```c
#include <stdio.h>
int main(void)
{
    //variable declaraions

    //IN-LINE INITIALIZATION
    int iArray[5][3][2] = { { { 9, 18 }, { 27, 36 }, { 45, 54 } },
                            { { 8, 16 }, { 24, 32 }, { 40, 48 } },
                            { { 7, 14 }, { 21, 28 }, { 35, 42 } },
                            { { 6, 12 }, { 18, 24 }, { 30, 36 } },
                            { { 5, 10 }, { 15, 20 }, { 25, 30 } } };
    int int_size;
    int iArray_size;
    int iArray_num_elements, iArray_width, iArray_height, iArray_depth;

    //code
    printf("\n\n");

    int_size = sizeof(int);

    iArray_size = sizeof(iArray);
    printf("Size Of Three Dimensional ( 3D ) Integer Array Is = %d\n\n",
      iArray_size);

    iArray_width = iArray_size / sizeof(iArray[0]);
    printf("Number of Rows (Width) In Three Dimensional ( 3D ) Integer Array Is =
       %d\n\n", iArray_width);

    iArray_height = sizeof(iArray[0]) / sizeof(iArray[0][0]);
    printf("Number of Columns (Height) In Three Dimensional ( 3D ) Integer Array
      Is = %d\n\n", iArray_height);

    iArray_depth = sizeof(iArray[0][0]) / int_size;
    printf("Depth In Three Dimensional ( 3D ) Integer Array Is = %d\n\n",
      iArray_depth);

    iArray_num_elements = iArray_width * iArray_height * iArray_depth;
    printf("Number of Elements In Three Dimensional ( 3D ) Integer Array Is = %d
      \n\n", iArray_num_elements);

    printf("\n\n");
    printf("Elements In Integer 3D Array : \n\n");

    // *** PIECE-MEAL DISPLAY ***
    // ****** ROW 1 ******
    printf("****** ROW 1 ******\n");
    printf("****** COLUMN 1 ******\n");
    printf("iArray[0][0][0] = %d\n", iArray[0][0][0]);
    printf("iArray[0][0][1] = %d\n", iArray[0][0][1]);
    printf("\n");

    printf("****** COLUMN 2 ******\n");
```

```c
    printf("iArray[0][1][0] = %d\n", iArray[0][1][0]);
    printf("iArray[0][1][1] = %d\n", iArray[0][1][1]);
    printf("\n");

    printf("****** COLUMN 3 ******\n");
    printf("iArray[0][2][0] = %d\n", iArray[0][2][0]);
    printf("iArray[0][2][1] = %d\n", iArray[0][2][1]);
    printf("\n\n");

    // ****** ROW 2 ******
    printf("****** ROW 2 ******\n");
    printf("****** COLUMN 1 ******\n");
    printf("iArray[1][0][0] = %d\n", iArray[1][0][0]);
    printf("iArray[1][0][1] = %d\n", iArray[1][0][1]);
    printf("\n");

    printf("****** COLUMN 2 ******\n");
    printf("iArray[1][1][0] = %d\n", iArray[1][1][0]);
    printf("iArray[1][1][1] = %d\n", iArray[1][1][1]);
    printf("\n");

    printf("****** COLUMN 3 ******\n");
    printf("iArray[1][2][0] = %d\n", iArray[1][2][0]);
    printf("iArray[1][2][1] = %d\n", iArray[1][2][1]);
    printf("\n\n");

    // ****** ROW 3 ******
    printf("****** ROW 3 ******\n");
    printf("****** COLUMN 1 ******\n");
    printf("iArray[2][0][0] = %d\n", iArray[2][0][0]);
    printf("iArray[2][0][1] = %d\n", iArray[2][0][1]);
    printf("\n");

    printf("****** COLUMN 2 ******\n");
    printf("iArray[2][1][0] = %d\n", iArray[2][1][0]);
    printf("iArray[2][1][1] = %d\n", iArray[2][1][1]);
    printf("\n");

    printf("****** COLUMN 3 ******\n");
    printf("iArray[2][2][0] = %d\n", iArray[2][2][0]);
    printf("iArray[2][2][1] = %d\n", iArray[2][2][1]);
    printf("\n\n");

    // ****** ROW 4 ******
    printf("****** ROW 4 ******\n");
    printf("****** COLUMN 1 ******\n");
    printf("iArray[3][0][0] = %d\n", iArray[3][0][0]);
    printf("iArray[3][0][1] = %d\n", iArray[3][0][1]);
    printf("\n");

    printf("****** COLUMN 2 ******\n");
    printf("iArray[3][1][0] = %d\n", iArray[3][1][0]);
```

```c
    printf("iArray[3][1][1] = %d\n", iArray[3][1][1]);
    printf("\n");

    printf("****** COLUMN 3 ******\n");
    printf("iArray[3][2][0] = %d\n", iArray[3][2][0]);
    printf("iArray[3][2][1] = %d\n", iArray[3][2][1]);
    printf("\n\n");

    // ****** ROW 5 ******
    printf("****** ROW 5 ******\n");
    printf("****** COLUMN 1 ******\n");
    printf("iArray[4][0][0] = %d\n", iArray[4][0][0]);
    printf("iArray[4][0][1] = %d\n", iArray[4][0][1]);
    printf("\n");

    printf("****** COLUMN 2 ******\n");
    printf("iArray[4][1][0] = %d\n", iArray[4][1][0]);
    printf("iArray[4][1][1] = %d\n", iArray[4][1][1]);
    printf("\n");

    printf("****** COLUMN 3 ******\n");
    printf("iArray[4][2][0] = %d\n", iArray[4][2][0]);
    printf("iArray[4][2][1] = %d\n", iArray[4][2][1]);
    printf("\n\n");

    return(0);
}
```