|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Clase | Método | Escenario | Valores de entrada | Resultado |
| CQueue | enqueue() | escenario1(): Se inicializa un objeto de tipo jugador, una nueva cola y se procede a agrega el nuevo jugador a la estrucutra | Queue<Jugador> miCola  Jugador shaggy | Verdadero. El tamaño de la cola debe ser igual a 1. |
| CQueue | isEmpty() | escenario2(): Se inicializa una cola vacía. | Queue<Jugador> miCola | Verdadero. El método isEmpty() devuelve true dado que la cola esta vacía. |
| CQueue | dequeue() | escenario3():Se inicializa un objeto de tipo jugador, una nueva cola y se procede a agregar dos nuevos jugadores a ella, se guarda su tamaño, luego se ejecuta el método en cuestión y se vuelve a guardar su tamaño. | Queue<Jugador> miCola  Jugador shaggy Jugador rockellar | Verdadero. El número de elementos de la cola antes y después de ejecutar el método en cuestión difiere en una unidad |
| CLIst | remove() | escenario1(): Se inicializa una lista agregando un nuevo jugador a ella, se guarda el tamaño de la estructura y luego se procede a agregar otro jugador para después guardar el tamaño en otra variable. | CList <Jugador> myList  Jugador shaggy Jugador rockellar | Verdadero. El tamaño de la lista ha disminuido en una unidad |
| CList | add() | escenario1(): Se inicializa una lista agregando un nuevo jugador a ella, se guarda el tamaño de la estructura y luego se procede a agregar otro jugador para después guardar su en otra variable. | CList<Jugador> myList  Jugador shaggy Jugador rockellar | Verdadero. EL tamaño de la lista se ha incrementado en una unidad. |
| CLIst | isEmpty() | escenario2(): Se crea una lista vacía. | CList <Jugador> myList | Verdadero. El método en cuestión retorna true debido a que el tamaño de la lista es cero. |
| CStack | push() | escenario1(): Se inicializa una pila agregando un nuevo jugador a ella, se guarda el tamaño de la estructura y luego se procede a agregar otro jugador para después guardar el tamaño en otra variable. | CStack<Jugador> miPila  Jugador shaggy Jugador rockellar | Verdadero. EL tamaño de la lista se ha incrementado en una unidad. |
| CStack | isEmpty() | escenario2(): Se inicializa una cola vacía. | CStack<Jugador> miPila | Verdadero. El método isEmpty() devuelve true dado que la pila está vacía. |
| CHashTable | set() | escenario1(): Se inicializa una tabla Hash con dos elementos de tipo jugador. | CHashTable<Integer, String> myHashTable  Jugador shaggy Jugador rockellar | Verdadero. El número de elementos de la tabla ha aumentado en una unidad. |
| CHashTable | get() | escenario1(): Se inicializa una tabla Hash con dos elementos de tipo jugador. | CHashTable<Integer, String> myHashTable  Jugador shaggy Jugador rockellar | Verdadero. La clave ingresada corresponde al objeto devuelto por el método en cuestión. |
| CHashTable | remove() | escenario1(): Se inicializa una tabla Hash con dos elementos de tipo jugador. | CHashTable<Integer, String> myHashTable  Jugador shaggy Jugador rockellar | Verdadero. El número de elementos de la tabla ha aumentado en una unidad. |
| CHashTable | isEmpty() | escenario2(): Se inicializa una tabla Hash vacía. | CHashTable<Integer, String> myHashTable | Verdadero. El método en cuestión retorna true debido a que la tabla se encuentra. |
| SCHeap | add() | escenario1(): Se inicializa una tabla Hash con dos elementos de tipo jugador. | SCHeap<Jugador> myHeap  Jugador shaggy Jugador rockellar | Verdadero. El length del heap ha aumentado en una unidad. |
| SCHeap | heapSort() | escenario2(): Se inicializa una tabla Hash con tres elementos de tipo jugador. | SCHeap<Jugador> myHeap  Jugador shaggy Jugador rockellar  Jugador Carlos | Verdadero. Los jugadores se organizaron de manera descendente |
| SCHeap | remove() | escenario1(): Se inicializa una tabla Hash con dos elementos de tipo jugador. | SCHeap<Jugador> myHeap  Jugador shaggy Jugador rockellar | Verdadero. El length del heap ha aumentado en una unidad. |