

## תרגיל בית 1 - יבש

### חלק א

שגיאות תכנות:

1. assert הראשון נבדק אם המצביע הוא 0 (מצביע ל NULL) וממשיך, מה שיגרום לשגיאה בפונקציה strlen
2. בפקודת malloc לא הוקצה מקום לתו ה-0\ הדרוש בסוף כל מחרוזת.
3. assert לאחר פקודת malloc אינו פונקציה שבודקת לוגיקה אלא הנחות, בנוסף אינו יחזיר NULL במקרה של שגיאה אלא רק יקריס את התוכנית.
4. בלולאת for יש times+1 איטרציות כלומר המחרוזת תועתק פעם אחת יותר מהדרוש.
5. הסדר של הפונקציות הפוך, קודם מקדמים ורק אז מעתיקים, מה שיצור ערכי זבל בתחילת המחרוזת וחריגת זיכרון.
6. בפקודת ה-return המצביע שמוחזר אינו מצביע לתחילת המחרוזת אותה אנחנו רוצים להחזיר.

שגיאות קונבנציה:

1. LEN אמור להיות באותיות קטנות שכן הוא משתנה ולא קבוע.
2. בתוך לולאת for חסר הזחה לשתי השורות.

## חלק ב

גרסה מתוקנת של הפונקציה:

```
#include "stdlib.h"
#include "string.h"
#include "assert.h"

char *stringDuplicator(char *s, int times)
{
    assert(s);
    assert(times > 0);
    int len = strlen(s);
    char *out = malloc(len * times + 1);
    if (out == NULL)
    {
        return NULL;
    }
    for (int i = 0; i < times; i++)
    {
        strcpy(out, s);
        out = out + len;
    }
    return out - (len*times);
}
```