```c
#include <stdio.h>
#include <stdlib.h>
#define MAX 10
void main()
{
	int *ptr, *arr[MAX];
	int i, j;
	for (i=MAX-1;i>=0; i--)
	{
	    if (arr[i] = (int *) malloc(i * sizeof(int)))
	    for (j=0; j<i; j++)
	        *(*(arr+i)+j) = j*i;

	}
	ptr = *(arr+MAX-1); // &arr[9][8] fixes problem
	while (*ptr)
	    printf("%d, ", *ptr--);
}



#include <stdio.h>

int x;
char *c[] = {"DOOR", "ME", "POINTERS", "BALIL"};
char **cp[] = {c+3, c+2, c+1, c};
char ***cpp = cp;

void main(void) {
	printf("%s\n", **++cpp); // c + 2 == "POINTERS"
	printf("%s\n", ++**cpp); // (c + 2)[1]
	printf("%s\n", cpp[-1]+3); // trash, warning
	printf("%c\n", cpp[-1][-1][-1]); // P
	scanf("%d", &x);
}
```

```c
#include <stdio.h>

int *a[5]; // arr of integer pointers
int (*a)[5]; // pointer to arr of integers (not integer pointers)

int main(void) {
	int mat[2][3] = {345, 12, 23, 92, 45, 76};
	int *p1[] = {*mat + 2, *(mat + 1) + 1, *(mat + 1) - 2};
	int **p2[] = {p1 + 1, p1 + 2, p1};
	int ***p3 = p2;

	printf("%d\n", ++*--*++*p3); // 346
	printf("%d\n", *(*p3[2] - 1)); // 12
	printf("%d\n", *(p3[0][-1] + 1)); // 76
	printf("%d\n", *(*(*(p3+2) + 2))); // 346
}
```

```
int main()
{
	//		1060
	int mat2[][4] = { 14,78,90,134,67,8,29,874,54,28,-9,13 };
	int mat[][4] = {
		//  1000    1004   1008     1012
			{ 14,    78,     90,     134 },
			//  1016    1020   1024     1028
				{ 67,    8,     29,     874 },
				//  1032    1036   1040     1044      1048
					{ 54,    28,     -9,     13 }
	};
	//		int(*p)[2] = mat + 1;    //  warning
	//		int(*)[2]     int(*)[4]
	int(*p1)[4] = mat + 1;    //  p1 <======= 1016
	int  n;
	int* ar[] = { *mat, *(mat + 1), *(mat + 2) };     // array of pointers
	int** p3 = &ar[1];

	//  p1      mat+1   -----   pointer to second row
	//  int (*)[4]
	//  p1[0]   *(p1+0) *(mat+1)   -----   pointer to first element in second row
	//  int*
	//  p1[1]   *(p1+1)     -----   pointer to third row

	n = *(p1[0] + 2);   //  29
	n = *(p1[1] + 2);   //  -9
	n = *(p1[0] + 7);   //  13
	n = *(p1[0] + 8);   //  trash
	n = *(*(p1 - 1) + 7);   //  874       p1   ---  pointer to second row
	                                        //          p1-1 ---  pointer to first row
	                                        //          *(p1-1) ---  pointer to first element in first
row

	return 0;
}
```