

Ejemplos de uso de las funciones malloc() y free()

1. Cargue un vector de números enteros, vectorNumero, con elementos que cumplan la siguiente condición:
 - a. Distintos valores del múltiplo
 - b. Vectores de distintos tamaños

```
#include <stdio.h>
#include <stdlib.h>
#define MAX 100

int main(){

    int multiplo, tamano, i, control=1;
    int *pNumero;

    printf("Ingrese el tamaño del vector: ");
    scanf("%d", &tamano);
    printf("\nIngrese un número para calcular sus múltiplos: ");
    scanf("%d", &multiplo);

    *pNumero=malloc(tamano*sizeof(int));

    for (i = 0; i < tamano ; i++)
    {
        *pNumero=i * multiplo;
        printf("%d \n", *pNumero);
        pNumero++;
    }

    free(pNumero);
    return 0;
}
```

2. Cuando se baraja un mazo de cartas, se toma el mazo completo y se divide en dos, posteriormente se juntan los dos montones en un nuevo mazo poniendo una carta del primer montón y una carta del segundo montón, y así posteriormente hasta que no quede ninguna carta en ninguno de los montones. Escriba un programa que simule el barajeo perfecto de un mazo de cartas.

```
#include<stdio.h>
#include<stdlib.h>

void leerMazoEntero(int *pMazoIni, int Tama);
void dividirMazo(int *pMazoIni, int *pMazo1, int *pMazo2, int MitadTama, int Tama);
void mezclarMazo(int *pMazoFinal, int *pMazo1, int *pMazo2, int MitadTama);
void mostrarMazo(int *pMazoFinal, int Tama);

int main(){

    int Tama, MitadTama;
    int *pMazoIni, *pMazo1, *pMazo2, *pMazoFinal;

    printf("Ingrese la cantidad de cartas ha mezclar: ");
    scanf("%d", &Tama);

    MitadTama=Tama/2;

    pMazoIni = malloc(Tama*sizeof(int));
    pMazo1 = malloc(MitadTama*sizeof(int));
    pMazo2 = malloc(MitadTama*sizeof(int));
    pMazoFinal = malloc(Tama*sizeof(int));

    leerMazoEntero(pMazoIni, Tama);
    dividirMazo(pMazoIni, pMazo1, pMazo2, MitadTama, Tama);
    mezclarMazo(pMazoFinal, pMazo1, pMazo2, MitadTama);
    mostrarMazo(pMazoFinal, Tama);

    free(pMazoIni);
    free(pMazo1);
    free(pMazo2);
    free(pMazoFinal);

    return 0;
}

void leerMazoEntero(int *pMazoIni, int Tama)
{
    //carga el mazo de cartas
    for(int i=0; i<Tama; i++){
        *pMazoIni = i+1;
        printf("%d \t", *pMazoIni);
        pMazoIni++;
    }
}
```

```
void dividirMazo(int *pMazoIni,int *pMazo1,int *pMazo2,int MitadTama, int Tama)
{
    //El mazo original se divide en dos partes
    for(int i=0; i<MitadTama; i++){
        *pMazo1 = *pMazoIni;
        pMazo1++;
        pMazoIni++;
    }
    for(int i= MitadTama; i<Tama; i++){
        *pMazo2 = *pMazoIni;
        pMazo2++;
        pMazoIni++;
    }
}

void mezclarMazo(int *pMazoFinal,int *pMazo1,int *pMazo2,int MitadTama)
{
    for(int i=0;i<MitadTama;i++){

        *pMazoFinal= *pMazo1;
        pMazoFinal++;
        *pMazoFinal= *pMazo2;

        pMazo1++;
        pMazo2++;
        pMazoFinal++;
    }
}

void mostrarMazo(int *pMazoFinal, int Tama)
{
    //Muestro el mazo final mezclado
    printf("\n Mazo Final \n");
    for(int i=0; i<Tama; i++){
        printf("%d \t", *pMazoFinal);
        pMazoFinal++;
    }
}
```

3. Diseñe un algoritmo que le permita obtener las siguientes medidas estadísticas en un arreglo numérico de una dimensión: El valor máximo, el valor mínimo, el promedio de los elementos del arreglo y la cantidad de elementos que superan el promedio. Implemente en C.

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int *pNumeros;
    int n, mayor = 0, menor = 9999, suma = 0, supera_promedio=0;
    float promedio;

    printf("Cantidad de numeros a ingresar: \n ");
    scanf("%d", &n);

    pNumeros = malloc(n * sizeof(int));

    //carga el arreglo de numeros (puntero indexado)
    for (int i=0; i<n; i++){
        printf("\nIngrese numeros[%d] = ", i);
        scanf("%d", &pNumeros[i]);
    }

    //analizo los datos
    for(int i=0; i<n; i++){
        if(pNumeros[i] > mayor){
            mayor = pNumeros[i];
        }
        if(pNumeros[i] < menor){
            menor = pNumeros[i];
        }
        suma = suma + pNumeros[i];
    }
    promedio = suma/n;

    for(int i=0; i<n; i++){
        if(pNumeros[i] > promedio){
            supera_promedio++;
            printf("Entra en %d: %d\n", i, pNumeros[i]);
        }
    }

    printf("\nEl MAYOR de los elementos es: %d", mayor);
    printf("\nEl MENOR de los elementos es: %d", menor);
    printf("\nEl PROMEDIO de los elementos es: %.2f", promedio);
    printf("\nCantidad de elementos que superan el PROMEDIO son: %d", supera_promedio);

    free(pNumeros);
    return 0;
}
```