Exemplu de folosire a unei matrice alocata dinamic folosind pointerul int ** mat.
Functia de citire face transferul datelor prin parametri, respectiv returneaza valoare. In functie se citesc dimensiunile matricei, se face alocarea de memorie si se citesc valorile elementelor matricei.
Eliberarea memoriei alocata dinamic se face prin functia free_mem().

```
/*
	Varianta I
*/

#include <conio.h>
#include <stdio.h>

int** citire(int*n, int*m)
{
	int i, j;
	int ** mat;
	printf("Citire date matrice\n");
	printf("\nnr. linii = "); scanf("%d", n);
	printf("\nnr. coloane = "); scanf("%d", m);

	// alocare memorie
	mat=new int*[*n];
	for(i=0; i<*n;i++)
		mat[i]=new int[*m];

	//citirea elementelor matricei
	  for(i=0;i<*n; i++)
	    for(j=0;j<*m; j++)
	      {    printf("mat[%d][%d]=", i, j);
		   scanf("%d",&mat[i][j]);
	      }
	return mat;
}
```

```
/*
	Varianta II
*/

#include <conio.h>
#include <stdio.h>

void citire(int ***mat, int*n, int*m)
{
	int i, j;

	printf("Citire date matrice\n");
	printf("\nnr. linii = "); scanf("%d", n);
	printf("\nnr. coloane = "); scanf("%d", m);

	// alocare memorie
	*mat=new int*[*n];
	for(i=0; i<*n;i++)
		(*mat)[i]=new int[*m];

	//citirea elementelor matricei
	for(i=0;i<*n;i++)
		for(j=0;j<*m;j++)
		{	printf("mat[%d][%d]=",i,j);
			scanf("%d",&(*mat)[i][j]);
			//sau: scanf("%d",(*mat)[i]+j);
		}
}
```

```
void afisare( int** mat, int n, int m)
{
	int i, j;
	printf("\nAfisare elemente matrice");
	for(i=0;i<n;i++)
	{	printf("\n");
		for(j=0;j<m;j++)
			printf("%5d",mat[i][j]);
	}
}

void free_mem(int*** mat, int n)
{	int i;
	printf("\nEliberare memorie\n");
	for(i=0;i<n;i++)
		delete [](*mat)[i];
	delete [](*mat);
}

int main()
{	int ** mat;
	int linii, coloane;
	mat = citire(&linii, &coloane);
	afisare(mat, linii, coloane);
	free_mem(&mat, linii);
}
```

```
void afisare( int** mat, int n, int m)
{
	int i, j;
	printf("\nAfisare elemente matrice");
	for(i=0;i<n;i++)
	{	printf("\n");
		for(j=0;j<m;j++)
			printf("%5d", mat[i][j]);
	}
}

void free(int** mat, int n)
{	int i;
	printf("\nEliberare memorie\n");
	for(i=0;i<n;i++)
		delete []mat[i];
	delete []mat;
}

int main()
{	int ** mat;
	int linii, coloane;
	citire(&mat, &linii, &coloane);
	afisare(mat, linii, coloane);
	free_mem(mat, linii);
}
```

TEMA

Sa se defineasca o functie care sa elimine o linie a matricei de pe o pozitie indicata prin parametru.
Sa se defineasca o functie care sa elimine o coloana a matricei de pe o pozitie indicata prin parametru..
Sa se defineasca o functie care sa insereze o linie pe pozitia indicata printr-un parametru. Elementele adaugate vor fi citite de la tastatura.
Sa se defineasca o functie care sa insereze o coloana pe pozitia indicata printr-un parametru. Elementele adaugate vor fi citite de la tastatura.

In functia main sa se exemplifice folosirea functiilor.
Sa se rescrie programul folosind functiile de alocare definite in fisierul malloc.h.