

תרגיל בית 0 – ריענון שפת C		
מועד ההגשה:	יום ה', 04/11/2021, בשעה 23:55	
האחראי על התרגיל:	פבל ליפשיץ	pavel@ee.technion.ac.il

שאלה 1 – שכיחות ספרות

בהינתן מחרוזת  $s$ , המכילה גם אותיות וגם ספרות. יש למצוא את השכיחות של כל אחת מהספרות 0 עד 9 במחרוזת הנתונה.

פורמט קלט

הקלט הינו מחרוזת  $s$ .

מגבלות קלט

נתון כי אורך המחרוזת הוא 1 לפחות, ולא יותר מ-1000.

$$1 \leq \text{len}(s) \leq 1000$$

כמו כן, מובטח שמרכיבי המחרוזת הם ספרות ואותיות באנגלית בלבד.

פורמט הפלט

הדפיסו 10 מספרים שלמים מופרדים ברווח, המהווים את השכיחות של הספרות 0 עד 9 במחרוזת שנקלטה.

דוגמא 1קלט

a11472o5t6

פלט צפוי

0 2 1 0 1 1 1 1 0 0

הסבר לדוגמא 1

במחרוזת הנתונה:

הספרה 1 מופיעה 2 פעמים.

הספרות 2,4,5,6 ו-7 מופיעות פעם אחת.

הספרות הנותרות 0,3,8 ו-9 אינן מופיעות כלל.

דוגמא 2קלט

lw4n88j12n1

פלט צפוי

0 2 1 0 1 0 0 0 2 0

דוגמא 3קלט

lv88886l256338ar0ekk

פלט צפוי

1 1 1 2 0 1 2 0 5 0

ממשו את הקוד בקובץ הבא בקובץ בשם digit\_freq.c. תבנית הקובץ נתונה גם באתר הקורס במודל.

כמו כן, נתונים לכם באתר הקורס קובץ הקלט digit\_freq.in ופורמט הפלט המצופה digit\_freq.out. ודאו שהפלט שלכם זהה לפלט המצופה ע"י פקודת diff (שימו לב במיוחד לירידת שורה בסוף הפלט).

```
// 123456789
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int main() {

    /* Enter your code here. Read input from STDIN. Print output to STDOUT
    */
    return 0;
}
```

שאלה 2 – צירופים

מאיה זכתה במכונות משחקים בפארק אטרקציות בכרטיסים בשווי של \$43.94.

בחנות הצעצועים של הפארק נמכרים הצעצועים הבאים:

סירה \$8.04	פיל \$3.16	כדור \$2.16	יו-יו \$1.22
רכבת \$6.71	דובי \$4.89	מכונת מרוץ \$7.13	בובה \$2.75
קוביות \$2.31	קסילופון \$7.11	כלב \$4.57	ברווז גומי \$1.85
משאית \$6.21	טנק \$6.45	דילגית \$1.46	טרקטור \$5.97
משרוקית \$0.98	דמקה \$4.77	מכונת \$5.18	אוורון \$6.47
שבשבת \$0.87			

מאיה מתכננת לקנות צעצועים ולתרום אותם לילדים נזקקים. כיוון שניתן להשתמש בכרטיסים אלו אך ורק בחנות של הפארק, היא מעוניינת להשתמש בכל הכסף (עד הסנט האחרון).

כתבו תוכנית בשם `at_the_park_perm.c` שתעזור למאיה להרכיב רשימה של צעצועים, כך שתבזבז את כל הכסף בו זכתה.

עצרו את ריצת התוכנית לאחר שנמצאה רשימה בת 13 פריטים.

שאלה 2 – חלק א'

- האם ישנה רק רשימה אחת כזו? מהי?
- האם אפשרי למצוא רשימה בת 14 פריטים? בת 15 פריטים? מדוע?
- ענו בכתב – כיצד תשתנה התוכנית אם מותר היה לרכוש יותר מיחידה אחת מכל צעצוע?

שאלה 2 – חלק ב'

במידה וביצעתם קודם הדפסות למסך, הכניסו שורות אלו להערה.

שנו את התוכנית כך שתעצור אחרי שנמצאו 100, 1000, ו-10000 פתרונות.

קמפלו מחדש את התוכנית.

על גבי המכונה הוירטואלית של הקורס הריצו את התוכנית ומדדו את זמן הריצה שלה באמצעות הפקודה `time`, לדוגמא:

```
# time ./at_the_park_perm
```

חזרו על המדידה 10 פעמים עבור הגדלים –

- האם זמני הריצה היו זהים בכל פעם? תוכלו לשאר מדוע?

ב. כיצד לדעתכם ישתנו זמני הריצה כפונקציה של מספר הצעצועים בחנות?

את התשובות לכל השאלות יש לענות **באנגלית** בקובץ טקסט בשם answers.txt.

רמז לפתרון אפשרי (לא היחיד ולא הכיוון ההכרחי): האם ניתן לפרק את הבעיה לבעיה קטנה יותר? (עם רשימה קצרה יותר)?

שאלה 3 – פונקציות וריאדיות (לימוד עצמי)

פונקציות וריאדיות הן פונקציות המקבלות מספר משתנה של ארגומנטים. בתכנות C, פונקציה וריאדית תורמת לגמישות של התוכנית אותה אתם מפתחים. דוגמא שמוכרת לכם מהעבר היא הפונקציה printf.

הצהרה על פונקציה וריאדית מתחילה בהצהרה על לפחות משתנה אחד, ועושה שימוש ב... (ellipsis) בתור הפרמטר האחרון. לדוגמא:

```
int printf(const char* format, ...);
```

התרגיל זה תבצע לימוד עצמי של מימוש פונקציות וריאדיות. ותממשו 3 פונקציות בשמות m\_min(), m\_max() לחישוב סכום, מינימום, ומקסימום (בהתאמה) של מספר משתנה של ארגומנטים. הארגומנט הראשון שמועבר לפונקציה הוא מספר הארגומנטים, ולאחריו יגיעו הארגומנטים עצמם.

קראו עוד על פונקציות וריאדיות פה:

[https://www.gnu.org/software/libc/manual/html\\_node/Variadic-Functions.html](https://www.gnu.org/software/libc/manual/html_node/Variadic-Functions.html)

הקלט לתוכנית הנתונה לכם הוא מספר חיובי ושלם, של מספר מקרי הבדיקה שיוצרו.

כל מקרה בדיקה בודק את הקוד שלכם על ידי שליחה של 3, 5 ו-10 אלמנטים בהתאמה.

אתם יותר ממוזמנים לבדוק את הקוד שלכם מול קלט שבחרתם.

ה- error log מדפיס את הפרמטרים שהועברו לבדיקה וכן מדפיס את הסכום, והאלמנט המינימלי והמקסימלי.

יודפס "Correct Answer" לכל הרצה תקינה של בדיקה ו-"Wrong Answer" אחרת.

ממשו את הקוד שלכם בתוך התבנית הנתונה. התבנית נתונה לכם כקובץ גם באתר הקורס במודל.

var\_func.c

```
#include <stdarg.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define MIN_ELEMENT 1
#define MAX_ELEMENT 1000000
int m_sum (int count,...) {

}
```

```
int m_min(int count,...) {  
  
}  
  
int m_max(int count,...) {  
  
}  
  
int test_implementations_by_sending_three_elements() {  
    srand(time(NULL));  
  
    int elements[3];  
  
    elements[0] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;  
    elements[1] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;  
    elements[2] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;  
  
    fprintf(stderr, "Sending following three elements:\n");  
    for (int i = 0; i < 3; i++) {  
        fprintf(stderr, "%d\n", elements[i]);  
    }  
  
    int elements_sum = m_sum(3, elements[0], elements[1], elements[2]);  
    int minimum_element = m_min(3, elements[0], elements[1], elements[2]);  
    int maximum_element = m_max(3, elements[0], elements[1], elements[2]);  
  
    fprintf(stderr, "Your output is:\n");  
    fprintf(stderr, "Elements sum is %d\n", elements_sum);  
    fprintf(stderr, "Minimum element is %d\n", minimum_element);  
    fprintf(stderr, "Maximum element is %d\n\n", maximum_element);  
  
    int expected_elements_sum = 0;  
    for (int i = 0; i < 3; i++) {  
        if (elements[i] < minimum_element) {  
            return 0;  
        }  
  
        if (elements[i] > maximum_element) {  
            return 0;  
        }  
  
        expected_elements_sum += elements[i];  
    }  
}
```

```
    return elements_sum == expected_elements_sum;
}

int test_implementations_by_sending_five_elements() {
    srand(time(NULL));

    int elements[5];

    elements[0] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[1] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[2] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[3] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[4] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;

    fprintf(stderr, "Sending following five elements:\n");
    for (int i = 0; i < 5; i++) {
        fprintf(stderr, "%d\n", elements[i]);
    }

    int elements_sum = m_sum(5, elements[0], elements[1], elements[2], elements[3], elements[4]);
    int minimum_element = m_min(5, elements[0], elements[1], elements[2], elements[3], elements[4]);
    int maximum_element = m_max(5, elements[0], elements[1], elements[2], elements[3], elements[4]);

    fprintf(stderr, "Your output is:\n");
    fprintf(stderr, "Elements sum is %d\n", elements_sum);
    fprintf(stderr, "Minimum element is %d\n", minimum_element);
    fprintf(stderr, "Maximum element is %d\n\n", maximum_element);

    int expected_elements_sum = 0;
    for (int i = 0; i < 5; i++) {
        if (elements[i] < minimum_element) {
            return 0;
        }

        if (elements[i] > maximum_element) {
            return 0;
        }

        expected_elements_sum += elements[i];
    }
}
```

```
}

return elements_sum == expected_elements_sum;
}

int test_implementations_by_sending_ten_elements() {
    srand(time(NULL));

    int elements[10];

    elements[0] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[1] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[2] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[3] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[4] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[5] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[6] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[7] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[8] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[9] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;

    fprintf(stderr, "Sending following ten elements:\n");
    for (int i = 0; i < 10; i++) {
        fprintf(stderr, "%d\n", elements[i]);
    }

    int elements_sum = m_sum(10, elements[0], elements[1], elements[2], elements[3], elements[4],
                           elements[5], elements[6], elements[7], elements[8], elements[9]);
    int minimum_element = m_min(10, elements[0], elements[1], elements[2], elements[3], elements[4],
                                elements[5], elements[6], elements[7], elements[8], elements[9]);
    int maximum_element = m_max(10, elements[0], elements[1], elements[2], elements[3], elements[4],
                                elements[5], elements[6], elements[7], elements[8], elements[9]);

    fprintf(stderr, "Your output is:\n");
    fprintf(stderr, "Elements sum is %d\n", elements_sum);
    fprintf(stderr, "Minimum element is %d\n", minimum_element);
    fprintf(stderr, "Maximum element is %d\n\n", maximum_element);
}
```



```
int expected_elements_sum = 0;
for (int i = 0; i < 10; i++) {
    if (elements[i] < minimum_element) {
        return 0;
    }

    if (elements[i] > maximum_element) {
        return 0;
    }

    expected_elements_sum += elements[i];
}

return elements_sum == expected_elements_sum;
}

int main ()
{
    int number_of_test_cases;
    scanf("%d", &number_of_test_cases);

    while (number_of_test_cases--) {
        if (test_implementations_by_sending_three_elements()) {
            printf("Correct Answer\n");
        } else {
            printf("Wrong Answer\n");
        }

        if (test_implementations_by_sending_five_elements()) {
            printf("Correct Answer\n");
        } else {
            printf("Wrong Answer\n");
        }

        if (test_implementations_by_sending_ten_elements()) {
            printf("Correct Answer\n");
        } else {
            printf("Wrong Answer\n");
        }
    }

    return 0;
}
```

}

הוראות הגשה:

1. באתר הקורס במודל מופיעות הנחיות ההגשה לתרגילי הבית. עברו עליהם לפני ההגשה וודאו שאתם עומדים בהן במלואן.

2. יש להגיש בנוסף קובץ בשם readme המכיל שתי שורות של המגישים:

```
<id1> student_1_mail@campus.technion.ac.il <full name 1>
<id2> student_2_mail@campus.technion.ac.il <full name 2>
```

לדוגמה עבור הסטודנטים אבי כהן ומשה לוי:

123456789 avi.cohen@campus.technion.ac.il Avi Cohen

987654321 moshe.levy@campus.technion.ac.il Moshe Levy

את כל הקבצים יש לארוז בקובץ בשם <id>.zip כאשר <id> הוא מספר תעודת הזהות של אחד המגישים.

אין לכלול בקובץ ה-zip תתי תיקיות.

3. שאלות בנוגע לתרגיל יש להפנות לפורום התרגיל ב-moodle בלבד – ניתן לשלוח שאלות במייל **למתרגל האחראי על התרגיל בלבד**, ורק במידה והשאלה מכילה פתרון חלקי.

4. סיכום מפרט התרגיל:

סעיף	תיאור
נושא התרגיל	ריענון שפת C – לולאות, מחרוזות, מערכים, מצביעים
תאריך ההגשה	יום ה', 04/11/2021, בשעה 23:55
האחראי על התרגיל	פבל ליפשיץ pavel@ee.technion.ac.il
קבצי הקוד הנתונים	digit_freq.c var_func.c
קבצי הקלט והפלט הנתונים	digit_freq.in digit_freq.out
הקבצים שיש להגיש	var_func.c digit_freq.c at_the_park_perm.c answers.txt readme

**בהצלחה!**