

מבוא למדעי המחשב מ'ח' (234114/7), סמסטר חורף 2020-21

תרגיל בית 5

מועד אחרון להגשה: 27.01.2021 23:59

המתרגלת האחראית על תרגיל זה: ירון חי

- שעות קבלה מוגדרות עבור התרגיל: ימי שני ה-11 בינואר וה-18 בינואר, בשעות 10:30-11:30
- שעות קבלה יתקיימו בקישור המופיע באתר הקורס, לנוחיותכם הקישור הוא:
- דואר אלקטרוני של המתרגל: aron.hay@cs.technion.ac.il

הנחיות כלליות

- הגשה **בבודדים**. עליכם לכתוב את הפתרונות לבד ולהגיש ביחידים.
- קראו את השאלות בעיון לפני שתתחילו בפתרון.
- הקפידו לתעד את הקוד שלכם בהערות באנגלית.
- מלבד מילואים, לא יתקבלו תרגילים אחרי מועד הגשה. הגשה באיחור לאחר מועד הגשה נחשבת כאי-הגשה.

מדיניות ערעורים, הארכות ובדיקה

- כל יום מילואים = יום דחייה. על מנת לקבל את הדחייה, עליכם לשלוח באי-מייל, עותק של האישור המראה שהייתם במילואים (טופס 3010). אם האישור יגיע אליכם בתאריך מאוחר, יש להודיע על כך למתרגל האחראי על התרגיל לפני תאריך הגשת התרגיל.
- ערעורים ניתן להגיש עד שבוע לאחר קבלת הציון.
- לא ניתן לערער על תוצאות הבדיקה האוטומטית.
- **שימו לב! הבדיקה הינה אוטומטית, ולכן הקפידו להדפיס בדיוק בפורמט שהתבקשתם**

(פונקציות הדפסה סופקו עם התרגיל) ובדיקו עם DiffMerge את הפלט שלכם מול הפלט של

הדוגמאות שקיבלתם.

- השתמשו ב-redirection כדי להפנות את הפלט לקובץ טקסט.
- וודאו את האותיות הגדולות והקטנות לפי הדוגמאות וההסברים בתרגיל.
- אין להדפיס רווחים שלא התבקשתם להדפיס (בתחילת שורה או בסופה).
- בכל סוף שורה יש להדפיס תו ירידת שורה, גם בשורה האחרונה.
- השתמשו באתר הבדיקה העצמית.

ספריות מותרות לשימוש

בתרגיל זה מותר להשתמש בפונקציות מהספריות הבאות **בלבד**, שנלמדו בהרצאות ובתרגולים, אלא אם צוין אחרת:

- `stdio.h`
- `stdbool.h`
- `stdlib.h`
- `string.h` רק בפונקציות `strlen` ו-`strcpy`

אופן ההגשה

- ההגשה הינה אלקטרונית ו**בבודדים** דרך אתר הקורס. קובץ ההגשה יהיה מסוג **zip** (ולא אף פורמט אחר) ויכיל בתוכו את הקבצים הבאים בלבד, ללא כל תיקיות:
 - קובץ **students.txt** עם מספר תעודת הזהות שלך וכתובת האי-מייל שלך.
 - קובץ פתרון **hw5q1.c** לשאלה 1
 - קובץ פתרון **hw5q2.c** לשאלה 2
- **חובה לשמור את קוד אישור ההגשה שמקבלים מהמערכת לאחר שמגישים, עד לסיום הקורס.**
- יש להקפיד להגיש את כל הקבצים בדיוק עם השמות שמופיעים לעיל. הגשה שלא תעמוד בתנאי זה **לא תתקבל ע"י המערכת!** אם המערכת לא מקבלת את התרגיל שלכם, חפשו את הפתרון לבעיה באתר הקורס תחת הכפתור FAQ.

שאלות לגבי התרגיל או פניות בנושא התרגיל

- לכל פניה או שאלה בנושא התרגיל יש לשלוח מייל אל המתרגל האחראי על התרגיל, באופן הבא:
1. בנושא המייל יש לציין את הנושא (שאלה לגבי התרגיל או פנייה אישית), ובנוסף לכך יש לציין בכותרת **stud#234114#hw5** או **stud#234117#hw5**.
 2. בפניות אישיות יש לציין את תעודת הזהות שלכם, ואת שמכם.
 3. בשאלות בנושא התרגיל:
 - a. יש לבדוק קודם האם יצאה הבהרה בנושא בקישור, לפני משלוח השאלה:
<https://docs.google.com/document/d/1SyFvKxFO5Sw1xuerVMJJ1YTj1KvonH-I0VfKCl63iX4/edit?usp=sharing>
 - b. יש למספר את השאלות שלכם.
 - c. בתחילת כל שאלה יש לציין באיזה שאלה מדובר.
 - d. לא מובטח כי תתקבל תשובה לשאלה שתשלח בסוף יום ההגשה.
 4. יש להמתין בסבלנות לתשובה.
 5. פניות שיתקבלו שלא לפי ההנחיות האלה, לא יענו.

מיונים (שאלה 1)

בשאלה זו נמין מערך של מחרוזות לפי כללי השוואה שונים. בכל סעיפי השאלה יש להניח תקינות של הקלט.

סעיף א

כתבו פונקציה שחתימתה:

```
int compareStr(const char* str1,
               const char* str2,
               int rule);
```

הפונקציה מקבלת שתי מחרוזות str1 ו-str2 ומס' שלם rule.

הפונקציה מחזירה:

1. מספר חיובי כלשהו אם str1 גדולה מ-str2.
2. מספר שלילי כלשהו אם str2 גדולה מ-str1.
3. אפס אם str1 שווה ל-str2 (או אם rule לא תקין).

המשמעות של "גדולה מ-" ושל "שווה ל-" משתנה בהתאם לערך rule, שמקבל אחד משני הקבועים:

1. LEX שערך 1
2. LET_DIVER שערך 2

שיש להגדיר בעזרת define.

עבור rule = LEX

עבור שתי מחרוזות s1 ו-s2 נאמר ש-s1 גדולה מ-s2 במידה והיא גדולה ממנה על פי הסדר הלקסיקוגרפי. **ההשוואה לא מבדילה בין אותיות קטנות לגדולות.** אם אחד התווים אינו אות אנגלית, התו בעל ערך ה-ASCII הגדול יותר, ייחשב גדול יותר.

דוגמאות:

1. המחרוזת "AB" גדולה מהמחרוזת "AA", וכן "AAA" גדולה מ-"AA".
2. "aa" ו-"AA" שוות.
3. "?" גדולה מ-"!" מכיוון שערך ה-ASCII של סימן שאלה גדול מערכו של סימן קריאה.

עבור rule = LET_DIVER (קיצור של letters diversity):

עבור שתי מחרוזות s1 ו-s2 נאמר ש-s1 גדולה מ-s2 במידה וכמות סוגי האותיות מהאלפבית האנגלי הנמצאת ב-s1 גדולה מהכמות הנמצאת ב-s2. **ההשוואה לא מבדילה בין אותיות קטנות לגדולות.**

דוגמאות:

1. המחזורת "ab" גדולה מהמחזורת "CCCCcccccCC", מכיוון שבמחזורת הראשונה מופיעות שתי אותיות (a ו-b) מהאלפבית האנגלי, ובמחזורת השנייה מופיעה בסך הכל אות אחת (c).
2. "aaabbbb" ו-"CCCddd" שוות.

סעיף ב

ממשו את הפונקציה:

```
void sortString(
    const char* str_arr[],
    int n,
    int rule);
```

המקבלת מערך מחזורות str_arr באורך n, וממיינת אותו בהתאם לכלל ההשוואה rule בין המחזורות (יש להיעזר בפונקציה שהגדרתם בסעיף הקודם). כלומר, אם מחזורת s1 גדולה ממחזורת s2, כפי שמוגדר על פי rule, היא צריכה להופיע אחריה במערך הממויין. בנוסף, המיין צריך להיות יציב. כלומר, בהינתן שתי מחזורות s1 ו-s2 המקיימות שבמערך המקורי s1 ממוקמת אחרי s2, מתקיים שגם במערך הממויין s1 ממוקמת אחרי s2. דוגמאות מובאות בסעיף הבא.

סעיף ג

כתבו את פונקציית ה-main שקולטת מהמשתמש את מערך המחזורות ומדפיסה למסך את המחזורות באופן ממויין. הניחו שאורך מחזורת אינו עולה על 21 תווים (לא כולל ה-null terminator).

מהלך התוכנית הראשית יהיה באופן הבא:

1. תודפס למסך הודעה המבקשת מהמשתמש להכניס את מספר המחזורות n בעזרת הפונקציה הנתונה:

```
void printNumStringsInputMessage();
```

2. ייקלט מספר שלם n מהמשתמש. לא ניתן להניח דבר על ערכו של n מלבד הוא שהוא ערך תקין (חיובי ואינו 0).

3. תודפס הודעה למשתמש המבקשת ממנו להכניס n מחזורות בעזרת הפונקציה:

```
void printStringsInputMessage(int n);
```

4. תיקלטנה n מחזורות (המופרדות לפי תווים לבנים).

5. תודפס הודעה המבקשת מהמשתמש להכניס את כלל ההשוואה בעזרת הפונקציה:

```
void printRuleOfComparisonInputMessage();
```

6. ייקלט מספר שלם rule שהוא כלל ההשוואה מהמשתמש. הניחו שכלל תקין.

7. יודפס מערך המחרוזות הממויין למסך בעזרת הפונקציה:

```
void printSortedStrings(const char *str_arr[], int n);
```

אם במהלך התכנית ישנה שגיאה בהקצאת זיכרון, תודפס הודעה למסך באמצעות פונקציית ההדפסה:

```
void printAllocationError();
```

והתכנית תסתיים.

דוגמות הרצה

```
Please enter the number of strings:
3
Please enter the 3 strings:
bc
ab
AB
Please enter the rule of comparison between strings.
1: Lexicographic order.
2: By the diversity of letters.
1
The sorted strings are:
ab
AB
Bc
```

```
Please enter the number of strings:
5
Please enter the 5 strings:
abcd
adf
nnnnnnnnnnMMMMMM
gh
11kk%$#
Please enter the rule of comparison between strings.
1: Lexicographic order.
2: By the diversity of letters.
2
The sorted strings are:
11kk%$#
nnnnnnnnnnMMMMMM
gh
adf
abcd
```

Backtracking (שאלה 2)

כתבו תכנית הקולטת את הערכים הבאים:

1. מערך לא ריק של ערכי מטבעות שלמים וחיוביים coins
2. את אורך המערך n
3. מספר שלם וחיובי k
4. מספר שלם וחיובי sum

התכנית תדפיס את מספר הפריטות השונות האפשריות של sum ל-k מטבעות לכל היותר. ניתן להניח שערכי המטבעות שונים זה מזה, ושאינן הגבלה על מספר ההופעות של מטבע מסוים בפריטה. יש לפתור את שאלה זו באמצעות backtracking.

שימו לב: פריטות A, B נחשבות שונות אם "מ קיים מטבע שמס' הופעותיו ב-A שונה ממס' הופעותיו ב-B.

מהלך התוכנית הראשית יהיה באופן הבא:

1. תודפס למסך הודעה המבקשת מהמשתמש להכניס את מספר המטבעות n בעזרת הפונקציה הנתונה:

```
void printNumCoinsInputMessage();
```

2. ייקלט מספר שלם n מהמשתמש. לא ניתן להניח דבר על ערכו של n מלבד הוא שהוא ערך תקין (חיובי ואינו 0).

3. תודפס למסך הודעה המבקשת מהמשתמש להכניס את הערכים של n המטבעות בעזרת הפונקציה הנתונה:

```
void printCoinValsInputMessage();
```

4. יקלטו n מספרים שלמים וחיוביים מהמשתמש. הניחו תקינות קלט על כל המספרים.
5. תודפס למסך הודעה המבקשת מהמשתמש להכניס את הסכום sum בעזרת הפונקציה הנתונה:

```
void printSumInputMessage();
```

6. ייקלט מספר שלם sum מהמשתמש. לא ניתן להניח דבר על ערכו של sum מלבד הוא שהוא ערך תקין.

7. תודפס למסך הודעה המבקשת מהמשתמש להכניס את מספר המטבעות המקסימלי k בפריטה כלשהי בעזרת הפונקציה הנתונה:

```
void printLimitInputMessage();
```

8. ייקלט מספר שלם k מהמשתמש. לא ניתן להניח דבר על ערכו של k מלבד הוא שהוא ערך תקין.

9. יודפס מספר הפריטות השונות האפשריות של sum ל- k מטבעות לכל היותר, בעזרת הפונקציה הנתונה:

```
void printAnswerMessage(int n);
```

אם במהלך התכנית ישנה שגיאה בהקצאת זיכרון, תודפס הודעה למסך באמצעות פונקציית ההדפסה:

```
void printAllocationError();
```

והתכנית תסתיים.

הערות

1. ניתן להניח שהקלט תקין ואין צורך לבדוק זאת. כל הערכים נמצאים בטווח הייצוג של `int`.

דוגמאות הרצה

Please enter coins num: 2 Please enter coins vals: 1 2 Please enter the requested sum: 2 Please enter coins limit: 1 There are 1 possible partitions.	Please enter coins num: 4 Please enter coins vals: 2 5 3 7 Please enter the requested sum: 10 Please enter the coins limit: 2 There are 2 possible partitions.
---	--

בהצלחה!