Assignments-1 on gdb

Name- Mehendi Sil Roll- 002311001094

a. Compile it so that it compiles with debugging symbols [using proper option]

gcc a.c -g -c -I.
gcc b.c -g -c -I.
gcc a.o b.o -o output
gdb output

```
C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1>gcc a.c -g -c -I.

C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1>gcc b.c -g -c -I.
b.c: In function 'f1':
b.c:4:5: warning: implicit declaration of function 'printf' [-Wimplicit-function-declaration]
     printf("The numbers are : ");
     ^~~~~~
b.c:4:5: warning: incompatible implicit declaration of built-in function 'printf'
b.c:4:5: note: include '<stdio.h>' or provide a declaration of 'printf'

C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1>gcc a.o b.o -o output

C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1>gdb output
GNU gdb (GDB) 7.6.1
Copyright (C) 2013 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 "mingw32".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1\output.exe...done.
```

b. Put breakpoint to function f1.

break f1

```
(gdb) break f1
Breakpoint 1 at 0x4015d6: file b.c, line 4.
```

c. Put breakpoint to line 10 of b.c

break b.c:10

```
Breakpoint 2 at 0x401613: file b.c, line 10.
(gdb) run
```

d. Run the program until it finishes. Which commands are you using to take it to completion?

run
c

```
(gdb) run
Starting program: C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1/output.exe
[New Thread 18192.0x24ec]
[New Thread 18192.0x3db4]
Enter a number between 2 and 6 (non-inclusive):
4
You have entered 4

Breakpoint 1, f1 (x=50, y=163) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 50, 163>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 1
Breakpoint 1, f1 (x=163, y=50) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 163, 50>

Breakpoint 1, f1 (x=33, y=109) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 33, 109>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 2
Breakpoint 1, f1 (x=109, y=33) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
```

```
The numbers are : < 109, 33>

Breakpoint 1, f1 (x=25, y=81) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 25, 81>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 3
Breakpoint 1, f1 (x=81, y=25) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 81, 25>

Breakpoint 1, f1 (x=20, y=65) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 20, 65>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 4
Breakpoint 1, f1 (x=65, y=20) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 65, 20>
[Inferior 1 (process 18192) exited normally]
```

e. How many times breakpoint “1” is hit in one run of the program ?

info break 1

```
(gdb) info break 1
Num     Type           Disp Enb Address    What
1       breakpoint     keep y   0x004015d6 in f1 at b.c:4
        breakpoint already hit 8 times
```

f. How many times breakpoint “2” is hit in one run of the program

info break 2

```
(gdb) info break 2
Num     Type           Disp Enb Address    What
2       breakpoint     keep y   0x00401613 in f2 at b.c:10
        breakpoint already hit 4 times
```

g. How you can see details about a breakpoint ?

info break

```
(gdb) info break
Num     Type           Disp Enb Address    What
1       breakpoint     keep y   0x004015d6 in f1 at b.c:4
        breakpoint already hit 8 times
2       breakpoint     keep y   0x00401613 in f2 at b.c:10
        breakpoint already hit 4 times
```

h. How you can see details about all breakpoints ?

info breakpoints

```
(gdb) info breakpoints
Num     Type           Disp Enb Address    What
1       breakpoint     keep y   0x004015d6 in f1 at b.c:4
        breakpoint already hit 8 times
2       breakpoint     keep y   0x00401613 in f2 at b.c:10
        breakpoint already hit 4 times
```

i. What is value of variable x in f1 when breakpoint “1” is hit for 3rd time ? How you can examine it ?

$1= 33

```
(gdb) break f1
Breakpoint 3 at 0x4015d6: file b.c, line 4.
(gdb) break b.c:10
Note: breakpoint 2 also set at pc 0x401613.
Breakpoint 4 at 0x401613: file b.c, line 10.
(gdb) run
Starting program: C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1/output.exe
[New Thread 16972.0x1d3c]
[New Thread 16972.0x3654]
Enter a number between 2 and 6 (non-inclusive):
4
You have entered 4

Breakpoint 3, f1 (x=50, y=163) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 50, 163>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 1
Breakpoint 3, f1 (x=163, y=50) at b.c:4
4           printf("The numbers are : ");
(gdb) print x
$1 = 163
(gdb) c
Continuing.
The numbers are : < 163, 50>

Breakpoint 3, f1 (x=33, y=109) at b.c:4
4           printf("The numbers are : ");
(gdb) print x
$2 = 33
```

j. Rerun the program. put a breakpoint at function f0. list 5 lines where it has stopped with breakpoint 3 for first time.

```
(gdb) break f0
Breakpoint 5 at 0x401466: file a.c, line 6.
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y
error return ../../gdb-7.6.1/gdb/windows-nat.c:1275 was 5
Starting program: C:\Users\ADMIN\Downloads\Assignments\Assignments\assign1/output.exe
[New Thread 8336.0x4350]
[New Thread 8336.0x15e4]

Breakpoint 5, f0 (p=0x61ff18) at a.c:6
6           int x, cntr = 1;
(gdb) c
Continuing.
Enter a number between 2 and 6 (non-inclusive):
4
You have entered 4

Breakpoint 3, f1 (x=50, y=163) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 50, 163>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 1
Breakpoint 3, f1 (x=163, y=50) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 163, 50>

Breakpoint 3, f1 (x=33, y=109) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 33, 109>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
```

```
Continuing.
After operation 2
Breakpoint 3, f1 (x=109, y=33) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 109, 33>

Breakpoint 3, f1 (x=25, y=81) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 25, 81>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 3
Breakpoint 3, f1 (x=81, y=25) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 81, 25>

Breakpoint 3, f1 (x=20, y=65) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 20, 65>

Breakpoint 2, f2 (p=0x61ff14, q=0x61ff10) at b.c:10
10          *q = (*p) -(*q);
(gdb) c
Continuing.
After operation 4
Breakpoint 3, f1 (x=65, y=20) at b.c:4
4           printf("The numbers are : ");
(gdb) c
Continuing.
The numbers are : < 65, 20>
[Inferior 1 (process 8336) exited normally]
```