Task 1E: Operation with double pointers. Learn to modify the data using a double pointer.

```c
// observe layout of data in memory using gdb

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



int main()
{
int x=20;
int *a=&x;
int **b=&a;
**b=(**b)*10;
printf("changed value of x=%d\n",x);
printf("changed value of x in pointer a=%d\n",*a);
printf("changed value of x in pointer b=%d\n",**b);
printf("Address of x %p\n",&a);
printf("Address of x in pointer a %p\n",a);
printf("Address of x in double pointer b %p\n",*b);
return 0;
}
```

```
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./prog1...
(gdb) b main
Breakpoint 1 at 0x1175: file program1.c, line 9.
(gdb) watch *a
No symbol "a" in current context.
(gdb) run
Starting program: /home/student/prog1
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 1, main () at program1.c:9
9       {
(gdb) next
10      int x=20;
(gdb) next
11      int *a=&x;
(gdb) next
12      int **b=&a;
(gdb) watch *a
Hardware watchpoint 2: *a
(gdb) watch **b
Hardware watchpoint 3: **b
(gdb) next
13      **b=(**b)*10;
(gdb) next
14      printf("changed value of x=%d\n",x);
(gdb) print x
$1 = 200
(gdb) display *a
1: *a = 200
(gdb) display **b
2: **b = 200
(gdb)
```