

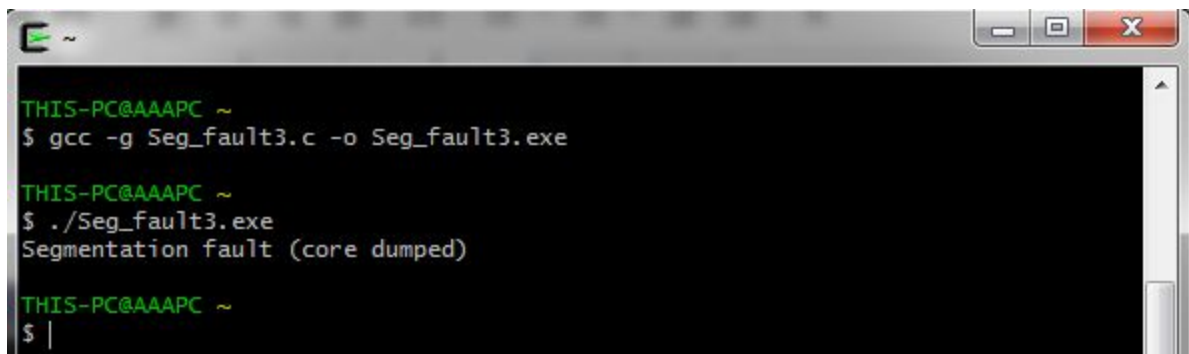
# Answers to Assignment No. 1 - GDB :

(BT17ECE021 - Ambarish .P. Chandurkar)

Program File: Seg\_fault3.c

```
#include <stdio.h>
int main(void)
{
    const int data[10] = {1,2,3,4,5,6,7};
    int i,sum;
    for(i=0;i>=0;i++){
        sum+=data[i];
    }
    printf("sum=%i \n",sum);
    return 0;
}
```

Program Compilation and Runtime Error:

A terminal window with a dark background and green text. The prompt is 'THIS-PC@AAAPC ~'. The user enters '\$ gcc -g Seg\_fault3.c -o Seg\_fault3.exe'. The prompt changes to 'THIS-PC@AAAPC ~' again. The user enters '\$ ./Seg\_fault3.exe'. The output is 'Segmentation fault (core dumped)'. The prompt changes to 'THIS-PC@AAAPC ~' again. The user enters '\$ |'.

Error Shown by GDB:

```
(gdb) file Seg_fault3.exe
Reading symbols from Seg_fault3.exe...done.
(gdb) run
Starting program: /home/THIS-PC/Seg_fault3.exe
[New Thread 5128.0xc3c]
[New Thread 5128.0xc40]
[New Thread 5128.0x1ba0]
[New Thread 5128.0xae8]

Thread 1 received signal SIGSEGV, Segmentation fault.
0x00000001004010f4 in main () at Seg_fault3.c:7
7          sum+=data[i];
```

Observation:

- 1) It is known that when we try to access the memory which is **out of bounds of array**, an error may occur at runtime.

Conclusion: The variable 'i' may be going out of 'data' array limits in the 'for' loop

Action: We use "Conditional Breakpoint" as follows:

```
break Seg_fault3.c:7 if i >= 10
```

We use the condition that if 'i' goes beyond the Array Size of 'data' which is 10, then the execution is to be halted at line no. 7

```
Type "apropos word" to search for commands related to "word"...
Reading symbols from Seg_fault3.exe...done.
(gdb) break Seg_fault3.c:7 if i >= 10
Breakpoint 1 at 0x1004010ef: file Seg_fault3.c, line 7.
(gdb) |
```

Subsequent Execution under GDB:

```
(gdb) break Seg_fault3.c:7 if i >= 10
Breakpoint 1 at 0x1004010ef: file Seg_fault3.c, line 7.
(gdb) run
Starting program: /home/THIS-PC/Seg_fault3.exe
[New Thread 2972.0x1b00]
[New Thread 2972.0x1810]
[New Thread 2972.0x1308]
[New Thread 2972.0x160c]

Thread 1 hit Breakpoint 1, main () at Seg_fault3.c:7
7          sum+=data[i];
(gdb) |
```

**As expected, the execution halted at line no. 7, indicating that 'i' is going beyond the bounds of array 'data'.**

Solution to Bug:

- 1) It is seen in the source file that 'i' is not properly limited by condition in the 'for' loop.

```
for(i=0;i>=0;i++)
```

This is an "Infinite Loop" since 'i' is always going to be greater than zero and hence is always incremented

2) The condition is corrected as follows:

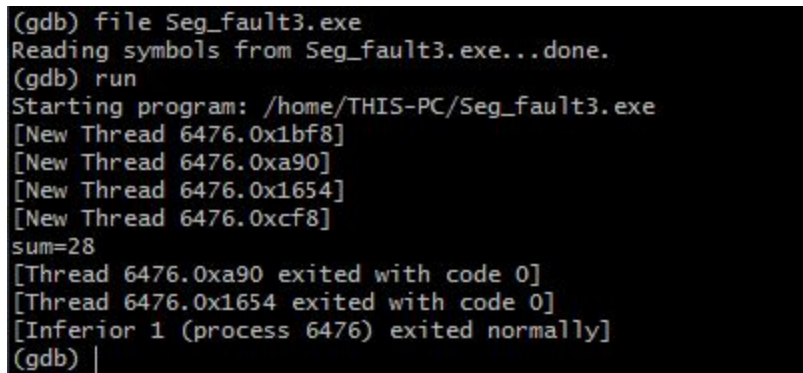
```
for(i=0;i<7;i++)
```

Though 10 = ARRAYSIZE of 'data', we put i<7 because 'data' has values only from 0 to 6 (i.e 7 values) .So now the variable 'i' is always under the ARRAYSIZE.

Corrected Program:

```
#include <stdio.h>
int main(void)
{
    const int data[10] = {1,2,3,4,5,6,7};
    int i,sum;
    for(i=0;i<=7;i++){ //Correction Performed
        sum+=data[i];
    }
    printf("sum=%i \n",sum);
    return 0;
}
```

Output Under GDB:



```
(gdb) file Seg_fault3.exe
Reading symbols from Seg_fault3.exe...done.
(gdb) run
Starting program: /home/THIS-PC/Seg_fault3.exe
[New Thread 6476.0x1bf8]
[New Thread 6476.0xa90]
[New Thread 6476.0x1654]
[New Thread 6476.0xcf8]
sum=28
[Thread 6476.0xa90 exited with code 0]
[Thread 6476.0x1654 exited with code 0]
[Inferior 1 (process 6476) exited normally]
(gdb) |
```

Since now GDB shows that our program has executed normally, it is finally debugged