# Assignment-4

Name: k.Mohan.
Sec: A
Roll No: 422166.

Q. 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 :

```
#include <stdio.h>

int main() {
    int *ptr = NULL; // Pointing to NULL, accessing it will cause segmentation fault
    int result = 0;

    for (int i = 0; i < 100; i++) {
        result += ptr[i]; // Addition operation (will cause segmentation fault)
        result -= ptr[i]; // Subtraction operation (will cause segmentation fault)
        result *= ptr[i]; // Multiplication operation (will cause segmentation fault)
    }

    printf("Result: %d\n", result); // This won't be reached due to the segmentation fault
    return 0;
}
```

Output :

```
student@system62:~/Desktop/422166$ gcc seg.c
student@system62:~/Desktop/422166$ ./a.out
Segmentation fault (core dumped)
student@system62:~/Desktop/422166$ gcc -g seg.c
student@system62:~/Desktop/422166$ gdb ./a.out
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 ./a.out...
(gdb) list
1
2       #include <stdio.h>
3
4       int main() {
5           int *ptr = NULL; // Pointing to NULL, accessing it will cause segmentation fault
6           int result = 0;
7
8           for (int i = 0; i < 100; i++) {
9               result += ptr[i]; // Addition operation (will cause segmentation fault)
10              result -= ptr[i]; // Subtraction operation (will cause segmentation fault)
(gdb) list
11              result *= ptr[i]; // Multiplication operation (will cause segmentation fault)
12          }
13
14          printf("Result: %d\n", result); // This won't be reached due to the segmentation fault
15          return 0;
16      }
(gdb) list
Line number 17 out of range; seg.c has 16 lines.
(gdb) run
Starting program: /home/student/Desktop/422166/a.out

Program received signal SIGSEGV, Segmentation fault.
0x0000555555555181 in main () at seg.c:9
```

```
0x0000555555555181 in main () at seg.c:9
9               result += ptr[i]; // Addition operation (will cause segmentation fault)
(gdb) break 5
Breakpoint 1 at 0x555555555155: file seg.c, line 5.
(gdb) next

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