Dynamic Memory In C++

Dynamic memory

- Suppose you want to put a toy in a box, but you only have an approximate idea of its size. For that, you would require a box whose size is equal to the approximate size of the toy.
- We face a similar situation in C++ also when we want to input a sentence as an array of characters but are not sure about the number of characters in the array.
- Now, while declaring the character array, if we specify its size smaller than the size of the input string, then we will get an error because the space in the memory allocated to the array is lesser than the size of the input string. This is the same case as trying to fit a big toy in a smaller box. If we specify its size much larger than the size of the input string, then the array will be allocated a space in the memory which is much larger than the size of the input string, thus unnecessarily consuming more memory even when it is not required. This is like putting a small toy in a large box.
- In the above case, we don't have the idea about the size of the array until the compile time (when computer compiles the code and the string is input by the user). In such cases, we use the **new** operator.
- But before going to the new operator, let's have a look at the two par memory is divided. These parts are as follows:
- **stack -** Memory from the stack is used by all the members which are functions. Note that main is also a function.
- **heap -** This memory is unused and can be used to dynamically alloca runtime.

new

The new operator is used to allocate memory at runtime. The memory is allocated in bytes.

Let's first see how to allocate a variable dynamically.

int *ptr = new int;

By writing new int, we allocated the space in memory required by an integer. Then we assigned the address of that memory to an integer pointer ptr.

We assign value to that memory as follows:

*ptr = 4;

Thus, we allocated that much space in memory that would be required by an int a~~~~LERNEN~~~~ assigned the address of that memory to a pointer ptr and assigned the memory a ~~~

02:35   05:09