שאלה 5

בקוד יש 2 בעיות עיקריות שהן גישה לערך לא מאותחל ודליפת זיכרון.

1. **ערך לא מאותחל**

בדו"ח valgrind זה מצויין בשורות האלו:

2. ==194697== Conditional jump or move depends on uninitialised value(s)
3. ==194697== at 0x109234: main (hello.c:19)

זה קורה כי אנחנו משתמשים במשתנה string_so_far בלי לאתחל אותו. מקור הבעיה נמצא בשורות האלו :

4. if (string_so_far != (char *)0)
   strcpy(string, string_so_far);

כשאנחנו משתמשים במשתנה הזה בפעם הראשונה, לא איתחלנו אותו והוא מכיל ערכי זבל.

בשביל לתקן את זה נוכל להוסיף שורה char *string_so_far = NULL; בתחילת התוכנית.

1) **דליפת זיכרון**

מתבצע הקצאת זיכרון למשתנה string, לאחר מכן מעתיקים את המילה בקלט לתוך string, ואז שמים את המצביע string_so_far על string. ולא עושים free. באיטרציה הבאה שנעשה השמה לstring_so_far אנחנו נאבד גישה למקום הקודם שהוקצה בזיכרון והוא יאבד, מה שיגרום לדליפת זיכרון.

בנוסף בסוף הריצה string_so_far עדיין מחזיק מצביע לבלוק בזיכרון שהוקצה ואנחנו לא עושים לו free ולכן תהיה דליפת זיכרון.

שאלה 7

השורות המוסמנות בדו"ח helgrind מצביעות על כך שבשורה 16 בתוך פונקציית square בקובץ race.c יש data race. רואים שthread 4 קורא או כותב מכתובת 0x10C018 וזה מוביל לקונפליקט עם כתיבה קודמת של thread 3 לאותה כתובת בזיכרון .
שורות 43 ו-60 מגלות לנו שאותה כתובת זיכרון היא המשתנה accum מה שאומר שהוא המשאב המשותף שבו קורה ה data race.

```
23  ==267417==    by 0x49011FF: pthread_create@@GLIBC_2.34 (pthread_create.c:828)
24  ==267417==    by 0x4853767: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
25  ==267417==    by 0x1092C3: main (race.c:28)
26  ==267417==
27  ==267417== ----------------------------------------------------------------
28  ==267417==
29
30  ==267417== Possible data race during read of size 8 at 0x10C018 by thread #4
31  ==267417== Locks held: none
32  ==267417==    at 0x10920B: square (race.c:16)
33  ==267417==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
34  ==267417==    by 0x4900AC2: start_thread (pthread_create.c:442)
35  ==267417==    by 0x4991A03: clone (clone.S:100)
36  ==267417==
37  ==267417== This conflicts with a previous write of size 8 by thread #3
38  ==267417== Locks held: none
39  ==267417==    at 0x109215: square (race.c:16)
40  ==267417==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
41  ==267417==    by 0x4900AC2: start_thread (pthread_create.c:442)
42  ==267417==    by 0x4991A03: clone (clone.S:100)
43  ==267417==  Address 0x10c018 is 0 bytes inside data symbol "accum"
44  ==267417==
45  ==267417== ----------------------------------------------------------------
46  ==267417==
47  ==267417== Possible data race during write of size 8 at 0x10C018 by thread #4
48  ==267417== Locks held: none
49  ==267417==    at 0x109215: square (race.c:16)
50  ==267417==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
51  ==267417==    by 0x4900AC2: start_thread (pthread_create.c:442)
52  ==267417==    by 0x4991A03: clone (clone.S:100)
53  ==267417==
54  ==267417== This conflicts with a previous write of size 8 by thread #3
55  ==267417== Locks held: none
56  ==267417==    at 0x109215: square (race.c:16)
57  ==267417==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
58  ==267417==    by 0x4900AC2: start_thread (pthread_create.c:442)
59  ==267417==    by 0x4991A03: clone (clone.S:100)
60  ==267417==  Address 0x10c018 is 0 bytes inside data symbol "accum"
61  ==267417==
```