

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

שאלה 1 – הדפסת תבנית בעזרת לולאות

הדפיסו תבנית של מספרים מ-1 עד n כפי שמוצג להלן. יש הפרדה של רווח בודד בין מספר למספר.

```

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

```

פורמט הקלט

הקלט הינו מספר שלם ויחיד n .

מגבלות על הקלט

$$1 \leq n \leq 100$$

דוגמא 1

עבור הקלט

2

יוצג הפלט

```

2 2 2
2 1 2
2 2 2

```

דוגמא 2

עבור הקלט

5

יוצג הפלט

```

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

```

```

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

```

דוגמא 3

עבור הקלט

7

יוצג הפלט

```

7 7 7 7 7 7 7 7 7 7 7 7 7
7 6 6 6 6 6 6 6 6 6 6 6 7
7 6 5 5 5 5 5 5 5 5 5 6 7
7 6 5 4 4 4 4 4 4 4 5 6 7
7 6 5 4 3 3 3 3 3 4 5 6 7
7 6 5 4 3 2 2 2 3 4 5 6 7
7 6 5 4 3 2 1 2 3 4 5 6 7
7 6 5 4 3 2 2 2 3 4 5 6 7
7 6 5 4 3 3 3 3 3 4 5 6 7
7 6 5 4 4 4 4 4 4 4 5 6 7
7 6 5 5 5 5 5 5 5 5 6 7
7 6 6 6 6 6 6 6 6 6 6 7
7 7 7 7 7 7 7 7 7 7 7 7

```

ממשו את התוכנית שלכם בקובץ הבא :

print_pattern.c

```

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

int main()
{
    int n;
    scanf("%d", &n);
    // Complete the code to print the pattern.
    return 0;
}

```

שאלה 2 – 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

1v888861256338ar0ekk

פלט צפוי

1 1 1 2 0 1 2 0 5 0

ממשו את הקוד בקובץ הבא

digit_freq.c

```
// 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 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 sum(), min() ו-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

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:

var_func.c

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

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

}

int min(int count,...) {
```

```
}

int 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 = sum(3, elements[0], elements[1], elements[2]);
    int minimum_element = min(3, elements[0], elements[1], elements[2]);
    int maximum_element = 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 = sum(5, elements[0], elements[1], elements[2], elements[3], elements[4]);
    int minimum_element = min(5, elements[0], elements[1], elements[2], elements[3], elements[4]);
    int maximum_element = 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 = 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 = 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 = 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;
}
```

תוכלו לקמפל ולהריץ את התוכניות שלכם על גבי המכונה הווירטואלית, כפי שיוצג בסדנא הראשונה

הוראות הגשה:

- עברו היטב על הוראות ההגשה של תרגילי הבית המופיעים באתר טרם ההגשה! ודאו כי התכנית שלכם עומדת בדרישות הבאות:
 - התכנית קריאה וברורה
 - התכנית מתועדת היטב לפי דרישות התייעוד המופיעות באתר
- יש להגיש בנוסף קובץ בשם readme המכיל שתי שורות של המגשים:

```
<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

- 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.
- סיכום מפרט התרגיל:

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

בהצלחה!