

Pila:

```
int main() {
    srand(time(NULL));
    struct pila pilaMain;
    int n;
    printf("Numero de operaciones por pila: ");
    scanf("%d", &n);
    inicializacion(&pilaMain);
    for (int i = 0; i < n; i++) {
        int n = rand() % 2;
        int num = rand() % 10;
        if(n == 0) {
            printf("Operacion Pop. \n");
            pop(&pilaMain);
        } else if(n == 1) {
            push(&pilaMain, num);
            printf("Operacion Push. \n");
        }
        mostrarElementos(&pilaMain);
    }
    printf("Estado final de la pila: \n");
    mostrarElementos(&pilaMain);
}
```

Después de crear la pila como se introdujo en clase, en la función main se utiliza un randomizador de numeros para poder decidir entre hacer un push y un pop a la pila. Tambien se pone un numero aleatorio en el caso de ser un push. AL final del numero de operaciones que proporciona el usuario, se muestra el estado de la pila final:

```
Numero de operaciones por pila: 8
Operacion Pop.
La pila esta vacia.
Elementos:

Operacion Pop.
La pila esta vacia.
Elementos:

Nuevo elemento en la pila: 9
Operacion Push.
Elementos:
9
Operacion Pop.
Numero eliminado: 9
Elementos:

Nuevo elemento en la pila: 6
Operacion Push.
Elementos:
6
Nuevo elemento en la pila: 9
Operacion Push.
Elementos:
9 6
Nuevo elemento en la pila: 4
Operacion Push.
Elementos:
4 9 6
Operacion Pop.
Numero eliminado: 4
Elementos:
9 6
Estado final de la pila:
Elementos:
9 6
```

Cola:

```
int main() {
    srand(time(NULL));
    int num = 0;
    struct cola c;
    inicializar(&c);
    for (int i = 0; i < MAX; i++) {
        num = (rand() % 5) + 1;
        queue(&c, num);
    }
    printf("Elementos de la cola: \n");
    mostrarElementos(&c);
    cuadrar(&c);
    printf("Elementos cuadrados de la cola: \n");
    mostrarElementos(&c);
}
```

En el main de esta, simplemente se llena la cola con numeros aleatorios, creados similarmente a los que se usan en la parte de la pila; y después se define una función cuadrar:

```
void cuadrar(struct cola *c) {
    for (int i = 0; i <= MAX; i++) {
        c->valores[i] = c->valores[i] * c->valores[i];
    }
}
```

Esta funcion toma cada valor dentro de la cola, y los cuadra. Entonces se muestra el estado final de la cola y eso es todo:

```
Nuevo elemento en la cola: 1
Nuevo elemento en la cola: 4
Nuevo elemento en la cola: 5
Nuevo elemento en la cola: 4
Nuevo elemento en la cola: 5
Nuevo elemento en la cola: 2
Elementos de la cola:
Elemento 0: 4
Elemento 1: 5
Elemento 2: 4
Elemento 3: 5
Elemento 4: 2
Elemento 5: 2
Elementos cuadrados de la cola:
Elemento 0: 16
Elemento 1: 25
Elemento 2: 16
Elemento 3: 25
Elemento 4: 4
Elemento 5: 4
```