Q1)

```
Breakpoint 1, main () at P1.c:21
21      int main() {
(gdb) n
25          printf("Enter the string\n");
(gdb) info r lr sp
lr             0xff6677b4          -10061900
sp             0xfffef0f0          0xfffef0f0
(gdb)
lr             0xff6677b4          -10061900
sp             0xfffef0f0          0xfffef0f0
(gdb) n
27          scanf("%s",large_input);
(gdb) n
29          safe_function(large_input);
(gdb) info r lr sp
lr             0x105dc             67036
sp             0xfffef0f0          0xfffef0f0
(gdb) s
safe_function (input=0xfffef0f0 "harith") at P1.c:15
15          printf("In safe_function\n");
(gdb) s
17          sample_function(input);
(gdb) info r lr sp
lr             0xff6b1894          -9758572
sp             0xfffef0e0          0xfffef0e0
(gdb) s
sample_function (input=0xfffef0f0 "harith") at P1.c:5
5       void sample_function(char *input) {
(gdb) n
9           strcpy(buffer, input);
(gdb) n
11      }
(gdb) info r lr sp
lr             0x69726168          1769103720
sp             0xfffef0c0          0xfffef0c0
(gdb) n
safe_function (input=0xfffef0f0 "harith") at P1.c:19
19      }
(gdb) info r lr sp
lr             0x69726168          1769103720
sp             0xfffef0e0          0xfffef0e0
(gdb) n
main () at P1.c:31
31          printf("Main function complete.\n");
(gdb) info r lr sp
lr             0x69726168          1769103720
sp             0xfffef0f0          0xfffef0f0
(gdb) n
33          return 0;
(gdb) info r lr sp
lr             0xff6b1894          -9758572
sp             0xfffef0f0          0xfffef0f0
(gdb) n
35      }
```

```
l209@admin:~/Desktop/CS23I1027$ qemu-arm -g 8080 -L /usr/arm-linux-gnueabi ./main
Enter the string
harith
In safe_function
Main function complete.
```

Q2)
If input is less than 10 characters:

```
l209@admin:~/Desktop/CS23I1027$ qemu-arm -g 8080 -L /usr/arm-linux-gnueabi ./main
Enter the string
abcd
```

```
(gdb) x/20x $sp
0xfffef0f0:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef100:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef110:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef120:     0xff7aa000      0xf63d4e2e      0x00000000      0xff7ccd14
0xfffef130:     0xff7a9080      0xff68151c      0xff7a7e14      0x00010654
(gdb) n
29          safe_function(large_input);
(gdb) x/20x $sp
0xfffef0f0:     0x64636261      0x00000000      0x00000000      0x00000000
0xfffef100:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef110:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef120:     0xff7aa000      0xf63d4e2e      0x00000000      0xff7ccd14
0xfffef130:     0xff7a9080      0xff68151c      0xff7a7e14      0x00010654
```

```
l209@admin:~/Desktop/CS23I1027$ qemu-arm -g 8080 -L /usr/arm-linux-gnueabi ./main
Enter the string
abcd
In safe_function
Main function complete.
```

If input is more than 10 characters:
It will overflow the buffer, corrupting the stack.

```
l209@admin:~/Desktop/CS23I1027$ qemu-arm -g 8080 -L /usr/arm-linux-gnueabi ./main
Enter the string
abcdefghijklmnopqrst
```

```
Breakpoint 1, main () at P1.c:21
21      int main() {
(gdb) n
25          printf("Enter the string\n");
(gdb) n
27          scanf("%s",large_input);
(gdb) info r
r0             0x11                17
r1             0x0                 0
r2             0xff7a7660          -8751520
r3             0x1                 1
r4             0x1062c             67116
r5             0x0                 0
r6             0x1042c             66604
r7             0x0                 0
r8             0x0                 0
r9             0x0                 0
r10            0xff7ee000          -8462336
r11            0xfffef15c          -69284
r12            0x498               1176
sp             0xfffef0f0          0xfffef0f0
lr             0xff6b1894          -9758572
pc             0x105cc             0x105cc <main+36>
cpsr           0x60000010          1610612752
```

```
(gdb) x/20x $sp
0xfffef0f0:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef100:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef110:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef120:     0xff7aa000      0xf63d4e2e      0x00000000      0xff7ccd14
0xfffef130:     0xff7a9080      0xff68151c      0xff7a7e14      0x00010654
(gdb)
```

```
(gdb) n
29          safe_function(large_input);
(gdb) x/20x $sp
0xfffef0f0:     0x64636261      0x68676665      0x6c6b6a69      0x706f6e6d
0xfffef100:     0x74737271      0x00000000      0x00000000      0x00000000
0xfffef110:     0x00000000      0x00000000      0x00000000      0x00000000
0xfffef120:     0xff7aa000      0xf63d4e2e      0x00000000      0xff7ccd14
0xfffef130:     0xff7a9080      0xff68151c      0xff7a7e14      0x00010654
(gdb)
```

```
l209@admin:~/Desktop/CS23I1027$ qemu-arm -g 8080 -L /usr/arm-linux-gnueabi ./main
Enter the string
abcdefghijklmnopqrst
In safe_function
*** stack smashing detected ***: terminated
```

```
sample_function (input=0xfffef0f0 "abcdefghijklmnopqrst") at P1.c:5
5       void sample_function(char *input) {
(gdb) n
9           strcpy(buffer, input);
(gdb) x/20x $sp
0xfffef0c0:     0x0001062c      0xfffef0f0      0x0001042c      0x00000000
0xfffef0d0:     0x00000000      0x00020f0c      0xfffef0ec      0x00010598
0xfffef0e0:     0x000106c8      0xfffef0f0      0xfffef15c      0x000105e8
0xfffef0f0:     0x64636261      0x68676665      0x6c6b6a69      0x706f6e6d
0xfffef100:     0x74737271      0x00000000      0x00000000      0x00000000
(gdb) n
11      }
(gdb) n

Program received signal SIGABRT, Aborted.
0xff67e4cc in ?? ()
(gdb)
```

Q3)

```
Breakpoint 1, main () at P1.c:21
21      int main() {
(gdb) n
25          printf("Enter the string\n");
(gdb) n
27          scanf("%s",large_input);
(gdb) n
29          safe_function(large_input);
(gdb) s
safe_function (input=0xfffef0f0 "Harith") at P1.c:15
15          printf("In safe_function\n");
(gdb) x/s 0xfffef0f0
0xfffef0f0:     "Harith"
(gdb)
```

```
l209@admin:~/Desktop/CS23I1027$ qemu-arm -g 8080 -L /usr/arm-linux-gnueabi ./main
Enter the string
Harith
In safe_function
Main function complete.
```

Q4)

We can replace the number of characters copied:

strncpy(buffer, input, sizeof(buffer) – 1);

Also ensure the null termination of the *buffer* array:

buffer[sizeof(buffer) - 1] = '\0';

Finally, Limit the size of characters inputted in *scanf* statement to prevent buffer overflow:

scanf("%99s", large_input);

(99 because the large input can take 100 characters.)

New Code:

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

void sample_function(char *input) {
    char buffer[10];
    strncpy(buffer, input, sizeof(buffer) - 1);
    buffer[sizeof(buffer) - 1] = '\0';
}

void safe_function(char *input) {
    printf("In safe_function\n");
    sample_function(input);
}

int main() {
    char large_input[100];
    printf("Enter the string\n");
    scanf(" %99s", large_input);
    safe_function(large_input);
    printf("Main function complete.\n");

    return 0;
}
```

Error Found:

The error is a **buffer overflow error**, which is occured when the input data is given beyond the bounds of a fixed-size buffer. This causes corruption of adjacent memory, leading to stack smashing, which then ultimately leads to unexpected program termination.