Batch: E1   Roll No.: 16010321007

Experiment / assignment / tutorial No. 9

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of the Staff In-charge with date

TITLE: Dynamic Memory Allocation.

AIM: Program to demonstrate dynamic memory allocation using malloc() & free () function.

_____

**Expected OUTCOME of Experimen
To allocate the memory**

_____ **Books/
Journals/ Websites referred:**

1. Programming in C, second edition, Pradeep Dey and Manas Ghosh, Oxford University Press.
2. Programming in ANSI C, fifth edition, E Balagurusamy, Tata McGraw Hill.
3. Introduction to programming and problem solving , G. Michael Schneider ,Wiley India edition.
4. **http://cse.iitkgp.ac.in/~rkumar/pds-vlab/**

_____

Problem Definition:

Department of Science and Humanities

Create memory for int, char and float variables at run time using malloc() function and before exiting the program release the memory allocated at run time by using free() function.

Algorithm:

Step 1: Start

Step2: Declare an integer pointer

Step3: Declare an char pointer

Step4: Declare an float pointer

Step5: Allocating memory to each pointer using dynamic memory allocation.

Step6: Terminating if value is Null.

Step7: Storing the entered value by user in pointers.

Step8: Using free function to erase the memory in space.

Implementation details:

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

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




// main function int
main()
{
//In this program we will create memory for integer, char and float
// variables at run time using malloc() function
// we will release the allocated memory using free() function.
```

```c
// Declare an integer pointer int
*ptr_1;



// Declare an char pointer char
*ptr_2;



// Declare an float pointer float
*ptr_3;



// Now allocating memory to each pointer
// using dynamic memory allocation
ptr_1 = (int*)malloc(1*sizeof(int)); ptr_2
= (char*)malloc(1*sizeof(char)*1);
ptr_3 = (float*)malloc(1*sizeof(float));
  if (ptr_1==NULL && ptr_2==NULL && ptr_3==NULL)
  {
    printf("Memory not allocated");
exit(0);  }   else {



printf("\nEnter the value for integer pointer : ");
scanf("%d",ptr_1);



printf("\nEnter the value for char pointer : ");
scanf(" %c",ptr_2);



printf("\nEnter the value for float pointer : ");
scanf("%f",ptr_3);
```

```
printf("\nThe value stored in integer pointer is : %d",*ptr_1);
printf("\nThe value stored in char pointer is : %c",*ptr_2);
printf("\nThe value stored in float pointer is : %f",*ptr_3);



free(ptr_1); free(ptr_2);
free(ptr_3);
 }



return 0;
}
```

Output(s):

Conclusion:

As per the user the value of integer pointer, char pointer and float pointer was accepted and printed.

Post Lab Descriptive Questions.

1. **What is the difference between malloc and calloc?**

   Ans: malloc() and calloc() functions are used for dynamic memory allocation in the C programming language. The main difference between the malloc() and calloc() is that calloc() always requires two arguments and malloc() requires only one. i*t would be better to use malloc over calloc, unless we want the zero-initialization because malloc is faster than calloc. So if we just want to copy some stuff or do something that doesn't require filling of the blocks with zeros, then malloc would be a better choice.*

2. **Consider the following C code. What will be the output?**

```
# include<stdio.h> #
include<stdlib.h>
void fun(int *a)
{
 a = (int*)malloc(sizeof(int));
} int main() {   int
*p;  fun(p);  *p =
6;
printf("%d\n",*p);
return(0);
}
```

(A) Compiler Error
(B) 6
(C) Runtime Error
(D) Garbage Value

Ans: C runtime error

Department of Science and Humanities

3. Difference between Static and Dynamic Memory allocation

ANS: Static Memory Allocation is done before program execution. Dynamic Memory Allocation is done during program execution. In static memory allocation, once the memory is allocated, the memory size cannot change. In dynamic memory allocation, when memory is allocated the memory size can be changed.

Static Memory Allocation: Static Memory is allocated for declared variables by the compiler. The address can be found using the *address of* operator and can be assigned to a pointer. The memory is allocated during compile time.

Dynamic Memory Allocation: Memory allocation done at the time of execution(run time) is known as dynamic memory allocation. Functions calloc() and malloc() support allocating dynamic memory. In the Dynamic allocation of memory space is allocated by using these functions when the value is returned by functions and assigned to pointer variables.

Department of Science and Humanities

Date: 14/01/2022   **Signature of faculty in-charge**

Department of Science and Humanities