

;[65536]where is allocated? **bss** (data segment) // char globBuf .1

.because we did not initialize the variable

קימפלתי את הקוד והשתמשי ב size (הsize הראשון שרואים בטרמינל).  
לאחר מכן מחקתי את אותה שורה, קימפלתי ושוב השתמשי ב size (השני שרואים  
בטרמינל).  
ניתן לראות שהגודל של **bss** השתנה

```
q1_313465114.c - fwork_313465114 - Visual Studio Code

1 #define BSD_SOURCE
2 #include <stdio.h>
3 #include <stdlib.h>
4
5 // char globBuf[65536]; /* 1. bss */
6 int primes[] = { 2, 3, 5, 7 }; /* 2. data */
7
8 static int
9 square(int x) /* 3. text */
10 {
11     int result; /* 4. stack */
12 }
```

```
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1829 628 10305568 10308025 9d49b9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$ gcc q1_313465114.c
In file included from /usr/include/x86_64-linux-gnu/bits/libc-header-start.h:33:0,
from /usr/include/stdio.h:27,
from q1_313465114.c:2:
/usr/include/features.h:184:3: warning: #warning " BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE" [-Wcpp]
# warning " BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE"
^~~~~~
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1829 628 10240032 10242489 9c49b9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$
```

{ 7 , 5 , 3 , 2 } = []where is allocated? **data** // int primes .2

because we initialized the array

קימפלתי את הקוד והשתמשי ב size (הsize הראשון שרואים בטרמינל).  
לאחר מכן מחקתי את אותה שורה, קימפלתי ושוב השתמשי ב size (השני שרואים  
בטרמינל).  
ניתן לראות שהגודל של **data** השתנה

```
q1_313465114.c - fwork_313465114 - Visual Studio Code

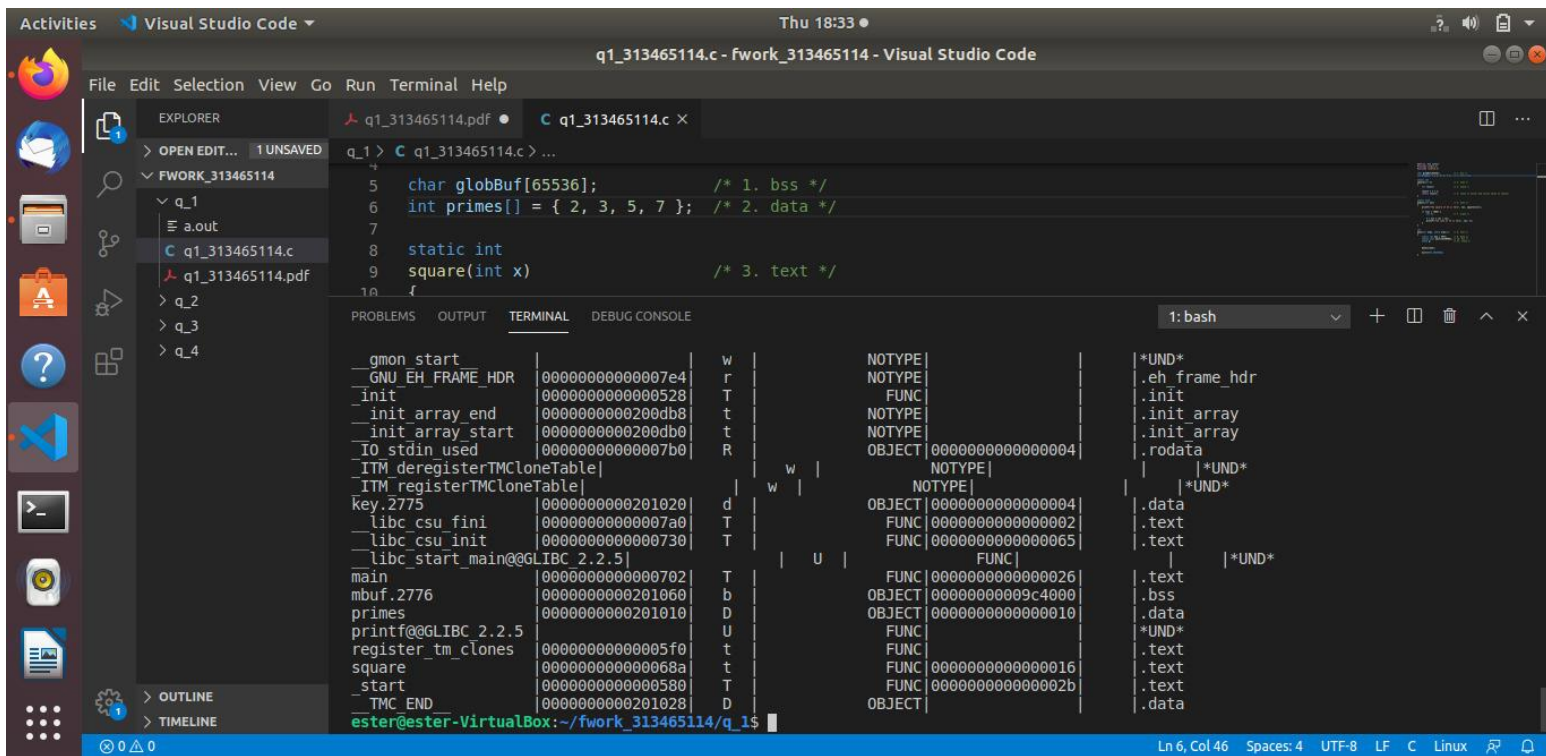
1 #define BSD_SOURCE
2 #include <stdio.h>
3 #include <stdlib.h>
4
5 char globBuf[65536]; /* 1. bss */
6 //int primes[] = { 2, 3, 5, 7 }; /* 2. data */
7
8 static int
9 square(int x) /* 3. text */
10 {
11     int result; /* 4. stack */
12 }
```

```
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1829 628 10305568 10308025 9d49b9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$ gcc q1_313465114.c
In file included from /usr/include/x86_64-linux-gnu/bits/libc-header-start.h:33:0,
from /usr/include/stdio.h:27,
from q1_313465114.c:2:
/usr/include/features.h:184:3: warning: #warning " BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE" [-Wcpp]
# warning " BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE"
^~~~~~
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1829 612 10305568 10308009 9d49a9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$
```

where is allocated? **text** // square(int x) .3

הרצתי את הפקודה nm -f sys a.out

שם כתוב איפה כל שורה בקוד ממוקמת

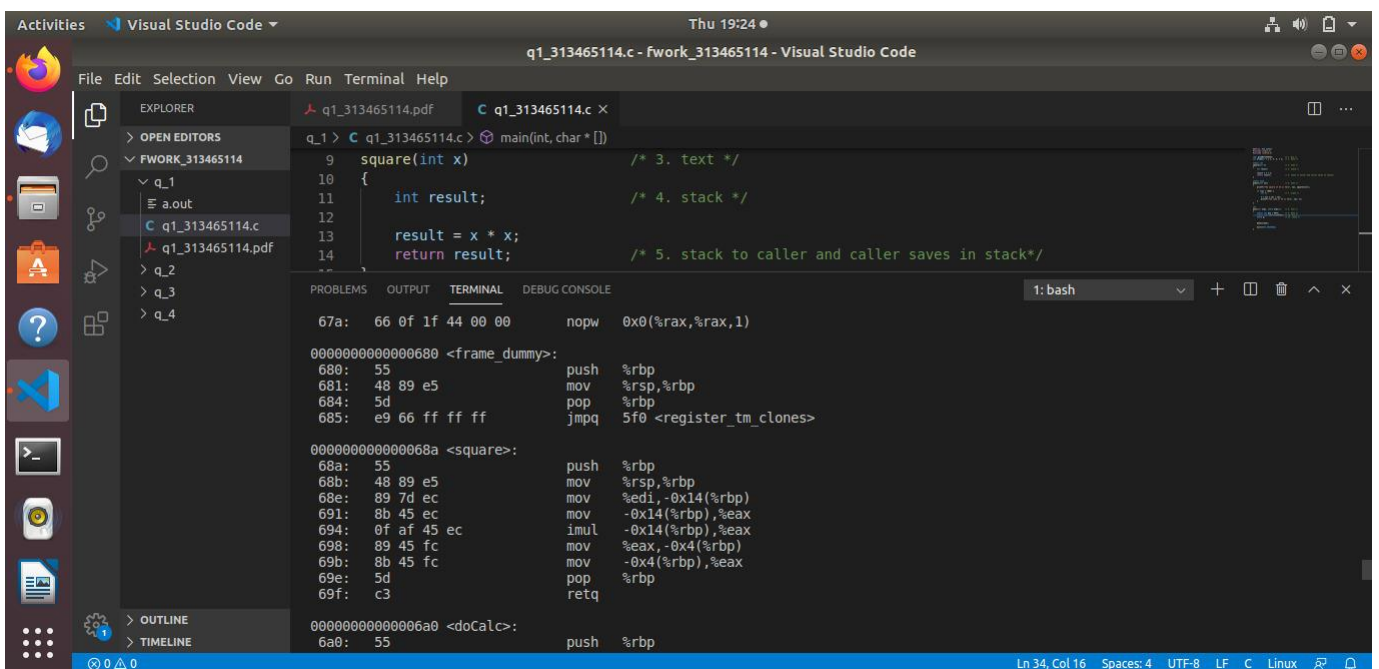


```
nm -f sys a.out
0000000000000000 T _start
0000000000000000 T _TMC_END_
0000000000000000 T _TMC_START_
0000000000000000 T _main
0000000000000000 T _square
0000000000000000 T _register_tm_clones
0000000000000000 T _printf@GLIBC_2.2.5
0000000000000000 T _mbuf.2776
0000000000000000 T _primes
0000000000000000 T _key.2775
0000000000000000 T _libc_start_main@@GLIBC_2.2.5
0000000000000000 T _libc_csu_init
0000000000000000 T _libc_csu_fini
0000000000000000 T _init_array_end
0000000000000000 T _init_array_start
0000000000000000 T _IO_stdin_used
0000000000000000 T _ITM_deregisterTMCloneTable
0000000000000000 T _ITM_registerTMCloneTable
0000000000000000 T _GNU_EH_FRAME_HDR
0000000000000000 T _gmon_start
0000000000000000 T _square
0000000000000000 T _register_tm_clones
0000000000000000 T _printf@GLIBC_2.2.5
0000000000000000 T _mbuf.2776
0000000000000000 T _primes
0000000000000000 T _key.2775
0000000000000000 T _libc_start_main@@GLIBC_2.2.5
0000000000000000 T _libc_csu_init
0000000000000000 T _libc_csu_fini
0000000000000000 T _init_array_end
0000000000000000 T _init_array_start
0000000000000000 T _IO_stdin_used
0000000000000000 T _ITM_deregisterTMCloneTable
0000000000000000 T _ITM_registerTMCloneTable
0000000000000000 T _GNU_EH_FRAME_HDR
0000000000000000 T _gmon_start
```

where is allocated? **stack** // int result .4

השתמשי בפקודה objdump

ניתן לראות את מה שקורה בתוך הפונקציה square (השורות בתוכה משתמשות בפונקציות, קוק, push, mov -פונקציות של מחסנית)



```
objdump -d a.out
Disassembly of section .text:

0000000000000000: <square>:
0000000000000000: push    %rbp
0000000000000001: mov     %rsp,%rbp
0000000000000002: mov     %edi,-0x14(%rbp)
0000000000000003: mov     -0x14(%rbp),%eax
0000000000000004: imul    -0x14(%rbp),%eax
0000000000000005: mov     %eax,-0x4(%rbp)
0000000000000006: mov     -0x4(%rbp),%eax
0000000000000007: pop     %rbp
0000000000000008: retq

0000000000000009: <doCalc>:
0000000000000009: push    %rbp
```

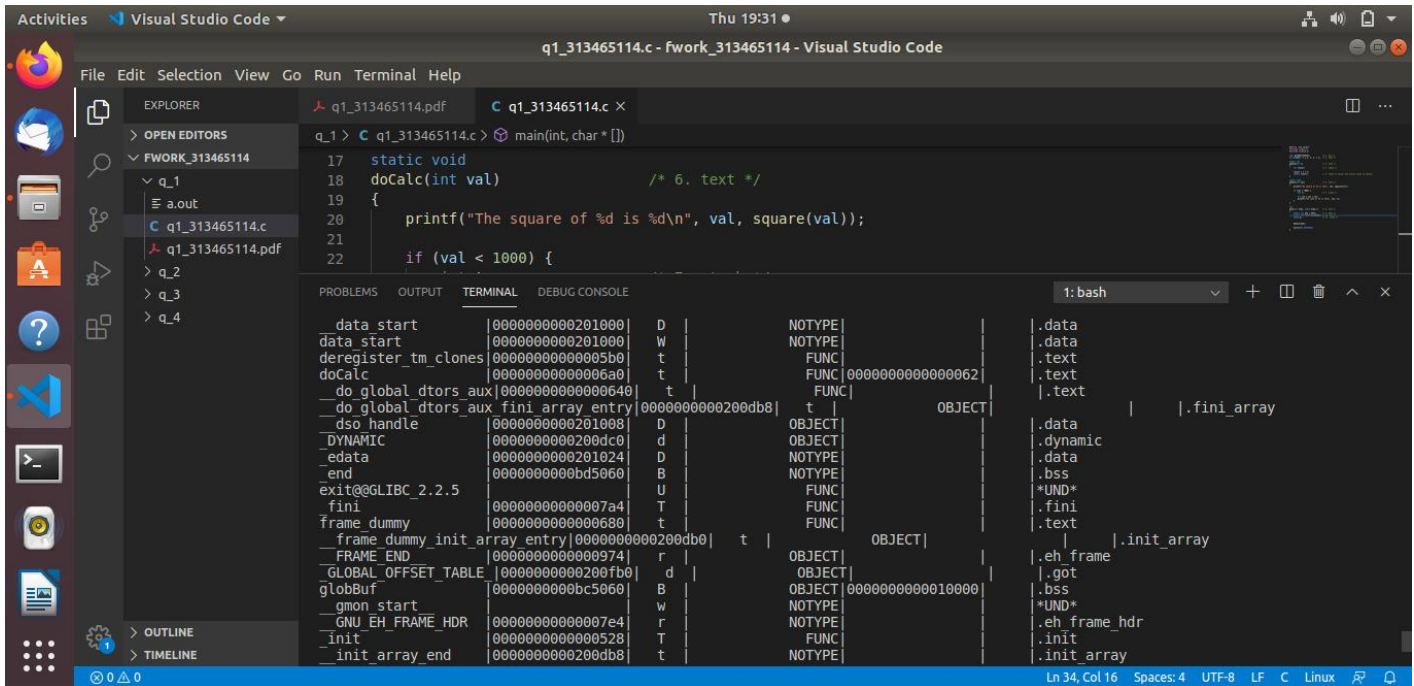
How the return value is passed? **stack to caller and caller saves in stack** .5

אותה תמונה מ4, בסוף החלק של הפונק square אנחנו רואים שהוא עושה קוק

where is allocated? text // doCalc(int val) .6

הרצתי את הפקודה nm -f sys a.out

שם כתוב איפה כל שורה בקוד ממוקמת

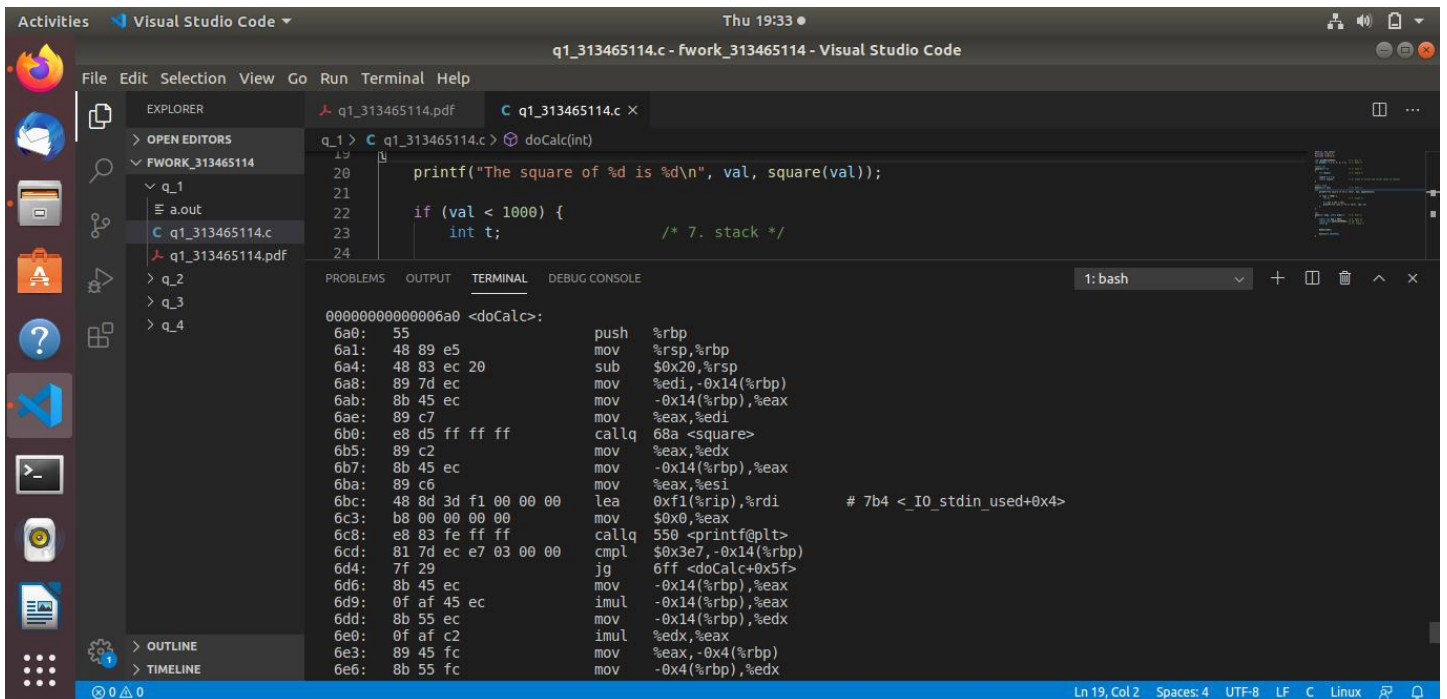


```
data start 000000000201000 D | NOTYPE | .data
data start 000000000201000 W | NOTYPE | .data
deregister_tm_clones 0000000000005b0 t | FUNC | .text
doCalc 00000000000006a0 t | FUNC | .text
do_global_dtors_aux 0000000000000640 t | FUNC | .text
do_global_dtors_aux_fini_array_entry 00000000000020db8 t | OBJECT | .fini_array
dso_handle 000000000201008 D | OBJECT | .data
_DYNAMIC 000000000200dc0 d | OBJECT | .dynamic
edata 000000000201024 D | NOTYPE | .data
end 000000000bd5060 B | NOTYPE | .bss
exit@GLIBC_2.2.5 0000000000007a4 U | FUNC | *UND*
fini 0000000000000680 t | FUNC | .fini
frame_dummy 0000000000000680 t | FUNC | .text
frame_dummy_init_array_entry 00000000000020db8 t | OBJECT | .init_array
FRAME_END 0000000000000974 r | OBJECT | .eh_frame
GLOBAL_OFFSET_TABLE 000000000200fb0 d | OBJECT | .got
globBuf 0000000000bc5060 B | OBJECT | .bss
gmon_start 0000000000000e4 w | NOTYPE | *UND*
GNU_EH_FRAME_HDR 0000000000000e4 r | NOTYPE | .eh_frame_hdr
init 0000000000000528 t | FUNC | .init
init_array_end 000000000200db8 t | NOTYPE | .init_array
```

;where is allocated? stack //int t .7

השתמתי בפקודה objdump

ניתן לראות את מה שקורה בתוך הפונקציה doCalc (השורות בתוכה משתמשות בפונקציות, pop, push, mov - פונקציות של מחסנית)



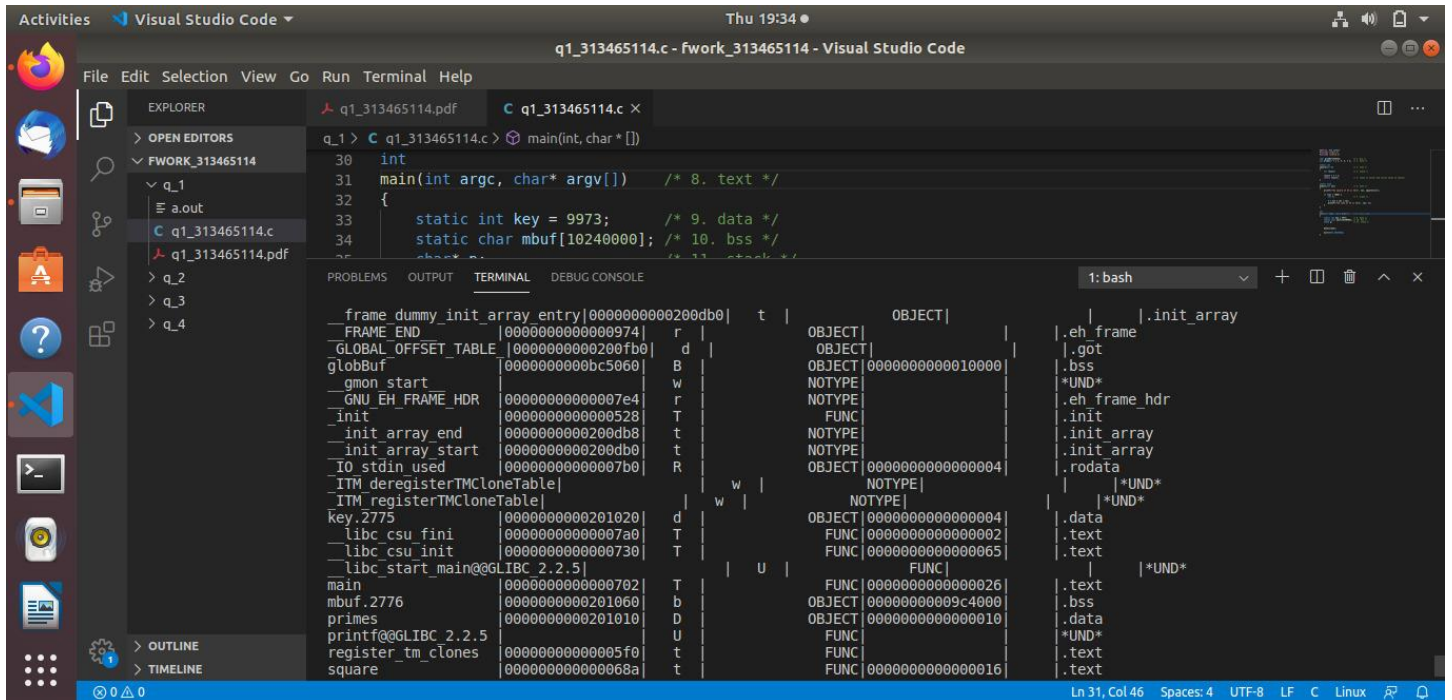
```
00000000000006a0 <doCalc>:
6a0: 55          push   %rbp
6a1: 48 89 e5    mov    %rsp,%rbp
6a4: 48 83 ec 20 sub    $0x20,%rsp
6a8: 89 7d ec    mov    %edi,-0x14(%rbp)
6ab: 8b 45 ec    mov    -0x14(%rbp),%eax
6ae: 89 c7      mov    %eax,%edi
6b0: e8 d5 ff ff callq  68a <square>
6b5: 89 c2      mov    %eax,%edx
6b7: 8b 45 ec    mov    -0x14(%rbp),%eax
6ba: 89 c6      mov    %eax,%esi
6bc: 48 8d 3d f1 00 00 00 lea    0xf1(%rip),%rdi
6c3: b8 00 00 00 00 mov    $0x0,%eax
6c8: e8 83 fe ff ff callq  550 <printf@plt>
6cd: 81 7d ec e7 03 00 00 cmpl   $0x3e7,-0x14(%rbp)
6d4: 7f 29      jg     6ff <doCalc+0x5f>
6d6: 8b 45 ec    mov    -0x14(%rbp),%eax
6d9: 0f af 45 ec imul   -0x14(%rbp),%eax
6dd: 8b 55 ec    mov    -0x14(%rbp),%edx
6e0: 0f af c2    imul   %edx,%eax
6e3: 89 45 fc    mov    %eax,-0x4(%rbp)
6e6: 8b 55 fc    mov    -0x4(%rbp),%edx
```



where is allocated? text //main(int argc, char\* argv[]) .8

הרצתי את הפקודה nm -f sys a.out

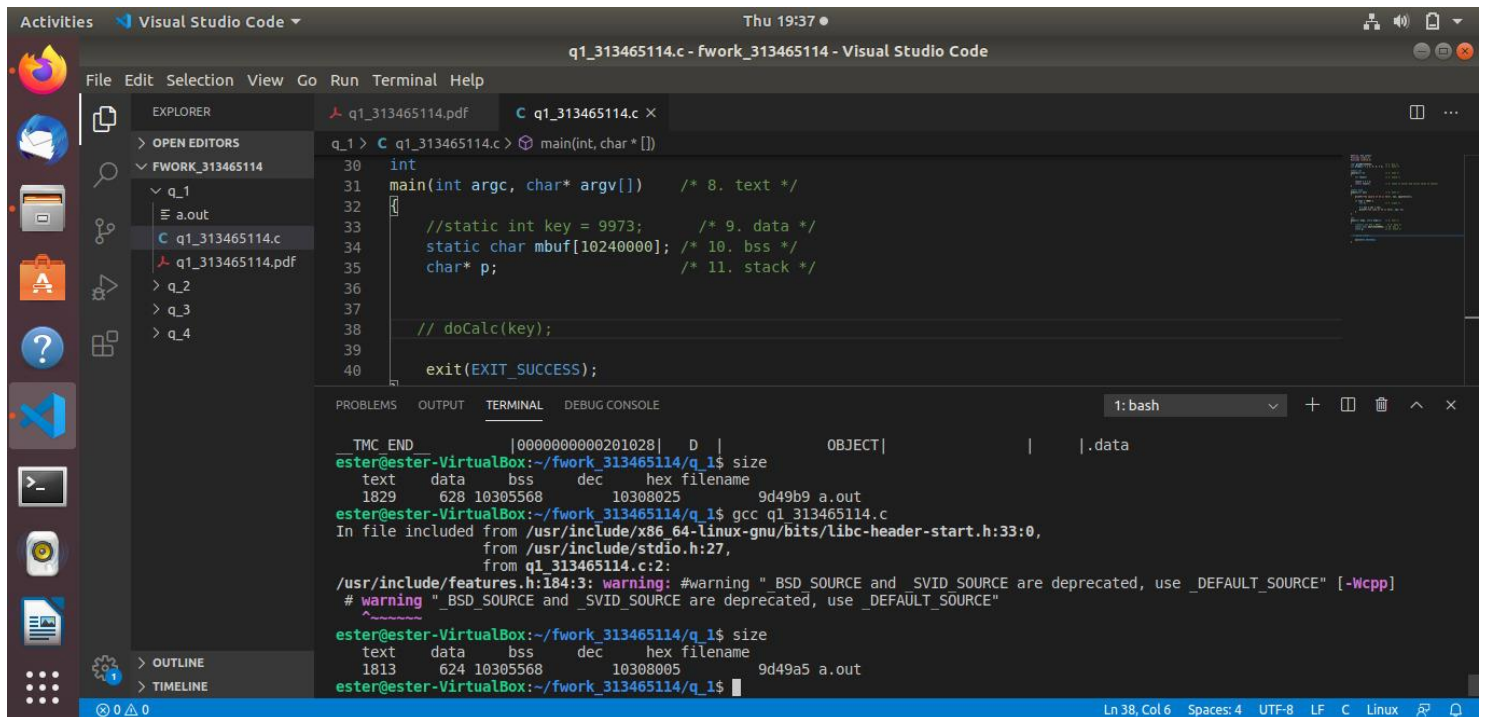
שם כתוב איפה כל שורה בקוד ממוקמת



```
q_1 > C q1_313465114.c > main(int, char * [])
30 int
31 main(int argc, char* argv[]) /* 8. text */
32 {
33     static int key = 9973; /* 9. data */
34     static char mbuf[10240000]; /* 10. bss */
35     char* p; /* 11. stack */
36
37     // doCalc(key);
38
39     exit(EXIT_SUCCESS);
40
__TMC_END__ |000000000201028| D | OBJECT| |.data
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1829 628 10305568 10308025 9d49b9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$ gcc q1_313465114.c
In file included from /usr/include/x86_64-linux-gnu/bits/libc-header-start.h:33:0,
from /usr/include/stdio.h:27,
from q1_313465114.c:2:
/usr/include/features.h:184:3: warning: #warning "BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE" [-Wcpp]
# warning "BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE"
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1813 624 10305568 10308005 9d49a5 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$
```

where is allocated? data // static int key = 9973 .9

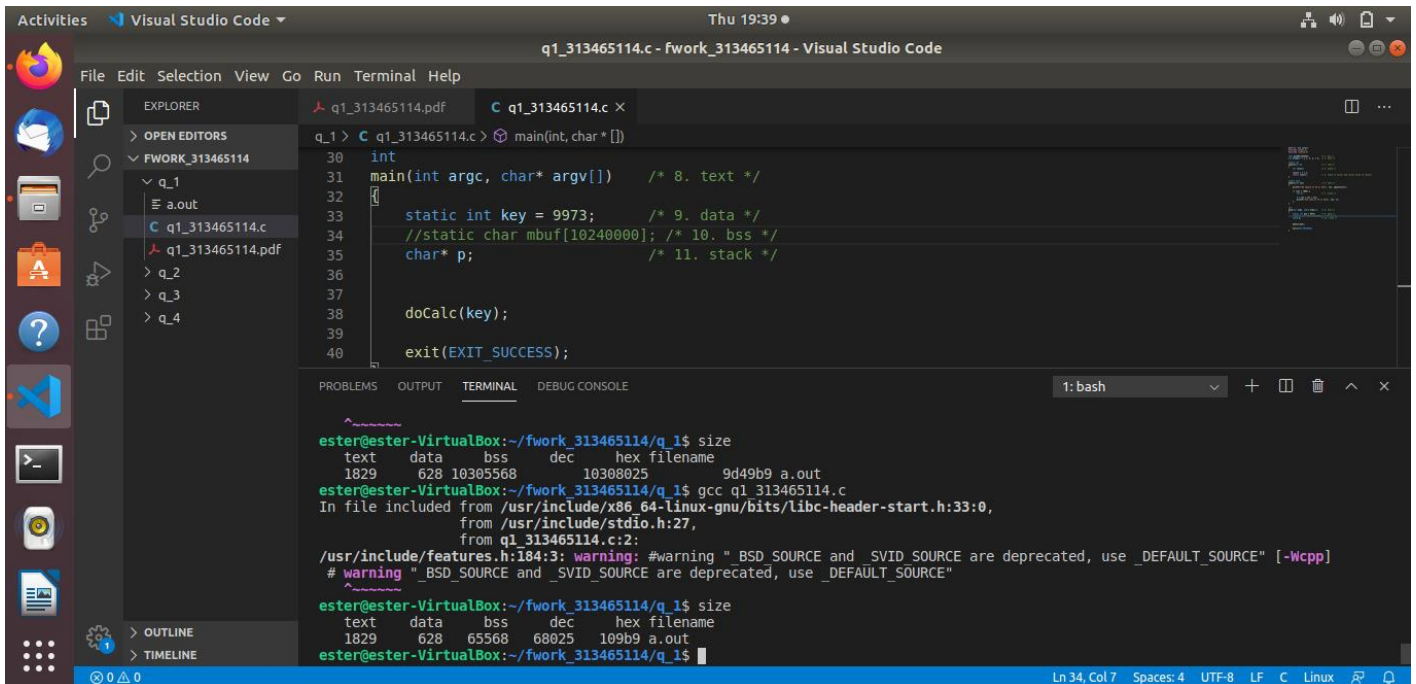
קימפלתי את הקוד והשתמשתי ב size (הsize הראשון שרואים בטרמינל).  
לאחר מכן מחקתי את אותה שורה, קימפלתי ושוב השתמשתי ב size (השני שרואים  
בטרמינל).  
ניתן לראות שהגודל של data השתנה



```
q_1 > C q1_313465114.c > main(int, char * [])
30 int
31 main(int argc, char* argv[]) /* 8. text */
32 {
33     //static int key = 9973; /* 9. data */
34     static char mbuf[10240000]; /* 10. bss */
35     char* p; /* 11. stack */
36
37     // doCalc(key);
38
39     exit(EXIT_SUCCESS);
40
__TMC_END__ |000000000201028| D | OBJECT| |.data
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1829 628 10305568 10308025 9d49b9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$ gcc q1_313465114.c
In file included from /usr/include/x86_64-linux-gnu/bits/libc-header-start.h:33:0,
from /usr/include/stdio.h:27,
from q1_313465114.c:2:
/usr/include/features.h:184:3: warning: #warning "BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE" [-Wcpp]
# warning "BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE"
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text data bss dec hex filename
1813 624 10305568 10308005 9d49a5 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$
```

10. static char mbuf[1024000]; where is allocated?

קימפלתי את הקוד והשתמשתי ב size (הsize הראשון שרואים בטרמינל).  
לאחר מכן מחקתי את אותה שורה, קימפלתי ושוב השתמשתי ב size (השני שרואים בטרמינל).  
ניתן לראות שהגודל של bss השתנה



```
q1_313465114.c - fwork_313465114 - Visual Studio Code

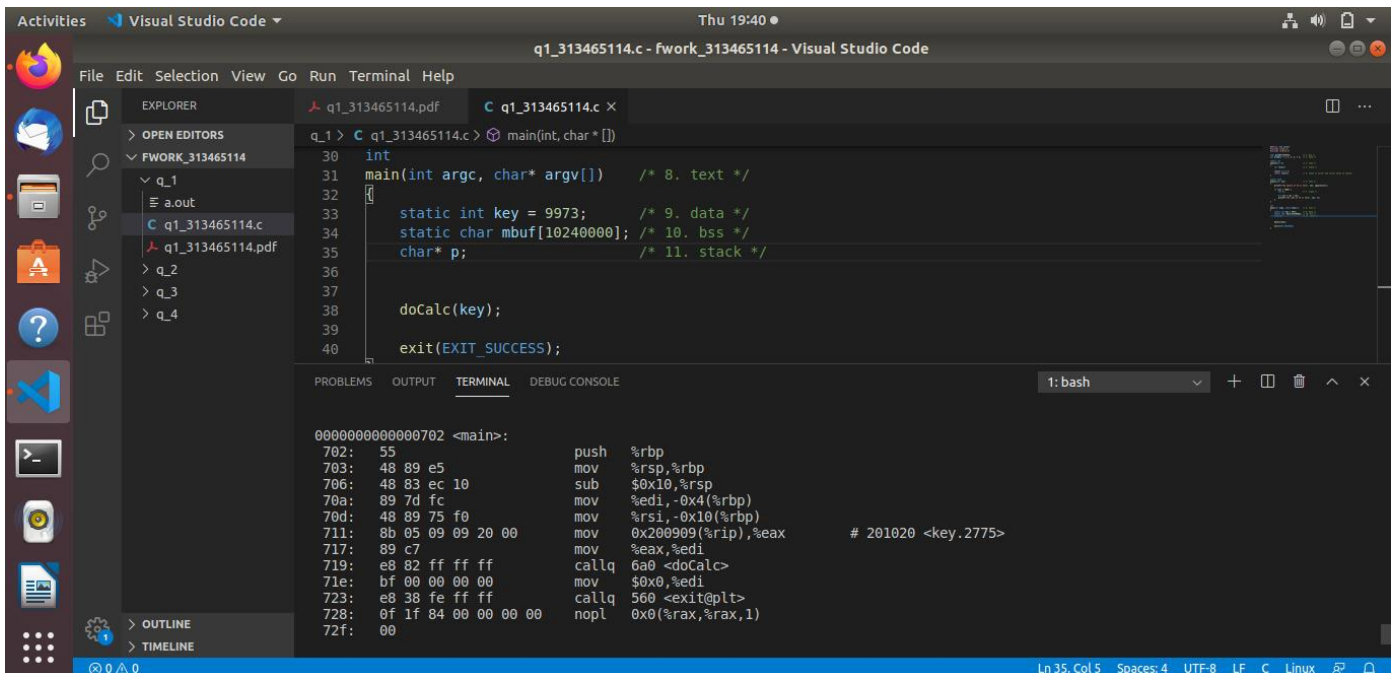
30 int
31 main(int argc, char* argv[]) /* 8. text */
32 {
33     static int key = 9973; /* 9. data */
34     //static char mbuf[1024000]; /* 10. bss */
35     char* p; /* 11. stack */
36
37
38     doCalc(key);
39
40     exit(EXIT_SUCCESS);
41 }
```

```
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text  data  bss  dec  hex filename
1829   628 10305568 10306197 9d49b9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$ gcc q1_313465114.c
In file included from /usr/include/x86_64-linux-gnu/bits/libc-header-start.h:33:0,
from /usr/include/stdio.h:27,
from q1_313465114.c:2:
/usr/include/features.h:184:3: warning: #warning "_BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE" [-Wcpp]
# warning "_BSD_SOURCE and _SVID_SOURCE are deprecated, use _DEFAULT_SOURCE"
^
ester@ester-VirtualBox:~/fwork_313465114/q_1$ size
text  data  bss  dec  hex filename
1829   628 65568 66197 100b9 a.out
ester@ester-VirtualBox:~/fwork_313465114/q_1$
```

11. char\* p // stack where is allocated?

השתמשתי בפקודה objdump

ניתן לראות את מה שקורה בתוך הפונקציה doCalc (השורות בתוכה משתמשות בפונקציות, pop, push, mov - פונקציות של מחסנית)



```
q1_313465114.c - fwork_313465114 - Visual Studio Code

30 int
31 main(int argc, char* argv[]) /* 8. text */
32 {
33     static int key = 9973; /* 9. data */
34     static char mbuf[1024000]; /* 10. bss */
35     char* p; /* 11. stack */
36
37
38     doCalc(key);
39
40     exit(EXIT_SUCCESS);
41 }
```

```
000000000000702: <main>:
702: 55                push    %rbp
703: 48 89 e5          mov     %rsp,%rbp
706: 48 83 ec 10       sub     $0x10,%rsp
70a: 89 7d fc          mov     %edi,-0x4(%rbp)
70d: 48 89 75 f0       mov     %rsi,-0x10(%rbp)
711: 8b 05 09 09 20 00 mov     0x200909(%rip),%eax # 201020 <key.2775>
717: 89 c7            mov     %eax,%edi
719: e8 82 ff ff ff    callq   6a0 <doCalc>
71e: bf 00 00 00 00    mov     $0x0,%edi
723: e8 38 fe ff ff    callq   560 <exit@plt>
728: 0f 1f 84 00 00 00 nopl    0x0(%rax,%rax,1)
72f: 00
```