Difference between Static and Dynamic Memory Allocation in C

**Memory Allocation:** Memory allocation is a process by which computer programs and services are assigned with physical or virtual memory space. The memory allocation is done either before or at the time of program execution. There are two types of memory allocations:

1. [Compile-time or Static Memory Allocation](https://www.geeksforgeeks.org/difference-between-static-allocation-and-heap-allocation/)
2. [Run-time or Dynamic Memory Allocation](https://www.geeksforgeeks.org/what-is-dynamic-memory-allocation/)

**Static Memory Allocation:** Static Memory is allocated for declared variables by the compiler. The address can be found using the [*address of*](https://www.geeksforgeeks.org/address-function-c-cpp/) 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()](https://www.geeksforgeeks.org/dynamic-memory-allocation-in-c-using-malloc-calloc-free-and-realloc/) 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.

**Tabular Difference Between Static and Dynamic Memory Allocation in C:**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Static Memory Allocation** | **Dynamic Memory Allocation** |
| 1 | In the static memory allocation, variables get allocated permanently. | In the Dynamic memory allocation, variables get allocated only if your program unit gets active. |
| 2 | Static Memory Allocation is done before program execution. | Dynamic Memory Allocation is done during program execution. |
| 3 | It uses [stack](http://www.geeksforgeeks.org/stack-data-structure/) for managing the static allocation of memory | It uses [heap](https://www.geeksforgeeks.org/heap-data-structure/) for managing the dynamic allocation of memory |
| 4 | It is less efficient | It is more efficient |
| 5 | In Static Memory Allocation, there is no memory re-usability | In Dynamic Memory Allocation, there is memory re-usability and memory can be freed when not required |
| 6 | In static memory allocation, once the memory is allocated, the memory size can not change. | In dynamic memory allocation, when memory is allocated the memory size can be changed. |
| 7 | In this memory allocation scheme, we cannot reuse the unused memory. | This allows reusing the memory. The user can allocate more memory when required. Also, the user can release the memory when the user needs it. |
| 8 | In this memory allocation scheme, execution is faster than dynamic memory allocation. | In this memory allocation scheme, execution is slower than static memory allocation. |
| 9 | In this memory is allocated at compile time. | In this memory is allocated at run time. |
| 10 | In this allocated memory remains from start to end of the program. | In this allocated memory can be released at any time during the program. |
| 11 | **Example:** This static memory allocation is generally used for [array](https://www.geeksforgeeks.org/introduction-to-arrays/). | **Example:** This dynamic memory allocation is generally used for [linked list](http://www.geeksforgeeks.org/data-structures/linked-list/). |