#include <iostream>  
#include "include/graph.h"  
#include <cstdlib>  
  
  
int create\_matrix(GraphElementType \*\*\*matrix, int size) {  
 auto \*tmp\_mat = (GraphElementType \*) calloc(size \* size, sizeof(GraphElementType));  
 auto \*\*tmp\_mat2 = (GraphElementType \*\*) calloc(size, sizeof(GraphElementType));  
  
 if (tmp\_mat == NULL) return -1;  
 if (tmp\_mat2 == NULL) {  
 free(tmp\_mat);  
 return -1;  
 }  
 for (int i = 0; i < size; i++) {  
 tmp\_mat2[i] = &tmp\_mat[size \* i];  
 }  
 \*matrix = tmp\_mat2;  
 return 1;  
}  
  
void fill\_matrix(GraphElementType \*\*matrix, int size) {  
 for (int i = 0; i < size; i++) {  
 for (int j = 0; j < size; j++) {  
 matrix[i][j] = 1;  
 }  
 }  
}  
  
void print\_matrix(GraphElementType \*\*matrix, int size) {  
 for (int i = 0; i < size; i++) {  
 for (int j = 0; j < size; j++) {  
 std::cout << matrix[i][j] << " ";  
 }  
 std::cout << "\n";  
 }  
}  
  
  
int main() {  
// int temp[SIZE][SIZE] = {  
// {1, 0, 0, 0, 1, 0, 1, 0, 0, 0},  
// {0, 0, 0, 1, 0, 1, 1, 1, 0, 0},  
// {0, 1, 0, 0, 0, 1, 0, 0, 1, 0},  
// {0, 0, 0, 1, 0, 0, 0, 0, 0, 1},  
// {0, 0, 0, 1, 0, 1, 0, 0, 1, 0},  
// {0, 1, 0, 1, 0, 1, 0, 0, 1, 1},  
// {1, 0, 0, 0, 0, 1, 1, 1, 0, 0},  
// {0, 0, 0, 1, 0, 1, 0, 0, 0, 0},  
// {0, 0, 1, 0, 0, 1, 0, 1, 1, 0},  
// {1, 0, 0, 0, 0, 1, 1, 0, 0, 1}  
// };  
 GraphElementType \*\*g;  
 int error = create\_matrix(&g, 10);  
 if (error == -1) return EXIT\_FAILURE;  
 fill\_matrix(g, 10);  
 print\_matrix(g, 10);  
// Graph graph(g);  
}