ROLL NO : 422176
NAME : KOTA VENKATA CHARAN TEJA
SECTION:A
QUESTION:

Generate different C programs that induce a segmentation fault error, select these examples of your choice, and employ the GDB utility for debugging on Linux.

Note:
1. Include multiple breakpoints while debugging
2. Upload your submission in a format consistent with the example provided in the material

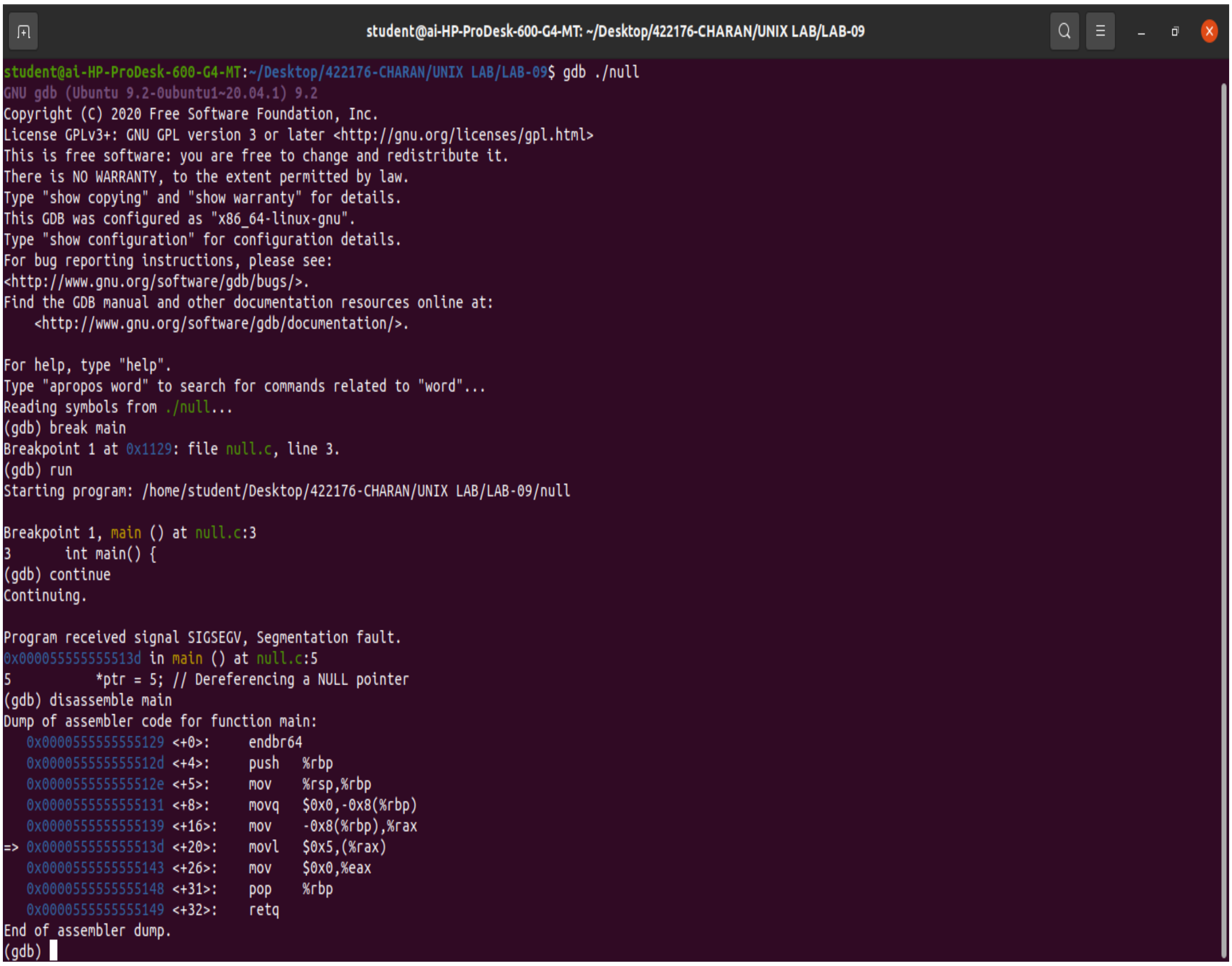
CODE:

null.c

```
#include <stdio.h>

int main() {
    int *ptr = NULL;
    *ptr = 5; // Dereferencing a NULL pointer
    return 0;
}
```

DEBUGGING null.c



student@ai-HP-ProDesk-600-G4-MT: ~/Desktop/422176-CHARAN/UNIX LAB/LAB-09

```
student@ai-HP-ProDesk-600-G4-MT:~/Desktop/422176-CHARAN/UNIX LAB/LAB-09$ gdb ./null
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./null...
(gdb) break main
Breakpoint 1 at 0x1129: file null.c, line 3.
(gdb) run
Starting program: /home/student/Desktop/422176-CHARAN/UNIX LAB/LAB-09/null

Breakpoint 1, main () at null.c:3
3       int main() {
(gdb) continue
Continuing.

Program received signal SIGSEGV, Segmentation fault.
0x000055555555513d in main () at null.c:5
5           *ptr = 5; // Dereferencing a NULL pointer
(gdb) disassemble main
Dump of assembler code for function main:
   0x0000555555555129 <+0>:     endbr64
   0x000055555555512d <+4>:     push   %rbp
   0x000055555555512e <+5>:     mov    %rsp,%rbp
   0x0000555555555131 <+8>:     movq   $0x0,-0x8(%rbp)
   0x0000555555555139 <+16>:    mov    -0x8(%rbp),%rax
=> 0x000055555555513d <+20>:    movl   $0x5,(%rax)
   0x0000555555555143 <+26>:    mov    $0x0,%eax
   0x0000555555555148 <+31>:    pop    %rbp
   0x0000555555555149 <+32>:    retq
End of assembler dump.
(gdb) 
```

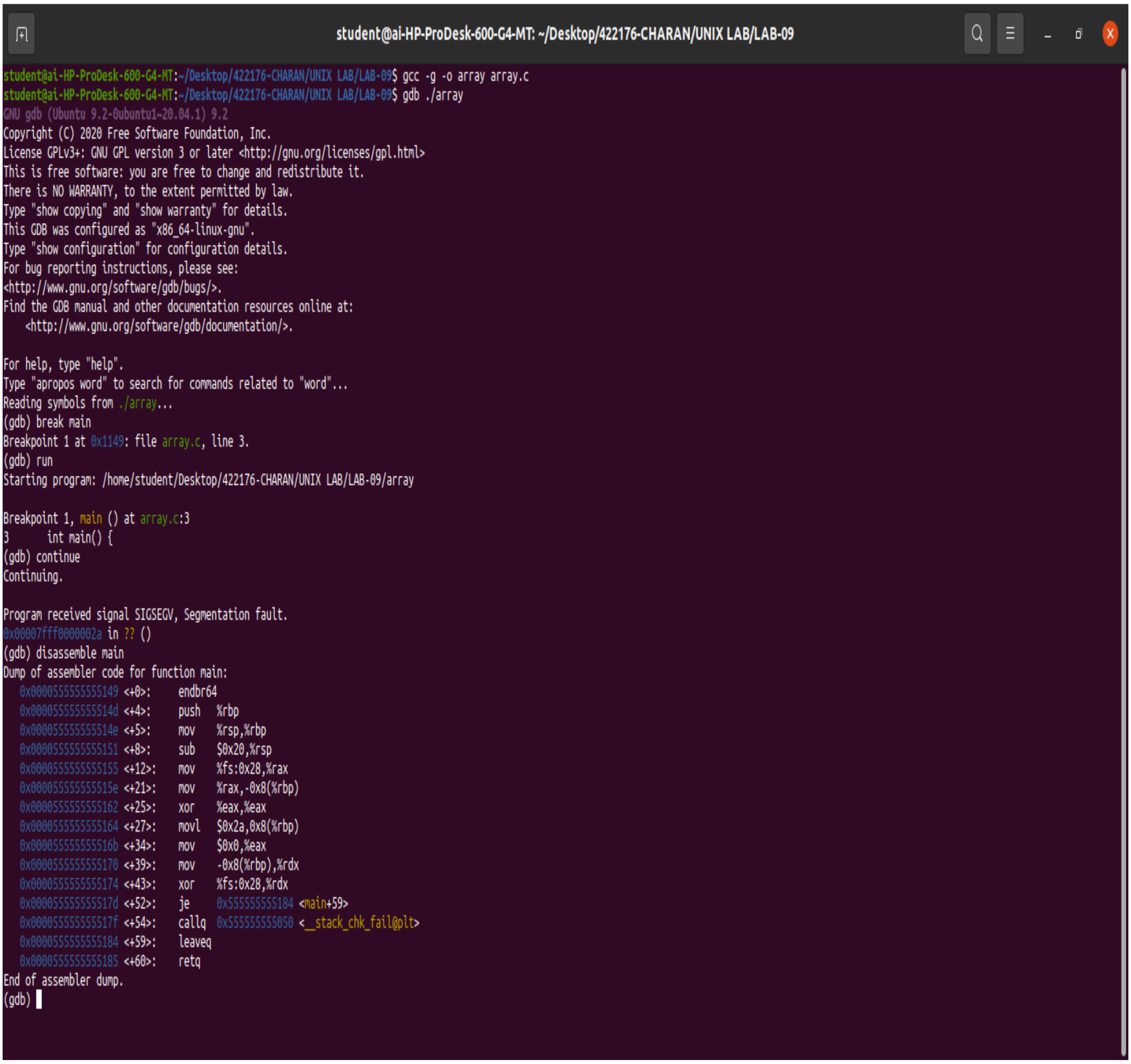
.

array.c

```
#include <stdio.h>

int main() {
    int arr[5];
    arr[10] = 42; // Accessing an array out of bounds
    return 0;
}
```

DEBUGGING array.c


student@ai-HP-ProDesk-600-G4-MT: ~/Desktop/422176-CHARAN/UNIX LAB/LAB-09

```
student@ai-HP-ProDesk-600-G4-MT:~/Desktop/422176-CHARAN/UNIX LAB/LAB-09$ gcc -g -o array array.c
student@ai-HP-ProDesk-600-G4-MT:~/Desktop/422176-CHARAN/UNIX LAB/LAB-09$ gdb ./array
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./array...
(gdb) break main
Breakpoint 1 at 0x1149: file array.c, line 3.
(gdb) run
Starting program: /home/student/Desktop/422176-CHARAN/UNIX LAB/LAB-09/array

Breakpoint 1, main () at array.c:3
3       int main() {
(gdb) continue
Continuing.

Program received signal SIGSEGV, Segmentation fault.
0x00007fff0000002a in ?? ()
(gdb) disassemble main
Dump of assembler code for function main:
   0x0000555555555149 <+0>:     endbr64
   0x000055555555514d <+4>:     push   %rbp
   0x000055555555514e <+5>:     mov    %rsp,%rbp
   0x0000555555555151 <+8>:     sub    $0x20,%rsp
   0x0000555555555155 <+12>:    mov    %fs:0x28,%rax
   0x000055555555515e <+21>:    mov    %rax,-0x8(%rbp)
   0x0000555555555162 <+25>:    xor    %eax,%eax
   0x0000555555555164 <+27>:    movl   $0x2a,0x8(%rbp)
   0x000055555555516b <+34>:    mov    $0x0,%eax
   0x0000555555555170 <+39>:    mov    -0x8(%rbp),%rdx
   0x0000555555555174 <+43>:    xor    %fs:0x28,%rdx
   0x000055555555517d <+52>:    je     0x555555555184 <main+59>
   0x000055555555517f <+54>:    callq  0x555555555050 <__stack_chk_fail@plt>
   0x0000555555555184 <+59>:    leaveq
   0x0000555555555185 <+60>:    retq
End of assembler dump.
(gdb) 
```

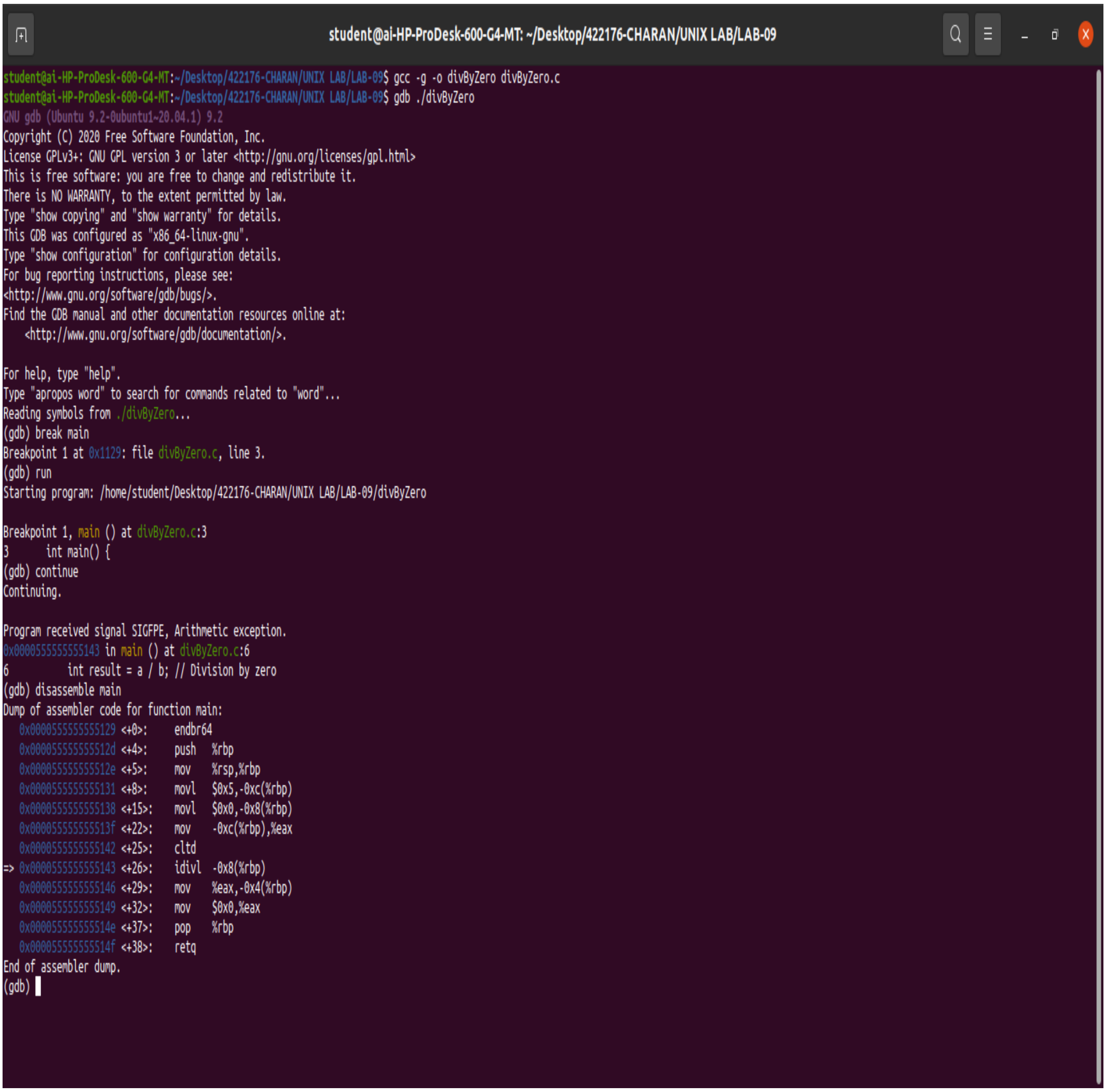
-

divByZero.c

```
#include <stdio.h>

int main() {
    int a = 5;
    int b = 0;
    int result = a / b; // Division by zero
    return 0;
}
```

DEBUGGING divByZero.c


student@ai-HP-ProDesk-600-G4-MT: ~/Desktop/422176-CHARAN/UNIX LAB/LAB-09

```
student@ai-HP-ProDesk-600-G4-MT:~/Desktop/422176-CHARAN/UNIX LAB/LAB-09$ gcc -g -o divByZero divByZero.c
student@ai-HP-ProDesk-600-G4-MT:~/Desktop/422176-CHARAN/UNIX LAB/LAB-09$ gdb ./divByZero
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./divByZero...
(gdb) break main
Breakpoint 1 at 0x1129: file divByZero.c, line 3.
(gdb) run
Starting program: /home/student/Desktop/422176-CHARAN/UNIX LAB/LAB-09/divByZero

Breakpoint 1, main () at divByZero.c:3
3       int main() {
(gdb) continue
Continuing.

Program received signal SIGFPE, Arithmetic exception.
0x0000555555555143 in main () at divByZero.c:6
6           int result = a / b; // Division by zero
(gdb) disassemble main
Dump of assembler code for function main:
   0x0000555555555129 <+0>:     endbr64
   0x000055555555512d <+4>:     push   %rbp
   0x000055555555512e <+5>:     mov    %rsp,%rbp
   0x0000555555555131 <+8>:     movl   $0x5,-0xc(%rbp)
   0x0000555555555138 <+15>:    movl   $0x0,-0x8(%rbp)
   0x000055555555513f <+22>:    mov    -0xc(%rbp),%eax
   0x0000555555555142 <+25>:    cltd
=> 0x0000555555555143 <+26>:    idivl  -0x8(%rbp)
   0x0000555555555146 <+29>:    mov    %eax,-0x4(%rbp)
   0x0000555555555149 <+32>:    mov    $0x0,%eax
   0x000055555555514e <+37>:    pop    %rbp
   0x000055555555514f <+38>:    retq
End of assembler dump.
(gdb) 
```