Study notes on C Programming Language

Shuo Yang

September 29, 2014

1 Two-dimensional array

A two-dimensional array (e.g. $A[M][N]$) actually uses a contiguous storage structure underneath such that A is the address of the first element and $A[0]$ is the start address of the first row. We can view it as a one-dimensional array with each row appears in sequence. So $A = A[0] = \&A[0][0]$ and $A[1] - A[0] = N$. The type of A is actually *int* $(*)[N]$ which means A is a pointer to an array of size N, that's how *sizeof* figures out the size of A correctly. The type for $A[0]$ is *int **, because it is a pointer to a one-dimensional array. Note that a two-dimensional array is not *an array of pointers to one-dimensional array, that is, int *[]*, it is not efficient to access array elements as we will have to jump back and forth to different locations.