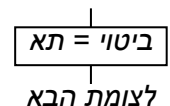
פלט

תחביר: מלבן עם בסיס גלי שבו רשימת ביטויים (לפחות ביטוי אחד).
משמעות: עבור כל ביטוי: הצג את ערכו.



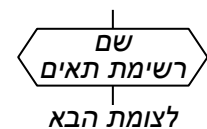
השמה

תחביר: מלבן שבו משתנה, הסימן = (מקבל) וביטוי.
משמעות: שמור את ערכו של הביטוי במשתנה.
הערה: אם אין משתנה בשם זה, הוא נוצר. אם יש, ערכו נדרס ע"י הערך החדש. כמו כן, ניתן לרשום מספר השמות רצופות באותו מלבן.



זימון

תחביר: משושה שבו שם של פונקציה. אם בצומת הכותרת של הפונקציה יש רשימת פרמטרים תרשם רשימת משתנים באותו אורך, מתחת לשם הפונקציה.
משמעות: בצע את הפונקציה.



מסופים

אם לא צוין אחרת, יהיו בצורת אליפסה סגלגלה עם שם המסוף.

start

משמעות: נקודת ההתחלה של כל ההוראות בפתרון.
הערה: בכל פתרון יש בדיוק צומת start אחד.



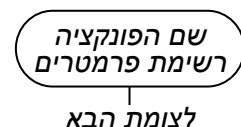
stop

משמעות: נקודת הסיום של הפתרון.
זרימה: אין יותר צמתים.
הערה: בכל פתרון יש בדיוק צומת stop אחד.



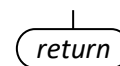
כותרת פונקציה

תחביר: סגלגל שבו שם הפונקציה, ורשימת פרמטרים אם צריך.
משמעות: נקודת ההתחלה של הפונקציה.
הערה: בכל פונקציה יש בדיוק צומת כותרת פונקציה אחד.



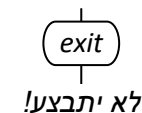
return

משמעות: נקודת הסיום של הפונקציה, הפניה לתרשים המזמן.
זרימה: מדלג לצומת הבא אחרי צומת הזימון בתרשים המזמן.
הערה: בכל פונקציה יש בדיוק צומת return אחד.



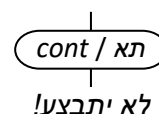
exit

משמעות: הפניה לסוף התרשים הנוכחי (צומת stop או צומת return).
זרימה: יש קו מתחת, אך יש לדלג לצומת המסיים של התרשים.
הערה: לא יכול להופיע בבולוק של לולאה.



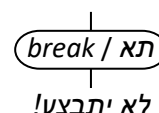
cont

תחביר: סגלגל שבו המלה cont או cont קו-נטוי משתנה.
משמעות: cont – הפניה לצומת של הלולאה הפנימית.
משתנה cont / – הפניה לצומת לולאת האינדקס המתאים.
זרימה: יש קו מתחת, אך יש לדלג חזרה לצומת הלולאה.
הערה: יכול להופיע אך ורק בבולוק של לולאה.



break

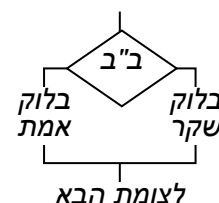
תחביר: סגלגל שבו המלה break או break קו-נטוי משתנה.
משמעות: break – הפניה לצומת הבא אחרי צומת הלולאה הפנימית.
משתנה break / – הפניה לצומת הבא אחרי צומת לולאת האינדקס המתאים.
זרימה: יש קו מתחת, אך יש לדלג לצומת הבא אחרי צומת הלולאה.
הערה: יכול להופיע אך ורק בבולוק של לולאה.



בקרת זרימה

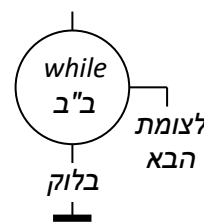
בצמתי בקרת זרימה, יש ביטוי בוליאני (ב"ב).
כשמגיעים לצומת, מעריכים את הב"ב כדי לקבוע את הפעולה.

תנאי תחביר: מעוין שבו ב"ב. מכל צד יוצא קו לבולוק. שני הבולוקים מתחברים בסופם.
משמעות: אמת – בצע בולוק ימני. שקר – בצע בולוק שמאלי.



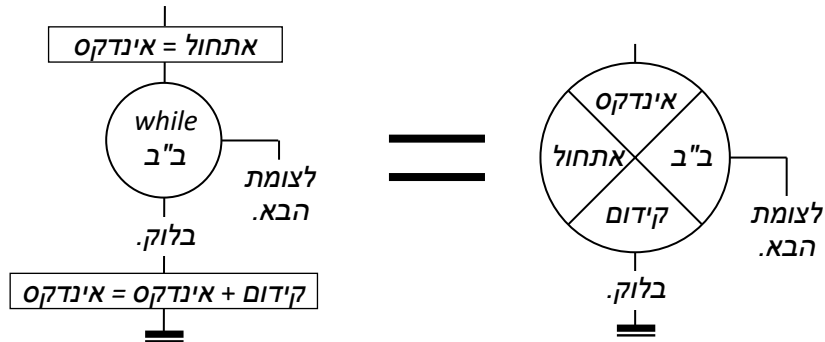
לולאת תנאי

תחביר: מעגל שבו המלה while וב"ב. מתחתית המעגל יוצא קו לבולוק.
משמעות: אמת – בצע את הבולוק וחזור לצומת הזה. שקר – המשך.
זרימה: הקו לצומת הבא יוצא מימין של המעגל.
הערה: לא יכול להופיע בבולוק של צומת תנאי.



לולאת אינדקס

לולאת אינדקס היא איחוד של לולאת תנאי ושתי השמות:



נספח ב האופרטורים בתרשים הזרימה

סוג	קדימות	אופרטור	משמעות
כללי	1	a (סוגריים)	a (משנה את סדר הפעולות).
		$ b $ ערך מוחלט.	אם a שלילי $-b$, אחרת b .
		$+b$ פלוס.	b .
		$-b$ מינוס.	הופך את b משלילי לחיובי, ולהפך.
	2	$b ** c$ חזקה.	b בחזקת c .
	3	$b * c$ כפל.	b פי c .
		b / c חילוק.	b חלקי c .
		$d \% e$ מודולו.	השארית מחלוקה של d ב- e .
	4	$b + c$ חיבור.	b ועוד c .
		$b - c$ חיסור.	b פחות c .
בוליאני (לוגי)	5	$b = c$ שווה.	האם לשני הביטויים אותו ערך?
		$b > c$ גדול מ-.	האם ל- b ערך גבוה מ- c ?
		$b < c$ קטן מ-.	האם ל- b ערך נמוך מ- c ?
		$b \neq c$ שונה מ-.	האם לשני הביטויים אין את אותו ערך?
		$b \leq c$ קטן/שווה.	האם ל- b ערך נמוך מ- c או שווה לו?
		$b \geq c$ גדול/שווה.	האם ל- b ערך גבוה מ- c או שווה לו?
	6	$f \&\& g$ וגם.	האם שני הביטויים הם אמת?
	7	$f g$ או.	האם לפחות אחד הביטויים אמת?

סוג

- סוגריים לא משפיעים על סוג הביטוי.
- אופרטורים חשבוניים מפיקים תוצאה מספרית.
- אופרטורים בוליאניים מפיקים תוצאה בוליאנית: או אמת או שקר.

קדימות

- כשמעריכים ביטוי מתחילים מהאופרטורים בעלי הקדימות הגבוהה ביותר (1) ויורדים בהדרגה. כדי לשנות את סדר הפעולות, משתמשים בסוגריים.

אופרנטיים

- a הוא ביטוי כלשהו (מספרי או בוליאני).
- b ו- c הם ביטויים מספריים.
- d ו- e הם ביטויים מספריים שערכם שלם.
- f ו- g הם ביטויים בוליאניים.

נספח ג מונחון

actual parameter פרמטר אקטואלי
ביטוי שהתהליך המזמן שולח לפונקציה.

formal parameter פרמטר פורמלי
משתנה פנימי של הפונקציה המקבל את ערכו מהתהליך המזמן.

input קליטה
התהליך של העברת נתונים מהתקן חיצוני כלשהו, לתוך משתנים בזיכרון.

input קלט
הנתונים המועברים בתהליך הקליטה.

input buffer תור הקלט
שטח בזיכרון שהוקצה כדי לקבל נתונים מהתקן חיצוני כלשהו לפני העברתם למשתנים המתאימים.

dry running ריצה יבשה
בדיקת נכונות של תרשים זרימה או תכנית ע"י רישום תוצאות הפעולות בטבלת מעקב לפני הרצת התכנית במחשב.

programming תכנות
תהליך בו מפיקים סדרת הוראות עבור מחשב לביצוע פעילות מסוימת.

programming language שפת תכנות
שפה בה ניתן להפיק סדרת הוראות חד משמעית עבור מחשב, תוך שימוש בקלט, פלט, משתנה, תנאי ולולאה. שפת תכנות מתורגמת לשפת המכונה של המחשב ע"י תוכנה: מהדר (compiler) או פרשן (interpreter).

structured programming תכנות מבני
תכנות לפי העקרונות המוצגים במודול זה (כללי זרימה קשוחים, שימוש בפונקציות) ועקרונות נוספים (מודולריות, הסתרת מידע, מבני נתונים). מתודולוגיות נפוצות היום, כגון תכנות מונחה עצמים (event driven), תכנות מונחה אירועים (event driven) ותכנות מרובה משימות (multi-threaded) כוללות בתוכן את עקרונות התכנות המבני.

flowchart תרשים זרימה
שרטוט המתאר תהליך. תקנים שונים לשרטוט תרשימי זרימה מתאימים לתיאור תהליכים בסדר גודל שונים, מאלגוריתמים ועד מערכות סבוכות. בד"כ תרשים זרימה מורכב מסמלים (symbols) מוסכמים ומחצים. בג'ון ברייס אין חצים, אלא קווים, והסמלים (צמתים), קובעים את כיוון הזרימה. תרשימי זרימה אלה מתאימים לתיאור אלגוריתמים לפי עקרונות התכנות המבני.

iteration איטרציה
"סיבוב". ביצוע יחיד של בלוק של לולאה.

algorithm אלגוריתם
סדרת הוראות קבועה לביצוע משימה, בה נתונים לא ידועים מראש.

bug באג
שגיאה לוגית או תחבירית בתכנית.

expression ביטוי
משתנה, קבוע או חישוב.

boolean expression ביטוי בוליאני
ביטוי שערכו ברגע נתון אמת או שקר.

block בלוק
רצף של הוראות (אפס או יותר). בשפות C, C++, C#, JavaScript ו-Java, הבלוק תחום בצומדיים {curly parenthesis} ונחשב תחבירית להוראה אחת.

assignment השמה
העתקת תוצאה של ביטוי לתוך משתנה.

(watch table) טבלת מעקב
טבלה בה רושמים את השינויים במשתנים, במהלך ריצה יבשה.

loop לולאה
בלוק שעשוי להתבצע מספר פעמים בהרצה בודדת.

identifier מזהה
שם ייחודי, למשל של משתנה או של פונקציה.

variable משתנה
משתנה. שטח בזיכרון בו ערך יחיד ברגע נתון.

debugging ניפוי שגיאות
איתור השגיאות בתכנית ע"י תוכנה (debugger) ו/או אדם.

function פונקציה
תת-תכנית.

function call זימון פונקציה
הפעלת פונקציה מתוך התכנית הראשית או מתוך פונקציה אחרת. כשתהליך מזמן פונקציה הוא נעצר, הפונקציה מתבצעת, וכשהיא מסתיימת התהליך ממשיך מההוראה הבאה.

output פלט
התוצר של הרצת תכנית.

parameter פרמטר
ערך הדרוש לפונקציה כדי שתוכל להתבצע.