

# מת"מ תרגיל 1

מגישים: זואי עדן, איתי שפיגלמן

14 בדצמבר 2020

## 2 חלק יבש

### סעיף א

6 שגיאות תכנות:

1. ה-`assert` הראשון יעבור אם המשתנה `NULL` ולא להפך
2. ב-`malloc` לא מבקשים מספיק מקום עבור ה-`null terminator` בסוף המערך `out`, לכן השימוש ב-`strcpy` לאחר מכן מסוכן
3. מוחזר מצביע לתחילת המילה האחרונה בתוך רצף המילים שהכפלנו במקום לתחילת הרצף
4. קטן שווה במקום קטן ממש בלולאה גורם לשכפול פעם אחת יותר מידי שזה באג לוגי וגם גישה מסוכנת למקום בזכרון שלא ביקשנו.
5. גם תחת ההנחה שהקלט תקין, עבור מקרה בו `malloc` יכשל לא יוחזר `NULL` כמו בדרישה. עם זאת אם הקובץ יקומפל לא ב"דיבאג" (הגדרת `NDEBUG`) מעבר לכך שהפונקציה לא תחזיר `NULL` כנראה יהיה `Segmentation Fault`.
6. ה-`include` מחפש קודם כל בתיקייה של אותו הקובץ היות ומשתמשים ב-`"",` זה עובד אבל כדאי לעשות `#include <stdlib.h>` במקרה של קבצי `header` סטנדרטים שלא אנחנו כתבנו ולא נמצאים בתיקיית הקובץ. אם היו קבצי `header` עם שם זהה בתיקייה אז ה-`preprocessor` היה מצרף את הקבצים הלא נכונים.

4 שגיאות קונבנציה:

1. המשתנה `LEN` צריך להיות באותיות קטנות
2. בתוך ה-`for` צריך הזחה
3. שם הפונקציה לא מנוסח כפעולה, היה צריך להיות בסגנון `duplicateString`
4. משתנה `s` לא קיצור ידוע או מילה ברורה

## סעיף ב

```
#include <stdlib.h>
#include <string.h>
#include <assert.h>

char* stringDuplicator(char* string_to_duplicate, int times) {
    assert(string_to_duplicate);
    assert(times > 0);
    int len = strlen(string_to_duplicate);

    char* out = malloc(len*times+1);
    assert(out);
    if (out == NULL) {
        return NULL;
    }

    for (int i=0; i<times; i++) {
        strcpy(out + i*len, string_to_duplicate);
    }

    return out;
}
```