HERITAGE INSTITUTE OF TECHNOLOGY

Stream :Computer Science & Engineering

Paper : Software Tools Lab

Paper Code : CSEN 2152

Date of Examination : 27/07/2021

Full Signature of Student : Soumyadip Ghosh

Roll No.	1	2	6	1	9	0	0	1	1	5	8	
Reg. No.	0	2	8	8	3	0						
of	2019..... 2020.....											

Q. No.	Marks	Q. No.	Marks
1.		7.	
2.		8.	
3.		9.	
4.		10.	
5.		11.	
6.		12.	
Total Marks			
Total in Words			
Signature			

(REGULAR) **ADMIT**

Name of the Candidate : SOUMYADIP GHOSH

Son / Daughter of : MR. SUDIPTA GHOSH

Registration Number : 028830 **of** **2019 - 2020**

Roll Number : 12619001158 **Session : 2019 - 2023**

For **3rd** Semester **B.Tech** Examination **2021** in **Computer Science & Engineering**

Compulsory Papers		**Elective Papers**
HMTS 2001	Human Values & Professional Ethics	
MATH 2111	Probability & Statistical Methods	
ECEN 2101	Analog Circuits	
ECEN 2104	Digital Logic	
CSEN 2101	Data Structure & Algorithms	
CSEN 2102	Discrete Mathematics	
ECEN 2154	Digital Logic Lab	
CSEN 2151	Data Structure & Algorithms Lab	
CSEN 2152	Software Tools Lab	

CONTROLLER OF EXAMINATION

Note: This Admit Card is valid subject to Rules & Regulations as per Heritage Institute of Technology (An Autonomous Institute under Maulana Abul Kalam Azad University of Technology) Act regarding Examinations.

Payment Mode: DEBIT_CARD Transaction Status: Success Transaction No: 21022890450225

Problem Statement:

12. Run the following program and see the type of memcheck error using valgrind. Explain the errors.

```
#include <stdlib.h>
#include <stdint.h>

int main() {
   char* alphabet = calloc(26, sizeof(char));

  for(uint8_t i = 0; i < 26; i++) {
      *(alphabet + i) = 'A' + i;
   }
   *(alphabet + 26) = '\0';

   free(alphabet);
   return 0;
}
```

Procedure followed:

- Compiling the program:
  gcc -g p1.c -o p1
- Using Valgrind:
  valgrind –tool=memcheck ./p1

Output Screen:

```
soumyadipghosh@SRG:~/softwaretool/exam$ valgrind --tool=memcheck ./p1
==32== Memcheck, a memory error detector
==32== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==32== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==32== Command: ./p1
==32==
==32== error calling PR_SET_PTRACER, vgdb might block
==32== Invalid write of size 1
==32==    at 0x1091BC: main (p1.c:10)
==32==  Address 0x4a4705a is 0 bytes after a block of size 26 alloc'd
==32==    at 0x483DD99: calloc (in /usr/lib/x86_64-linux-gnu/valgrind/vgpreload_memcheck-amd64-linux.so)
==32==    by 0x109183: main (p1.c:5)
==32==
==32==
==32== HEAP SUMMARY:
==32==     in use at exit: 0 bytes in 0 blocks
==32==   total heap usage: 1 allocs, 1 frees, 26 bytes allocated
==32==
==32== All heap blocks were freed -- no leaks are possible
==32==
==32== For lists of detected and suppressed errors, rerun with: -s
==32== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
soumyadipghosh@SRG:~/softwaretool/exam$
```

Error faced:

Here using valgrind we can see that there is an ‘Invalid write of size 1’ in line number 10. It says further that

The address is 0 bytes after a block of size 26 that was allocated. Now inspecting line number 10 we can see that we are assigning a null character in 26th index of the string. However, in line number 5, we have only allocated the size of string as 26 using calloc. Hence when we try to assign something in 26th index, we are trying to assign something beyond memory bounds.

Error correction:

We can correct the problem by assigning the space of 27 in calloc instead of 26.

Corrected code:

soumyadipghosh@SRG: ~/softwaretool/exam

```
  GNU nano 4.8
#include <stdlib.h>
#include <stdint.h>

int main() {
    char* alphabet = calloc(27, sizeof(char));

    for(uint8_t i = 0; i < 26; i++) {
        *(alphabet + i) = 'A' + i;
    }
    *(alphabet + 26) = '\0';

    free(alphabet);
    return 0;
}
```

Output screen after correction:

```
soumyadipghosh@SRG:~/softwaretool/exam$ nano p2.c
soumyadipghosh@SRG:~/softwaretool/exam$ nano p1.c
soumyadipghosh@SRG:~/softwaretool/exam$ gcc -g p1.c -o p1
soumyadipghosh@SRG:~/softwaretool/exam$ valgrind --tool=memcheck ./p1
==40== Memcheck, a memory error detector
==40== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==40== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==40== Command: ./p1
==40==
==40== error calling PR_SET_PTRACER, vgdb might block
==40==
==40== HEAP SUMMARY:
==40==     in use at exit: 0 bytes in 0 blocks
==40==   total heap usage: 1 allocs, 1 frees, 27 bytes allocated
==40==
==40== All heap blocks were freed -- no leaks are possible
==40==
==40== For lists of detected and suppressed errors, rerun with: -s
==40== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
soumyadipghosh@SRG:~/softwaretool/exam$
```