

C Dynamic Memory Allocation(DMA)

- Q. Why and when do we use DMA?
 - As you know, an array is a collection of a fixed number of values. Once the size of an array is declared, you cannot change it.
 - Sometimes the size of the array declared may be insufficient or more than required. To solve this issue, DMA came into existence.
- DMA enables the programmer to allocate memory at Runtime.
 - Following are the 4 functions built-in functions declared in <stdlib.h> file.
 1. malloc()
 2. calloc()
 3. realloc()
 4. free()

C Dynamic Memory Allocation(DMA)

- Before diving into the details of Dynamic Memory Allocation, let's first understand the difference between Static Memory Allocation and Dynamic Memory Allocation.

Q. What is meant by Static Memory Allocation and DMA?



static memory allocation

memory is allocated at compile time.

memory can't be increased while executing program.

used in array.

dynamic memory allocation

memory is allocated at run time.

memory can be increased while executing program.

used in linked list.

C Dynamic Memory Allocation(DMA)



FirstBitSolutions.com
...Learn IT Bit by Bit...

- malloc() :
 - “malloc” stands for memory allocation.
 - The malloc() function reserves a block of memory of the specified number of bytes. And, it returns a pointer of void which can be casted into pointers of any form.
 - Syntax :

```
ptr = (castType*) malloc(size);
```
 - Example:

```
ptr = (int*) malloc(50 * sizeof(int));
```

 - Here 200 bytes of memory gets allocated on heap, because size of int is 4 bytes. Hence (50*4=200) bytes gets allocated. ptr holds the address of the first byte in the allocated memory and it returns a NULL pointer if memory is not allocated.

C Dynamic Memory Allocation(DMA)



- calloc() :
 - "calloc" stands for contiguous allocation.
 - The malloc() function allocates memory and leaves the memory uninitialized, whereas the calloc() function allocates memory and initializes all bits to zero.
 - Syntax: `ptr = (castType*)calloc(n, size);`
 - Example: `ptr = (float*) calloc(25, sizeof(float));`

•C Dynamic Memory Allocation(DMA)

- free() :
 - free() is used to de-allocate the memory which is dynamically allocated on heap using malloc() and calloc().
 - free() needs to be called explicitly by the programmer.
 - Syntax: `free(ptr);`
- realloc() :
 - If the dynamically allocated memory is insufficient or more than required, you can change the size of previously allocated memory using realloc().
 - Syntax: `ptr = realloc(ptr, x);`