IN-LAB

7. 1

/\* It prints the addresses of all the segments and the address of variables residing in their respective segments. segments-during-runtime.c \*/

#include<stdio.h>

#include<malloc.h>

int glb\_uninit; /\* Part of BSS Segment -- global uninitialized variable, at

runtime it is initialized to zero \*/

int glb\_init = 10; /\* Part of DATA Segment -- global initialized variable \*/

void foo(void)

{

static int num = 0; /\* stack frame count \*/

int autovar; /\* automatic variable/Local variable \*/

int \*ptr\_foo = (int\*)malloc(sizeof(int)); /\* allocate memory using malloc\*/

if (++num == 4) /\* Creating four stack frames \*/

return;

printf("Stack frame number %d: address of autovar: %p\n", num, & autovar);

printf("Address of heap allocated inside foo() %p\n",ptr\_foo);

foo();/\* function call \*/

}

int main()

{

char \*p, \*b, \*nb;

int \*ptr\_main = (int\*)malloc(sizeof(int)); /\* allocate memory using malloc\*/

printf("Text Segment:\n");

printf("Address of main: %p\n", main);

printf("Address of afunc: %p\n",foo);

printf("Stack Locations:\n");

foo();

printf("Data Segment:\n");

printf("Address of glb\_init: %p\n", & glb\_init);

printf("BSS Segment:\n");

printf("Address of glb\_uninit: %p\n", & glb\_uninit);

printf("Heap Segment:\n");

printf("Address of heap allocated inside main() %p\n",ptr\_main);

return 0;

}

/\* OUTPUT:

Text Segment:

Address of main: 0x4006b4

Address of afunc: 0x400626

Stack Locations:

Stack frame number 1: address of autovar: 0x7fffa5e3fefc

Address of heap allocated inside foo() 0x2137440

Stack frame number 2: address of autovar: 0x7fffa5e3fecc

Address of heap allocated inside foo() 0x2137460

Stack frame number 3: address of autovar: 0x7fffa5e3fe9c

Address of heap allocated inside foo() 0x2137480

Data Segment:

Address of glb\_init: 0x601050

BSS Segment:

Address of glb\_uninit: 0x60105c

Heap Segment:

Address of heap allocated inside main() 0x2137010

\*/

OUTPUT:

