



॥वसुधैव कुटुम्बकम्॥

## SYMBIOSIS INSTITUTE OF TECHNOLOGY, PUNE

Constituent of Symbiosis International (Deemed University), Pune

Assignment No.: 14	
Course Name	Programming in C Lab
Name of Student	Faheemuddin Sayyed
PRN No.	23070122196
Branch	CSE
Class	C-1
Academic Year & Semester	2023-2024 & Semester 2
Date of Performance	19/04/2024
Assignment Title (Full):	1. Write a program to demonstrate compile time memory allocation 2. Write a program to demonstrate runtime memory allocation
<p>Theory: (Note: According to the assignment title, please write the background information as an introduction, then write the steps/logic/process/algorithm of the C program in the Journal Notebook, and add its screenshot in the below theory response.)</p>	
<p><b>Theory Response:</b></p> <p><b>Compile Time Memory Allocation:</b></p> <ol style="list-style-type: none"><li>1. Define a function Display to print elements of an integer array.</li><li>2. In main, declare an integer array A of size 7.</li><li>3. Display the array before user input using Display(A, 7).</li><li>4. Prompt user to enter elements for A using scanf.</li><li>5. Display the modified array using Display(A, 7).</li></ol> <p><b>Runtime Memory Allocation:</b></p> <ol style="list-style-type: none"><li>1. Define Display and Insert functions for printing and reading array elements respectively.</li><li>2. Declare variables n1, n2, *p1, *p2 for array size and pointers.</li><li>3. Prompt user to enter the size (n1) of the array.</li><li>4. Allocate memory for p1 using malloc and display its contents.</li><li>5. Input elements for p1 using Insert.</li><li>6. Allocate memory for p2 using calloc with the same size as p1.</li><li>7. Display and input elements for p2 as done for p1.</li><li>8. Prompt user to enter a new size (n2) for p1.</li><li>9. Reallocate memory for p1 using realloc based on n2.</li><li>10. Display and input elements for the reallocated p1.</li><li>11. Free allocated memory for p1 and p2 using free.</li></ol>	

Output: (Note: Execute the C program as per the assignment title, take an input code and output result screenshot with the date and time from your computer, and add its screenshot in the below output response.)

### Output Response:

1. Write a program to demonstrate compile time memory allocation

```
1  #include <stdio.h>
2
3  void Display(int arr[],int n){
4      printf("\nElements of the array are:\n");
5      for(int i=0;i<n;i++){
6          printf("%d ",arr[i]);
7      }
8      printf("\n");
9  }
10
11 int main(){
12     int A[7];
13     printf("\nDefault Initialization:\n");
14     Display(A,7);
15     printf("\nEnter the elements of the array:\n");
16     for(int i=0;i<7;i++){
17         scanf("%d",&A[i]);
18     }
19     Display(A,7);
20     return 0;
21 }
```

Default Initialization:

Elements of the array are:

1 1868672816 1 -2078766408 170754049 8289868 1

Enter the elements of the array:

1 2 3 4 5 6 7

Elements of the array are:

1 2 3 4 5 6 7

○ (base) fahee@Faheems-MacBook-Pro Programming\_in\_C %

2. Write a program to demonstrate runtime memory allocation:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  void Display(int arr[],int n){
5      printf("\nElements of the array are:\n");
6      for(int i=0;i<n;i++){
7          printf("%d ",arr[i]);
8      }
9      printf("\n");
10 }
11
12 void Insert(int p[],int n){
13     printf("\nEnter the elements of the array:\n");
14     for(int i=0;i<n;i++){
15         scanf("%d",&p[i]);
16     }
17 }
18
19 int main(){
20     int n1,n2,*p1,*p2;
21     printf("\nPlease enter the size of the array: ");
22     scanf("%d",&n1);
23
24     p1=malloc(n1*sizeof(int));
25     printf("\n-----MALLOC-----\n");
26     printf("\nDefault Initialization:\n");
27     Display(p1,n1);
28     Insert(p1,n1);
29     Display(p1,n1);
30
31     p2=calloc(n1,n1*sizeof(int));
32     printf("\n-----CALLOC-----\n");
33     printf("\nDefault Initialization:\n");
34     Display(p2,n1);
35     Insert(p2,n1);
36     Display(p2,n1);
37
38     printf("\nEnter the modified size of the array: ");
39     scanf("%d",&n2);
40     p1=realloc(p1,n2*sizeof(int));
41     printf("\n-----REALLOC-----\n");
42     printf("\nDefault Initialization:\n");
43     Display(p1,n2);
44     Insert(p1,n2);
45     Display(p1,n2);
46
47     free(p1);
48     free(p2);
49     return 0;
50 }
```

Please enter the size of the array: 5

-----MALLOC-----

Default Initialization:

Elements of the array are:

0 0 0 0 0

Enter the elements of the array:

10 20 30 40 50

Elements of the array are:

10 20 30 40 50

-----CALLOC-----

Default Initialization:

Elements of the array are:

0 0 0 0 0

Enter the elements of the array:

1 2 3 4 5

Elements of the array are:

1 2 3 4 5

Enter the modified size of the array: 7

-----REALLOC-----

Default Initialization:

Elements of the array are:

10 20 30 40 50 0 0

Enter the elements of the array:

10 20 30 40 50 60 70

Elements of the array are:

10 20 30 40 50 60 70

› (base) fahee@Faheems-MacBook-Pro Programming\_in\_C %

Conclusion: (Note: Write the key findings or outcome from this assignment, enlist their potential real-world applications in Journal Notebook, and add its screenshot in the below conclusion response.)

**Conclusion Response:**

The first program demonstrates compile-time memory allocation using a statically declared array, while the second program demonstrates runtime memory allocation using dynamic memory allocation functions (malloc, calloc, realloc) to allocate memory during program execution. Dynamic allocation allows flexibility in managing memory based on runtime conditions, improving efficiency and adaptability in memory usage. However, it requires explicit memory management to avoid memory leaks.

Please note that assignment content can be readable.

**Faculty Name:**

Dr. Kanhaiya Sharma  
Prof. Mahesh Arse  
Prof. Sachin R. Gaikwad  
Prof. Surabhi Thatte