

Bst_seg execution screenshots

```
pavan422178@pavan422178-VirtualBox: ~/Desktop/422178_shell_Script/LAB6

pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$ gcc -g bst_seg.c
pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$ ./a.out
Give number of nodes in binary search tree : 5
Enter node value: 3
Enter node value: 4
Enter node value: 1
Enter node value: 5
Enter node value: 2
To find sucessor of : 4
Segmentation fault (core dumped)
pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$
```

```
pavan422178@pavan422178-VirtualBox: ~/Desktop/422178_shell_Script/LAB6

pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$ gcc -g bst_seg.c
pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$ gdb ./a.out
GNU gdb (Ubuntu 12.1-0ubuntu1~22.04) 12.1
Copyright (C) 2022 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:
<https://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) run
Starting program: /home/pavan422178/Desktop/422178_shell_Script/LAB6/a.out
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Give number of nodes in binary search tree : 5
Enter node value: 3
Enter node value: 4
Enter node value: 5
Enter node value: 1
Enter node value: 2
To find sucessor of : 4

Program received signal SIGSEGV, Segmentation fault.
0x0000555555552fb in findmin (root=0x0) at bst_seg.c:42
42      return root->data;
```

```
pavan422178@pavan422178-VirtualBox: ~/Desktop/422178_shell_Script/LAB6

42      return root->data;
(gdb) list
37      }
38
39      int findmin(struct BSTNode *root){
40          while(root!=NULL)
41              root=root->left;
42          return root->data;
43      }
44
45      int inordersuccessor(struct BSTNode *root, int key){
46          struct BSTNode *succ=NULL;
(gdb) break 81
Breakpoint 1 at 0x555555555460: file bst_seg.c, line 81.
(gdb) break 49
Breakpoint 2 at 0x555555555334: file bst_seg.c, line 49.
(gdb) break 40
Breakpoint 3 at 0x5555555552e2: file bst_seg.c, line 40.
(gdb) break 41
Breakpoint 4 at 0x5555555552e4: file bst_seg.c, line 41.
(gdb) break 42
Breakpoint 5 at 0x5555555552f7: file bst_seg.c, line 42.
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/pavan422178/Desktop/422178_shell_Script/LAB6/a.out
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Give number of nodes in binary search tree : 5
Enter node value: 3
Enter node value: 4
Enter node value: 5
Enter node value: 1
Enter node value: 2
To find sucessor of : 4
```

```
pavan422178@pavan422178-VirtualBox: ~/Desktop/422178_shell_Script/LAB6

Breakpoint 1, main () at bst_seg.c:81
81      int succ=lnordersuccessor(root, key);
(gdb) print succ
$1 = 0
(gdb) print root
$2 = (struct BSTNode *) 0x55555559ac0
(gdb) print key
$3 = 4
(gdb) next

Breakpoint 2, lnordersuccessor (root=0x55555559ae0, key=4) at bst_seg.c:49
49      return findmin(root->right);
(gdb) print root->right
$4 = (struct BSTNode *) 0x55555559b00
(gdb) next

Breakpoint 3, findmin (root=0x55555559b00) at bst_seg.c:40
40      while(root!=NULL)
(gdb) print root
$5 = (struct BSTNode *) 0x55555559b00
(gdb) next

Breakpoint 4, findmin (root=0x55555559b00) at bst_seg.c:41
41      root=root->left;
(gdb) print root->left
$6 = (struct BSTNode *) 0x0
(gdb) print root->left->data
Cannot access memory at address 0x0
(gdb) next
40      while(root!=NULL)
(gdb) print root
$7 = (struct BSTNode *) 0x0
(gdb) continue
Continuing.
```

```
pavan422178@pavan422178-VirtualBox: ~/Desktop/422178_shell_Script/LAB6

Breakpoint 5, findmin (root=0x0) at bst_seg.c:42
42      return root->data;
(gdb) print root->data
Cannot access memory at address 0x0
(gdb) next

Program received signal SIGSEGV, Segmentation fault.
0x000055555552fb in findmin (root=0x0) at bst_seg.c:42
42      return root->data;
```

```
pavan422178@pavan422178-VirtualBox: ~/Desktop/422178_shell_Script/LAB6

Dump of assembler code for function main:
0x00005555555390 <+0>:    endbr64
0x00005555555394 <+4>:    push    %rbp
0x00005555555395 <+5>:    mov     %rsp,%rbp
0x00005555555398 <+8>:    sub     $0x20,%rsp
0x0000555555539c <+12>:   mov     %fs:0x28,%rax
0x000055555553a5 <+21>:   mov     %rax,-0x8(%rbp)
0x000055555553a9 <+25>:   xor     %eax,%eax
0x000055555553ab <+27>:   lea     0xc5e(%rip),%rax    # 0x55555556010
0x000055555553b2 <+34>:   mov     %rax,%rdi
0x000055555553b5 <+37>:   mov     $0x0,%eax
0x000055555553ba <+42>:   call    0x55555555090 <printf@plt>
0x000055555553bf <+47>:   lea     -0x1c(%rbp),%rax
0x000055555553c3 <+51>:   mov     %rax,%rsi
0x000055555553c6 <+54>:   lea     0xc71(%rip),%rax    # 0x5555555603e
0x000055555553cd <+61>:   mov     %rax,%rdi
0x000055555553d0 <+64>:   mov     $0x0,%eax
0x000055555553d5 <+69>:   call    0x555555550b0 <__isoc99_scanf@plt>
0x000055555553da <+74>:   movq    $0x0,-0x10(%rbp)
0x000055555553e2 <+82>:   jmp     0x55555555424 <main+148>
0x000055555553e4 <+84>:   lea     0xc56(%rip),%rax    # 0x55555556041
0x000055555553eb <+91>:   mov     %rax,%rdi
0x000055555553ee <+94>:   mov     $0x0,%eax
0x000055555553f3 <+99>:   call    0x55555555090 <printf@plt>
0x000055555553f8 <+104>:  lea     -0x18(%rbp),%rax
0x000055555553fc <+108>:  mov     %rax,%rsi
0x000055555553ff <+111>:  lea     0xc38(%rip),%rax    # 0x5555555603e
0x00005555555406 <+118>:  mov     %rax,%rdi
0x00005555555409 <+121>:  mov     $0x0,%eax
0x0000555555540e <+126>:  call    0x555555550b0 <__isoc99_scanf@plt>
0x00005555555413 <+131>:  mov     -0x18(%rbp),%edx
0x00005555555416 <+134>:  lea     -0x10(%rbp),%rax
0x0000555555541a <+138>:  mov     %edx,%esi
0x0000555555541c <+140>:  mov     %rax,%rdi
0x0000555555541f <+143>:  call    0x555555551a9 <insertnode>
0x00005555555424 <+148>:  mov     -0x1c(%rbp),%eax
0x00005555555427 <+151>:  lea     -0x1(%rax),%edx
0x0000555555542a <+154>:  mov     %edx,-0x1c(%rbp)
--Type <RET> for more, q to quit, c to continue without paging--
```

```
pavan422178@pavan422178-VirtualBox: ~/Desktop/422178_shell_Script/LAB6
pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$ gcc -g bst_seg.c
pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$ ./a.out
Give number of nodes in binary search tree : 5
Enter node value: 3
Enter node value: 4
Enter node value: 5
Enter node value: 1
Enter node value: 2
To find sucessor of : 4
Successor: 5
pavan422178@pavan422178-VirtualBox:~/Desktop/422178_shell_Script/LAB6$
```