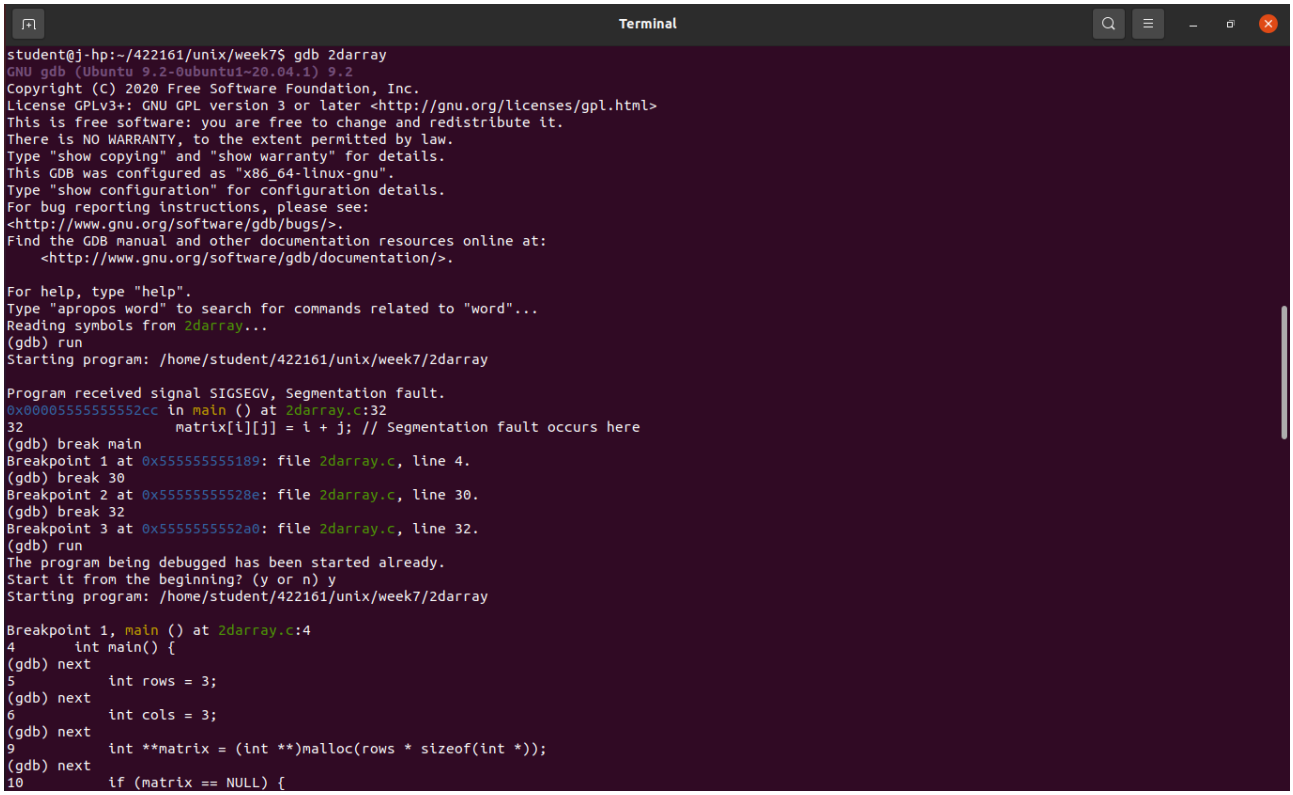
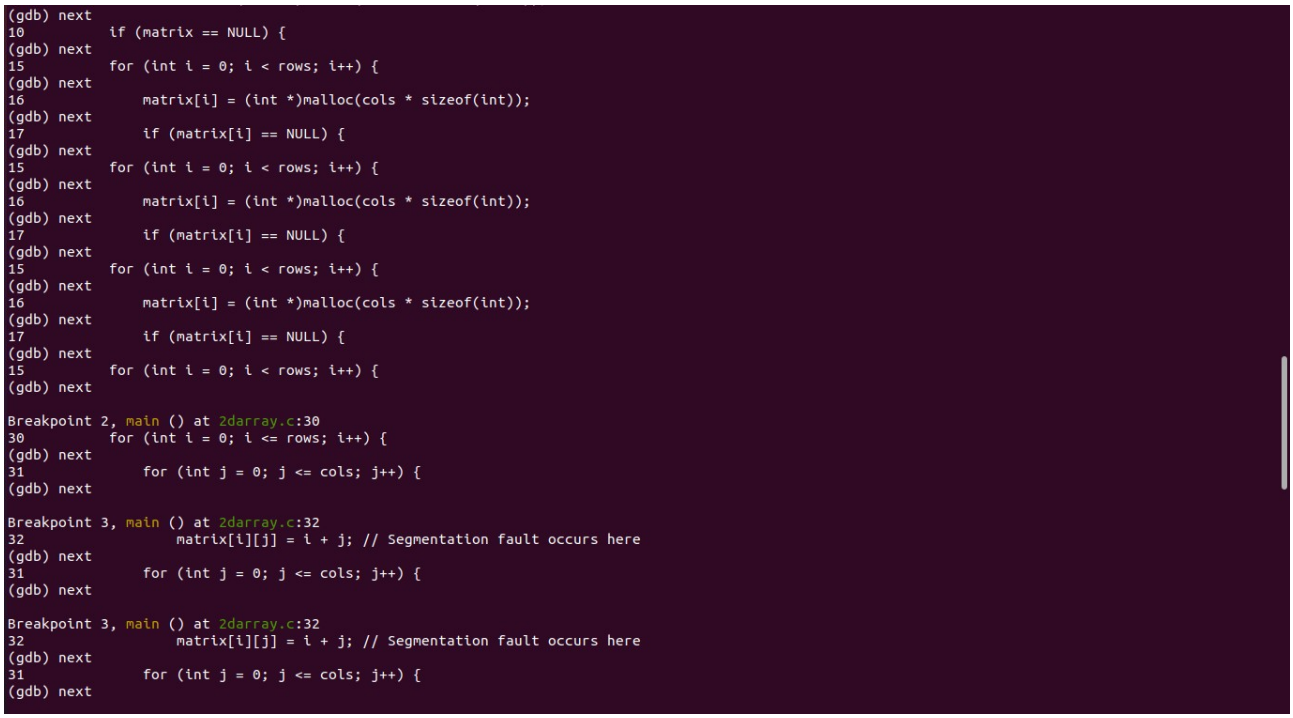
EXAMPLE 1 : Segementation fault in 2D array

Code:

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    int rows = 3;
    int cols = 3;

    // Dynamically allocate memory for a 2D array
    int **matrix = (int **)malloc(rows * sizeof(int *));
    if (matrix == NULL) {
        printf("Memory allocation failed\n");
        return 1;
    }

    for (int i = 0; i < rows; i++) {
        matrix[i] = (int *)malloc(cols * sizeof(int));
        if (matrix[i] == NULL) {
            printf("Memory allocation failed\n");

            // Free previously allocated memory
            for (int j = 0; j < i; j++) {
                free(matrix[j]);
            }
            free(matrix);
            return 1;
        }
    }

    // Accessing elements out of bounds
    for (int i = 0; i <= rows; i++) {
        for (int j = 0; j <= cols; j++) {
            matrix[i][j] = i + j; // Segmentation fault occurs here
        }
    }

    // Free allocated memory
    for (int i = 0; i < rows; i++) {
        free(matrix[i]);
    }
    free(matrix);

    return 0;
}
```

OUTPUT :


Terminal
student@j-hp:~/422161/unix/week7$ gdb 2darray
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 2darray...
(gdb) run
Starting program: /home/student/422161/unix/week7/2darray
Program received signal SIGSEGV, Segmentation fault.
0x00005555555552cc in main () at 2darray.c:32
32 matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) break main
Breakpoint 1 at 0x555555555189: file 2darray.c, line 4.
(gdb) break 30
Breakpoint 2 at 0x55555555528e: file 2darray.c, line 30.
(gdb) break 32
Breakpoint 3 at 0x5555555552a0: file 2darray.c, line 32.
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/student/422161/unix/week7/2darray
Breakpoint 1, main () at 2darray.c:4
4 int main() {
(gdb) next
5 int rows = 3;
(gdb) next
6 int cols = 3;
(gdb) next
9 int **matrix = (int **)malloc(rows * sizeof(int *));
(gdb) next
10 if (matrix == NULL) {


(gdb) next
10 if (matrix == NULL) {
(gdb) next
15 for (int i = 0; i < rows; i++) {
(gdb) next
16 matrix[i] = (int *)malloc(cols * sizeof(int));
(gdb) next
17 if (matrix[i] == NULL) {
(gdb) next
15 for (int i = 0; i < rows; i++) {
(gdb) next
16 matrix[i] = (int *)malloc(cols * sizeof(int));
(gdb) next
17 if (matrix[i] == NULL) {
(gdb) next
15 for (int i = 0; i < rows; i++) {
(gdb) next
16 matrix[i] = (int *)malloc(cols * sizeof(int));
(gdb) next
17 if (matrix[i] == NULL) {
(gdb) next
15 for (int i = 0; i < rows; i++) {
(gdb) next
Breakpoint 2, main () at 2darray.c:30
30 for (int i = 0; i <= rows; i++) {
(gdb) next
31 for (int j = 0; j <= cols; j++) {
(gdb) next
Breakpoint 3, main () at 2darray.c:32
32 matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31 for (int j = 0; j <= cols; j++) {
(gdb) next
Breakpoint 3, main () at 2darray.c:32
32 matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31 for (int j = 0; j <= cols; j++) {
(gdb) next

```
Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next
30          for (int i = 0; i <= rows; i++) {
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next
30          for (int i = 0; i <= rows; i++) {
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next
```

```
Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next
30          for (int i = 0; i <= rows; i++) {
(gdb) next
31              for (int j = 0; j <= cols; j++) {
(gdb) next

Breakpoint 3, main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next

Program received signal SIGSEGV, Segmentation fault.
0x00005555555552cc in main () at 2darray.c:32
32                  matrix[i][j] = i + j; // Segmentation fault occurs here
(gdb) next

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

EXAMPLE 2 :Accessing Invalid Memory Address

Code :

```
#include <stdio.h>

int main() {
    int *ptr = (int *)100; // Assigning an arbitrary address
    printf("%d", *ptr); // Accessing invalid memory address
    return 0;
}
```

OUTPUT :