

ב"ה

**תרגיל מס' 5 – מצביעים והקצאה דינאמית (מתרגלת אחראי: פריאל)
התרגיל שווה 20% מציון התרגול**

הוראות הגשה

- שאלות בנוגע לתרגיל נא להפנות דרך פורום הקורס שנפתח במיוחד לשם כך :
<https://piazza.com/biu.ac.il/fall2016/89110/>
אם לא נענתה תשובה תוך 24 שעות, נא לשלוח אלי (פריאל) מייל עם לינק לדיון הרלוונטי ואענה. המייל הוא : priel.levy@live.biu.ac.il. בכל מייל יש לציין שם, שם משתמש, מס' קורס, וקבוצת תרגול.
- מועד פרסום : 22/12/16
 - מועד אחרון להגשה : 23: 59 05/01/17
 - יש לשלוח את הקבצים באמצעות האתר :
<http://help.cs.biu.ac.il/submit.htm> לפני חלוף התאריך הנקוב לעיל.
 - יום איחור גורר הורדה אוטומטית של 10 נקודות.
 - יומיים איחור גורר הורדה אוטומטית של 20 נקודות.
 - שם ההגשה של התרגיל : ex5
 - יש להקפיד מאוד על כל הוראות עיצוב הקלט והפלט, כמפורט בכל סעיף וסעיף. על הפלט להיראות בדיוק כמו בדוגמאות. אין להוסיף או להשמיט רווחים או תווים אחרים ואין להחליף אותיות גדולות בקטנות או להיפך ⊗ אי-הקפדה על פרטים אלה עלול לגרור ירידה משמעותית ביותר בציון התרגיל עד כדי 0. ראו הוזהרתם!
 - להזכירכם, העבודה היא אישית. "עבודה משותפת" דינה כהעתקה.
 - אין להדפיס למסך שום דבר מעבר למה שנתבקש בתרגיל.
 - יש לוודא שהתרגיל מתקמפל ורץ על השרתים באוניברסיטה (u2) ללא שגיאות/אזהרות.
 - אתם יכולים לעבוד עם כל עורך טקסטואלי שאתם מעדיפים. להזכירכם – pico בשרתי linux שבמעבדות ; notepad ב-windows ; או בסביבת פיתוח (IDE) – עבוד C השתמשו ב-Visual Studio שכולל בתוכו מהדר ל-windows. מומלץ להתנסות בכולם.

**הקפידו על כתיבה לפי קובץ ה-Coding-Style שבאתר הקורס!!
אין להשתמש בחומר שלא נלמד עד תרגול 7 (כולל) בתרגיל זה
בתרגיל זה אין להשתמש במבנים (structs)**

Pointers and Dynamic Allocations

הנחיות עבור ex5

- שם התרגיל: ex5
- יש לשלוח את קבצי המקור (.c), קבצי ה header (.h) וקובץ makefile. בנוסף הגישו קובץ שנקרא main.c שיכיל את פונקציית ה main.

הוראות כלליות לתרגיל

כתוב תכנית בעלת הזרימה הבאה:

1. קלוט מספר שלם (ניתן להניח שיתקבל מספר שלם)
2. אם התקבל 1 בצע את **משימה 1**.
3. אם התקבל 2 בצע את **משימה 2**.
4. אם התקבל 3 בצע את **משימה 3**.
5. אם התקבל 4 בצע את **משימה 4**.
6. אם התקבל 5 בצע את **משימה 5**.
7. אם התקבל 6 בצע את **משימה 6**.
8. אם התקבל 0 סיים את התוכנית.
9. אם התוכנית לא הסתיימה הדפס תו ירידת שורה וחזור לשלב 1.

משימה 1:

המשימה תקלוט מהמשתמש מספר (שיציין את אורך המחרוזת, ניתן להניח גדול מאפס) ולאחר מכן תקלוט מחרוזת str באורך זה. ראו הסברים ודוגמאות בסוף.

לצורך המשימה יש לממש את הפונקציה `int SortString(char* str)` המקבלת את המחרוזת str (ניתן להניח שאינה מכילה רווחים) וממיינת אותה כך שכל האותיות הקטנות יימצאו בתחילת המחרוזת, מסודרות ביניהן לפי סדר הא"ב; וכל האותיות הגדולות יימצאו בסוף המחרוזת, גם מסודרות ביניהן לפי סדר הא"ב.

הפונקציה תחזיר 1 במקרה ונדרשה פעולת החלפה אחת לפחות בין אותיות המחרוזת, או 0 אחרת (כלומר, הפונקציה תחזיר 0 במקרה וקיבלה מחרוזת שכבר ממוינת בהתאם לדרישות).

על המחרוזת str להיות מורכבת מאותיות באנגלית קטנות או גדולות, כל קלט אחר אינו חוקי.

במקרה של קלט שאינו חוקי, תודפס ההודעה: `Error: Illegal input` והתוכנית תחזור לתפריט הראשי. במקרה והקצאת הזיכרון נכשלה, תודפס ההודעה: `Error: Cannot allocate Memory` והתוכנית תחזור לתפריט הראשי.

לדוגמא:

- המחרוזת bXAcaY לאחר מיון תהיה abcXY והפונקציה תחזיר 1.
- המחרוזת CVYH לאחר מיון תהיה CHVY והפונקציה תחזיר 1.
- עבור המחרוזת abCD לא יתבצע מיון והפונקציה תחזיר 0.

משימה 2:

המשימה תקלוט מהמשתמש שני מספרים מופרדים ברווח. המספר ראשון יציין את אורך המחרוזת הראשונה והמספר שני יציין את אורך המחרוזת השנייה. ניתן להניח ששניהם גדולים מאפס. לאחר מכן תקלוט מחרוזת str באורך המספר הראשון ומערך של תווים sub באורך המספר השני, גם הם מופרדים ברווח. ראו הסברים ודוגמאות בסוף.

לצורך המשימה יש לממש את הפונקציה `int SubGenerator(char *str, char *sub)` המקבלת את המחרוזת `str` (ניתן להניח שאינה מכילה רווחים) ומערך התווים `sub` (ניתן להניח שאינו מכיל רווחים) ומחזירה את מספר הפעמים ששרשור מערך התווים `sub` בונה את המחרוזת `str`, ו-0 אחרת (כלומר 0 במידה ושרשור `sub` אינו בונה את `str`). ניתן להניח שהקלט מורכב רק מאותיות ומספרים. במקרה והקצאת הזיכרון נכשלה, תודפס ההודעה: `Error: Cannot allocate Memory` והתוכנית תחזור לתפריט הראשי.

לדוגמא:

עבור המחרוזת `abcabcabcabc` ומערך תווים `abc` הפונקציה תחזיר 4 כי שרשור `abc` 4 פעמים בונה את המחרוזת. עבור המחרוזת `abcabccabcab` ומערך תווים `abc` הפונקציה תחזיר 0 כי כל שרשור של `abc` לא בונה את המחרוזת.

משימה 3:

רישא של מחרוזת תווים היא מערך תווים הנמצאת בתחילתה של המחרוזת. למשל: למחרוזת `abca` ישנן 4 רישות שונות והן:

- a – התו הראשון
- ab – שני התווים הראשונים
- abc – שלושת התווים הראשונים
- abca – כל איברי המחרוזת

המשימה תקלוט מהמשתמש מספר (שיציין את אורך המחרוזת, ניתן להניח גדול מאפס) ולאחר מכן תקלוט מחרוזת `str` באורך זה. ראו הסברים ודוגמאות בסוף.

לצורך המשימה יש לממש את הפונקציה `int ShortestStr(char *str)` המקבלת את המחרוזת `str` (ניתן להניח שאינה מכילה רווחים) ומחזירה את אורכה של הרישא הקצרה ביותר אשר שרשרה בונה את המחרוזת (ניתן להיעזר בפונקציה שמימשתם במשימה 2). ניתן להניח שהקלט מורכב רק מאותיות ומספרים. במקרה והקצאת הזיכרון נכשלה, תודפס ההודעה: `Error: Cannot allocate Memory` והתוכנית תחזור לתפריט הראשי.

לדוגמא:

עבור המחרוזת `abcabc` הפונקציה תחזיר 3 כי `abc` היא הרישא הקצרה ביותר הבונה את המחרוזת. עבור המחרוזת `abcab` הפונקציה תחזיר 5 כי `abcab` היא הרישא הקצרה ביותר הבונה את המחרוזת.

משימה 4:

המשימה תקלוט מהמשתמש מספר (שיציין את אורך המחרוזת, ניתן להניח גדול מאפס), לאחר מכן תקלוט מחרוזת `str` באורך זה, ולבסוף מספר `n` גדול מאפס. ראו הסברים ודוגמאות בסוף.

לצורך המשימה יש לממש את הפונקציה `void ReverseNumWords(char *str, int n)` המקבלת את המחרוזת `str` ואת המספר `n`. הפונקציה תשנה את המחרוזת `str` כך שרק `n` המילים הראשונות בסדר הפוך והשאר לא (כלומר בסדר המקורי).

מילה במחרוזת מוגדרת כרצף תווים שמופרדת ממילים אחרות ע"י רווח יחיד.
במקרה של קלט שאינו חוקי (כלומר מספר n שגדול ממספר המילים במשפט), תודפס ההודעה:
Error: Illegal input והתוכנית תחזור לתפריט הראשי. במקרה והקצאת הזיכרון נכשלה, תודפס
ההודעה: Error: Cannot allocate Memory והתוכנית תחזור לתפריט הראשי.

לדוגמא:

עבור המחרוזת הבאה ו $n=3$:

“Sometimes the questions are complicated and the answers are simple.” - Dr. Seuss
המחרוזת תשתנה ל –
questions the “Sometimes are complicated and the answers are simple.” - Dr. Seuss

הערות למשימות 1-4:

- הקפידו להשתמש בפונקציות עזר.
- אסור להגדיר מערכי עזר (כן ניתן להשתמש במצביעים נוספים).
- עליכם להשתמש בהקצאות דינאמיות.
- הקפידו לחלק את הקוד לקבצים נפרדים לפי הקשר הלוגי של הפונקציות.
- במידה והונס קלט ששונה מהאפשרויות בתפריט, תודפס ההודעה: Error: Illegal input והתוכנית תחזור לתפריט הראשי.

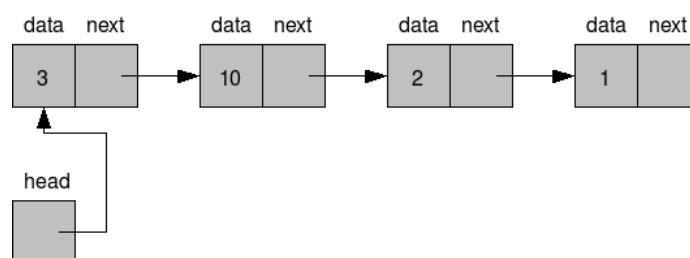
משימה 5:

במשימה זו, התוכנית תקלוט מספר לא מוגבל של שמות מהמשתמש, עד הקלט “QUIT”.
התוכנית תדפיס את מספר השמות שהוכנסו ולאחר מכן תדפיס את כל השמות ממוינים לפי סדר
האותיות באנגלית. ניתן להניח שהשמות מכילים רק אותיות קטנות באנגלית, ללא מספרים וללא
רווחים.
במקרה והקצאת הזיכרון נכשלה, תודפס ההודעה: Error: Cannot allocate Memory והתוכנית
תחזור לתפריט הראשי.
ראו הסברים ודוגמאות בסוף.

משימה 6:

רשימה מקושרת (Linked list) היא מבנה נתונים דינאמי שמורכב משרשרת חוליות – כל חוליה
מכילה תוכן מסוים ומצביעה לאיבר הבא. בשונה ממערך, האורך של הרשימה יכול להשתנות,
והחוליות לא בהכרח יושבות במקום רציף בזיכרון, אך עדיין קשורות אחת לשנייה באמצעות
מצביע.

לדוגמא סדרת המספרים [3, 10, 2, 1] מתוארת בתרשים הבא:



תור (queue) הוא סוג של רשימה מקושרת. זהו מבנה נתונים מסוג FIFO (First In First Out),
כלומר אלמנטים נכנסים לתור אחד אחרי השני ונשלפים לפי סדר הכנסתם.

במשימה זו עליכם לממש מערכת לניהול פעולות על תור של מספרים (ניתן להניח שלמים).
המערכת תאפשר את הפעולות הבאות (ראו הסברים ודוגמאות בהמשך):

0. יציאה מהתפריט הנוכחי
1. הוספת איבר לתור
2. הוצאת איבר מהתור
3. הדפסת התור
4. הדפסת האיבר המקסימלי בתור
5. הדפסת האיבר המינימלי בתור
6. הדפסת מיקום איבר בתור (כלומר אינדקס)
7. ניקוי התור
8. הדפסה חוזרת של התפריט הנוכחי

לדוגמא, אם ביצענו פעולת הכנסה של האיבר 10, לאחר מכן הכנסנו 20, לאחר מכן הכנסנו 30. אם נקרא עכשיו לפעולת הוצאת איבר נקבל 10 והתור כבר לא יכיל איבר זה. אם נקרא לעוד פעולת הוצאת איבר, נקבל 20 והתור יכיל כעת רק את האיבר 30.

הסברים:
המערכת מתחילה מהדפסת התפריט הבא:

Please select your choice:

0. Exit
1. Add item to the queue
2. Remove item from the queue
3. Print queue
4. Print the maximum item in the queue
5. Print the minimum item in the queue
6. Print index of given item
7. Clear queue
8. Print the menu

במקרה של קלט שאינו מתאים לאפשרויות בתפריט, תודפס ההודעה:

Error: Unrecognized choice

במקרה של פעולה שאינה חוקית על התור (לדוגמא מחיקת איבר כשהתור ריק), תודפס ההודעה:

Error: Queue is empty!

לאחר סיום כל פעולה, תודפס ההודעה הבאה:

Please select your next choice (select 8 for complete menu)

עליכם לוודא שכל הקצאת זיכרון שביצעתם הצליחה.

במידה וההקצאה נכשלה, עליכם להדפיס את ההודעה הבאה:

Error: Insufficient Memory

ולסיים את המשימה הנוכחית.

כמו כן, עליכם לוודא שבסיום התוכנית כל מקום בזיכרון שהקצתם במהלך הריצה משוחרר.

ראו הסברים ודוגמאות בסוף.

שימו לב !

בבדיקה הידנית ייבדק שימוש בהקצאות דינאמיות ומימוש התרגיל בצורה שהוגדרה.
מימוש לא לפי ההוראות עלול לגרום לפסילת התרגיל.

הקפידו על כתיבה נכונה ותואמת ל Coding Style.

חשבו איזה משתנים נדרשים ומה הסוג (type) של כל אחד מהם. השתמשו במבני הברירה המתאימים לכל חלק. להזכירכם בבדיקה הידנית נבדקת קריאות ופשוטות התוכנית, נוסף על נכונותה. יש להקפיד לכתוב הערות, לשים לב להזחות, ולא לשכוח לכתוב את הפרטים שלכם בראש כל קובץ.

הגישו לקבוצת התרגול שלכם (מדעי המחשב – 01, מתמטיקה – 02) בלבד.

בהצלחה!



דוגמאות (הדפסת התוכנית בשחור, הקלט מהמשתמש באדום):

```
Please enter your input:
1
Please enter how many chars to allocate:
7
Allocated 7 chars
Please enter string to be sorted:
bXAcaY
The string after sorting is: abcAXY
Please enter your input:
1
Please enter how many chars to allocate:
5
Allocated 5 chars
Please enter string to be sorted:
CVYH
The string after sorting is: CHVY
Please enter your input:
1
Please enter how many chars to allocate:
7
Allocated 7 chars
Please enter string to be sorted:
abcXYZ
The string is already sorted
Please enter your input:
2
Please enter how many chars to allocate to the two strings:
13 4
Allocated 13 chars and 4 chars
Please enter two arrays of chars:
abcbabcbabc abc
abcbabcbabc is a concatenation of abc 4 times
Please enter your input:
2
Please enter how many chars to allocate to the two strings:
13 4
Allocated 13 chars and 4 chars
Please enter two arrays of chars:
abcbccabab abc
No concatenation found
Please enter your input:
3
Please enter how many chars to allocate:
7
Allocated 7 chars
Please enter your string:
abcbc
The shortest prefix building the string is: abc of length 3
Please enter your input:
3
```

Please enter how many chars to allocate:

6

Allocated 6 chars

Please enter your string:

ab cab

The shortest prefix building the string is: ab cab of length 5

Please enter your input:

4

Please enter how many chars to allocate:

80

Allocated 80 chars

Please enter your string:

Sometimes the questions are complicated and the answers are simple. - Dr. Seuss

Please enter how many words to reverse:

3

The string after reverse is: questions the Sometimes are complicated and the answers are simple. - Dr. Seuss

Please enter your input:

5

Please enter list of names:

robert

vida

vivian

colin

alana

zoe

megan

addy

iris

QUIT

There are 9 names:

addy

Alana

colin

Iris

Megan

Robert

Vida

Vivian

Zoe

Please enter your input:

6

Please select your choice:

0. Exit
1. Add item to the queue
2. Remove item from the queue
3. Print queue
4. Print the maximum item in the queue
5. Print the minimum item in the queue
6. Print index of given item
7. Clear queue
8. Print the menu

1

Enter item value to add

15

Item 15 added


```
Please select your next choice (select 8 for complete menu)
1
Enter item value to add
20
Item 20 added
Please select your next choice (select 8 for complete menu)
1
Enter item value to add
35
Item 35 added
Please select your next choice (select 8 for complete menu)
2
Item 15 was removed
Please select your next choice (select 8 for complete menu)
3
Queue items are: 20 35
Please select your next choice (select 8 for complete menu)
4
Maximum item in queue is 35
Please select your next choice (select 8 for complete menu)
5
Minimum item in queue is 20
Please select your next choice (select 8 for complete menu)
7
Queue is clear
Please select your next choice (select 8 for complete menu)
2
Error. Queue is empty!
Please select your next choice (select 8 for complete menu)
3
Error. Queue is empty!
Please select your next choice (select 8 for complete menu)
1
Enter item value to add
10
Item 10 added
Please select your next choice (select 8 for complete menu)
1
Enter item value to add
20
Item 20 added
Please select your next choice (select 8 for complete menu)
1
Enter item value to add
30
Item 30 added
Please select your next choice (select 8 for complete menu)
6
Please enter the item you would like to know its index
30
Item 20 index is 2
Please select your next choice (select 8 for complete menu)
6
Please enter the item you would like to know its index
40
Error: no such item!
Please select your next choice (select 8 for complete menu)
0
Please enter your input:
0
```