- **Question 1 : Segmentation Fault** :

1. Q1 : خروجی Segmentation Fault میباشد به این دلیل که آدرسی که به pointer میدهیم آدرس معتبری نیست و اینکه ما مقدار 132 را در آنجا ذخیره میکنیم درست نیست.

2. Q2 : تگ g- اطلاعات debug را میسازد که با gdb بتوان استفاده کرد.

3. Q3 : با استفاده از gdb میتوان مرحله به مرحله برنامه را اجرا کرد و مشکلات را فهمید و درک خوبی نسبت به کد دریافت میکنیم.

```
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from null...
(gdb) r
Starting program: /mnt/f/term5/OS/Practical HW6/null

Program received signal SIGSEGV, Segmentation fault.
0x000055555555513d in main () at null.c:8
8           *ptr = 132;
(gdb) s

Program terminated with signal SIGSEGV, Segmentation fault.
The program no longer exists.
(gdb)
```

4. Q4 : وقتی اجرا شد یک سری پیام ها نمایش داده شد که داخل خط 8 که به pointer مقدار 132 داده شده ارور داره و متن ارور هم این هستش Address 0x0 is not stack'd, malloc'd or (recently) free'd که ارور را توضیح داده. خروجی کامند:

```
mohammad@mohammad:/mnt/f/term5/OS/Practical HW6$ valgrind --tool=memcheck --leak-check=yes ./null
==2483== Memcheck, a memory error detector
==2483== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==2483== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==2483== Command: ./null
==2483==
==2483== Invalid write of size 4
==2483==    at 0x10913D: main (null.c:8)
==2483==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==2483==
==2483==
==2483== Process terminating with default action of signal 11 (SIGSEGV)
==2483==  Access not within mapped region at address 0x0
==2483==    at 0x10913D: main (null.c:8)
==2483==  If you believe this happened as a result of a stack
==2483==  overflow in your program's main thread (unlikely but
==2483==  possible), you can try to increase the size of the
==2483==  main thread stack using the --main-stacksize= flag.
==2483==  The main thread stack size used in this run was 8388608.
==2483==
==2483== HEAP SUMMARY:
==2483==     in use at exit: 0 bytes in 0 blocks
==2483==   total heap usage: 0 allocs, 0 frees, 0 bytes allocated
==2483==
==2483== All heap blocks were freed -- no leaks are possible
==2483==
==2483== For lists of detected and suppressed errors, rerun with: -s
==2483== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
Segmentation fault
mohammad@mohammad:/mnt/f/term5/OS/Practical HW6$
```

- **Question 2 : Memory Leak** :

1. ./leak : همه چیز درست بود و برنامه بدون ارور اجرا شد.

2. gdb leak : با دستور b <num of line> در خط مورد نظر break point گذاشتم و برنامه مرحله به مرحله بدون ارور اجرا شد و اخرین پیام برنامه "[Inferior 1 (process 3470) exited normally] " بود.

3. valgrind --tool=memcheck --leak-check=yes ./leak : با این دستور هم خروجی نمایش داده شد اما یک ارور هم دارد که متن اروری هم که به ما نشان میدهد bytes in 1 blocks are definitely lost in loss record 1 of 1 12 میباشد که به این دلیل هستش که ما deallocate نکرده ایم و انگار که چند بلاک از مموری گم شده است.
تصویر خروجی:

```
mohammad@mohammad:/mnt/f/term5/OS/Practical HW6$ valgrind --tool=memcheck --leak-check=yes ./leak
==3753== Memcheck, a memory error detector
==3753== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==3753== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==3753== Command: ./leak
==3753==
1
2
3
==3753==
==3753== HEAP SUMMARY:
==3753==     in use at exit: 12 bytes in 1 blocks
==3753==   total heap usage: 2 allocs, 1 frees, 1,036 bytes allocated
==3753==
==3753== 12 bytes in 1 blocks are definitely lost in loss record 1 of 1
==3753==    at 0x483B7F3: malloc (in /usr/lib/x86_64-linux-gnu/valgrind/vgpreload_memcheck-amd64-linux.so)
==3753==    by 0x10918C: main (leak.c:12)
==3753==
==3753== LEAK SUMMARY:
==3753==    definitely lost: 12 bytes in 1 blocks
==3753==    indirectly lost: 0 bytes in 0 blocks
==3753==      possibly lost: 0 bytes in 0 blocks
==3753==    still reachable: 0 bytes in 0 blocks
==3753==         suppressed: 0 bytes in 0 blocks
==3753==
==3753== For lists of detected and suppressed errors, rerun with: -s
==3753== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
mohammad@mohammad:/mnt/f/term5/OS/Practical HW6$
```

- **Question 3 : Index Out of Range** :

1. Q1 : برنامه بدون هیچ اروری اجرا میشود. دلیل اینکه ارور نداریم این هستش که توی زبان c و cpp به حد و مرز آرایه ها توجه نمیشود و به سیستم عامل بستگی دارد و این آرایه ها stack based array هستند و سیستم عامل و thread ها بخش خاصی از حافظه را برای استک ها رزرو کرده اند و تا زمانی که ایندکسی از آرایه را که از فضای مشخص خارج نشده مقدار دهی نکنیم خطای seg fault نمیدهد.

2. Q2 : اروری نمایش نمیدهد و همه چی اوکی هست.

3. Q3 : ارور segmentation fault رخ میدهد و برنامه اجرا نمیشود. دلیل هم این است که از فضایی که سیستم عامل به این آرایه اختصاص داده خارج شدیم. در اصل با توجه به page size و آدرس آخرین ایندکس آرایه، با توجه به مقداری که سیستم عامل تعیین کرده ما تا اون ایندکس میتوانیم مقدار دهی کنیم و بیشتر از آن خطای segmentation fault میگیریم.

- **Question 4 : Access** :

1. Q1 : ارور invalid pointer میدهد (در صورتی که برای free پوینتر به data+50 داده باشیم) اما اگر پوینتر به خود آرایه داده باشیم، ارور نمیدهد.

2. Q2 : خطای definitely lost: 400 bytes in 1 blocks میدهد (در صورتی که برای free پوینتر به data+50 داده باشیم) اما اگر پوینتر به خود آرایه داده باشیم، ارور نمیدهد.

```
mohammad@mohammad:/mnt/f/term5/OS/Practical HW6$ valgrind --tool=memcheck --leak-check=yes ./access
==5316== Memcheck, a memory error detector
==5316== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==5316== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==5316== Command: ./access
==5316==
==5316== Invalid free() / delete / delete[] / realloc()
==5316==    at 0x483CA3F: free (in /usr/lib/x86_64-linux-gnu/valgrind/vgpreload_memcheck-amd64-linux.so)
==5316==    by 0x1091A2: main (access.c:12)
==5316==  Address 0x4a4e108 is 200 bytes inside a block of size 400 alloc'd
==5316==    at 0x483B7F3: malloc (in /usr/lib/x86_64-linux-gnu/valgrind/vgpreload_memcheck-amd64-linux.so)
==5316==    by 0x10918C: main (access.c:10)
==5316==
==5316==
==5316== HEAP SUMMARY:
==5316==     in use at exit: 400 bytes in 1 blocks
==5316==   total heap usage: 1 allocs, 1 frees, 400 bytes allocated
==5316==
==5316== 400 bytes in 1 blocks are definitely lost in loss record 1 of 1
==5316==    at 0x483B7F3: malloc (in /usr/lib/x86_64-linux-gnu/valgrind/vgpreload_memcheck-amd64-linux.so)
==5316==    by 0x10918C: main (access.c:10)
==5316==
==5316== LEAK SUMMARY:
==5316==    definitely lost: 400 bytes in 1 blocks
==5316==    indirectly lost: 0 bytes in 0 blocks
==5316==      possibly lost: 0 bytes in 0 blocks
==5316==    still reachable: 0 bytes in 0 blocks
==5316==         suppressed: 0 bytes in 0 blocks
==5316==
==5316== For lists of detected and suppressed errors, rerun with: -s
==5316== ERROR SUMMARY: 2 errors from 2 contexts (suppressed: 0 from 0)
mohammad@mohammad:/mnt/f/term5/OS/Practical HW6$
```

3. Q3 : ارور invalid pointer داده میشود زیرا نمیتوان در تابع free پوینتر به آخر آرایه مورد قبول است و نمیتوان برای مثال پوینتر به وسط آرایه به عنوان ورودی به تابع داد.