**memcpy() in C/C++**

memcpy() is used to copy a block of memory from a location to another. It is declared in **string.h**

// Copies "numBytes" bytes from address "from" to address "to"

/\* A C program to demonstrate working of memcpy \*/

#include <stdio.h>

#include <string.h>

int main ()

{

char str1[] = "Geeks";

char str2[] = "Quiz";

puts("str1 before memcpy ");

puts(str1);

/\* Copies contents of str2 to str1 \*/

memcpy (str1, str2, sizeof(str2));

puts("\nstr1 after memcpy ");

puts(str1);

return 0;

}

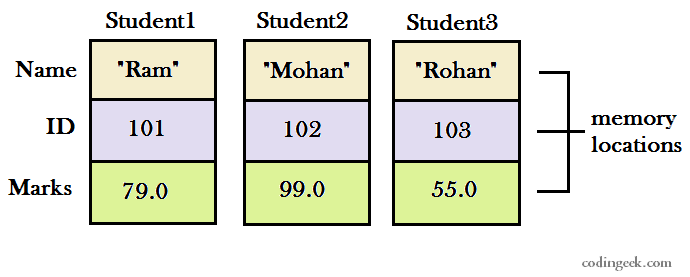
Output:

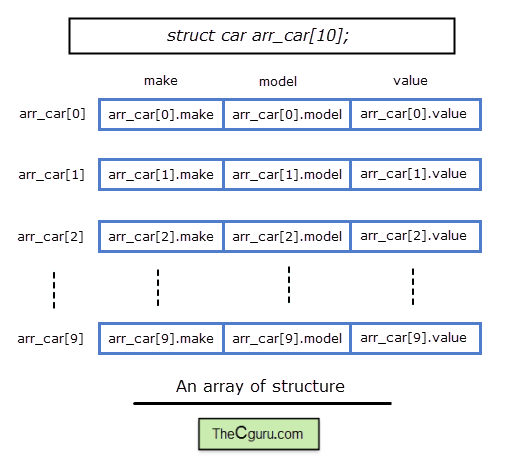
str1 before memcpy

Geeks

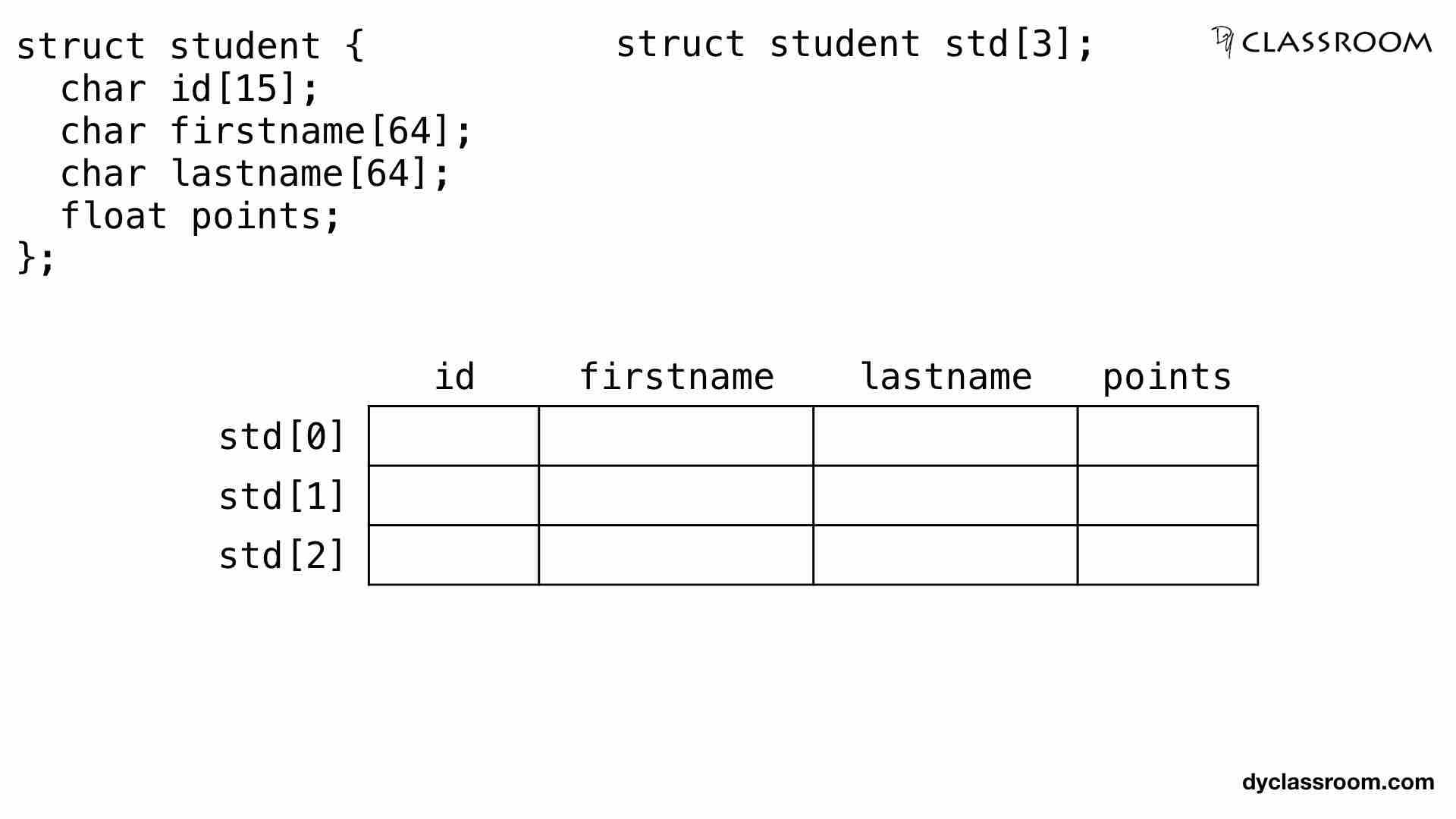
str1 after memcpy

Quiz

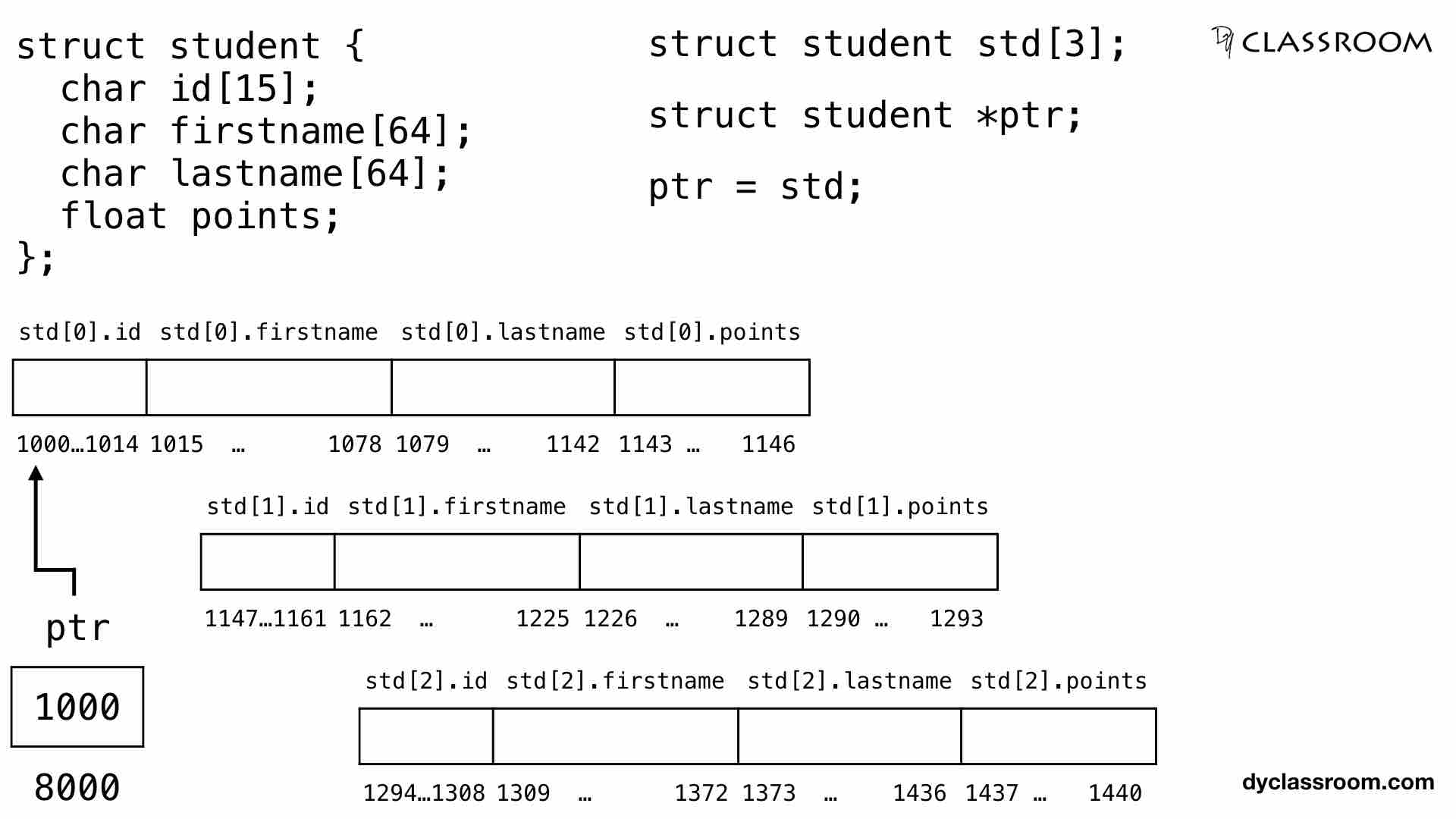




We can represent the std array variable as following.



We can represent the std array variable in memory as follows.



Points to note!

Each student data takes 147 bytes of memory.

|  |  |  |
| --- | --- | --- |
| **Member** | **Data Type** | **Size** |
| id | char | 15 bytes |
| firstname | char | 64 bytes |
| lastname | char | 64 bytes |
| points | float | 4 bytes |

And the array size is 3 so, total 147x3 i.e., 441 bytes is allocated to the std array variable.

The first element std[0] gets the memory location from 1000 to 1146.

The second element std[1] gets the memory location from 1147 to 1293.

And the third element std[2] gets the memory location from 1294 to 1440.

We start by first making the ptr pointer variable point at address 1000 which is the starting address of the first element std[0].

Then moving forward we increment the pointer ptr++ so, it points at the memory location 1147 i.e., the starting memory location of second element std[1].

Similarly, in the next run we point ptr at memory location 1294 i.e., starting location of third element std[2].

To access the members of the structure via pointe