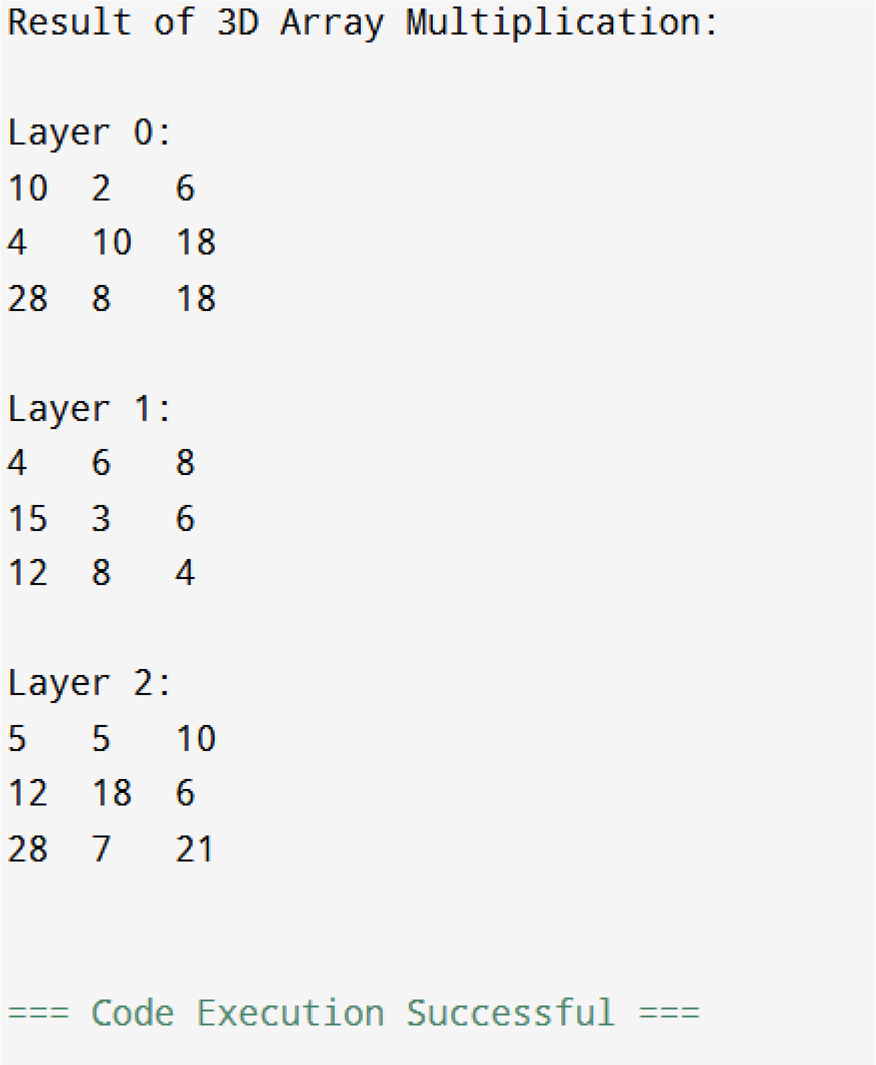
Experiment 4: 3D Matrix Mul

Code:

```
#include <stdio.h>

int main() {

    int a[3][3][3] = {

        {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}},

        {{2, 2, 2}, {3, 3, 3}, {4, 4, 4}},

        {{5, 5, 5}, {6, 6, 6}, {7, 7, 7}}

    };

    int b[3][3][3] = {

        {{10, 1, 2}, {1, 2, 3}, {4, 1, 2}},

        {{2, 3, 4}, {5, 1, 2}, {3, 2, 1}},

        {{1, 1, 2}, {2, 3, 1}, {4, 1, 3}}

    };

    int mul[3][3][3];

    int i, j, k;

    for (i = 0; i < 3; i++) {

        for (j = 0; j < 3; j++) {

            for (k = 0; k < 3; k++) {

                mul[i][j][k] = a[i][j][k] * b[i][j][k];

            }

        }

    }

    printf("Result of 3D Array Multiplication:\n");

    for (i = 0; i < 3; i++) {

        printf("\nLayer %d:\n", i);

        for (j = 0; j < 3; j++) {

            for (k = 0; k < 3; k++) {

                printf("%d\t", mul[i][j][k]);

            }

            printf("\n");
```

```
      }
    }
    return 0;
}
```

Output:



```
Result of 3D Array Multiplication:

Layer 0:
10  2   6
4   10  18
28  8   18

Layer 1:
4   6   8
15  3   6
12  8   4

Layer 2:
5   5   10
12  18  6
28  7   21


=== Code Execution Successful ===
```