שאלה 7

נשתמש ב– helgrind על מנת למצוא race condition עבור טרדים

```
valgrind --tool=helgrind ./race
```

נקבל קצת מידע לגבי טרדים שנוצרו:

```
==4886== ---Thread-Announcement------------------------------------------
==4886==
==4886== Thread #4 was created
==4886==    at 0x49999F3: clone (clone.S:76)
==4886==    by 0x499A8EE: __clone_internal (clone-internal.c:83)
==4886==    by 0x49086D8: create_thread (pthread_create.c:295)
==4886==    by 0x49091FF: pthread_create@@GLIBC_2.34 (pthread_create.c:828)
==4886==    by 0x4853767: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
==4886==    by 0x1092C3: main (race.c:20)
==4886==
==4886== ---Thread-Announcement------------------------------------------
==4886==
==4886== Thread #3 was created
==4886==    at 0x49999F3: clone (clone.S:76)
==4886==    by 0x499A8EE: __clone_internal (clone-internal.c:83)
==4886==    by 0x49086D8: create_thread (pthread_create.c:295)
==4886==    by 0x49091FF: pthread_create@@GLIBC_2.34 (pthread_create.c:828)
==4886==    by 0x4853767: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
==4886==    by 0x1092C3: main (race.c:20)
==4886==
==4886== ------------------------------------------------------------
```

קיבלנו פעמיים data race:

```
==4886== ------------------------------------------------------------
==4886==
==4886== Possible data race during read of size 8 at 0x10C018 by thread #4
==4886== Locks held: none
==4886==    at 0x10920B: square (race.c:10)
==4886==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
==4886==    by 0x4908AC2: start_thread (pthread_create.c:442)
==4886==    by 0x4999A03: clone (clone.S:100)
==4886==
==4886== This conflicts with a previous write of size 8 by thread #3
==4886== Locks held: none
==4886==    at 0x109215: square (race.c:10)
==4886==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
==4886==    by 0x4908AC2: start_thread (pthread_create.c:442)
==4886==    by 0x4999A03: clone (clone.S:100)
==4886==  Address 0x10c018 is 0 bytes inside data symbol "accum"
==4886==
==4886== ------------------------------------------------------------
```

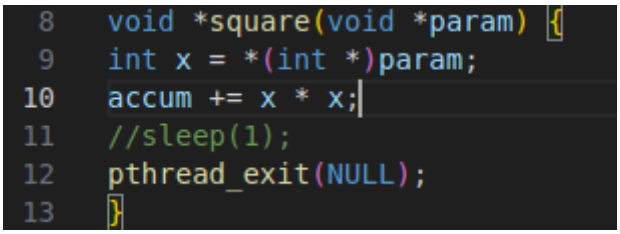
כאן טרד מספר 4 יכול לקרוא בעוד שטרד 3 יכול לכתוב

```
==4886== ------------------------------------------------------------
==4886==
==4886== Possible data race during write of size 8 at 0x10C018 by thread #4
==4886== Locks held: none
==4886==    at 0x109215: square (race.c:10)
==4886==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
==4886==    by 0x4908AC2: start_thread (pthread_create.c:442)
==4886==    by 0x4999A03: clone (clone.S:100)
==4886==
==4886== This conflicts with a previous write of size 8 by thread #3
==4886== Locks held: none
==4886==    at 0x109215: square (race.c:10)
==4886==    by 0x485396A: ??? (in /usr/libexec/valgrind/vgpreload_helgrind-amd64-linux.so)
==4886==    by 0x4908AC2: start_thread (pthread_create.c:442)
==4886==    by 0x4999A03: clone (clone.S:100)
==4886==  Address 0x10c018 is 0 bytes inside data symbol "accum"
==4886==
```

לעומת זאת כאן טרד מספר 4 יכול לכתוב בעוד שטרד מספר 3 יכול לכתוב גם כן.

נשים לב שבשני המקרים אין לנו מנעול (mutex).

שני הבעיות מתקיימות בשורה 10 בפונקציה square:



שניהם יכולים להשפיע בו זמנית על החישוב בפונקציה ושמה נמצאת ה- race condition.