---

### 1. Which of the following is not a collection in C#?

- A) List

- B) Dictionary

- C) Stack

- D) String

\*\*Answer\*\*: D) String

---

### 2. Which collection type allows for key-value pairs?

- A) List

- B) Dictionary

- C) Queue

- D) Stack

\*\*Answer\*\*: B) Dictionary

---

### 3. Which of the following is not a generic collection?

- A) ArrayList

- B) List<T>

- C) Dictionary<TKey, TValue>

- D) Queue<T>

\*\*Answer\*\*: A) ArrayList

---

### 4. What is the output of the following code?

```csharp

List<int> numbers = new List<int> { 1, 2, 3, 4 };

Console.WriteLine(numbers[2]);

```

- A) 1

- B) 2

- C) 3

- D) 4

\*\*Answer\*\*: C) 3

---

### 5. Which interface must a collection implement to allow enumeration?

- A) IComparable

- B) IEnumerable

- C) ICollection

- D) IList

\*\*Answer\*\*: B) IEnumerable

---

### 6. Which of the following is not a feature of generics in C#?

- A) Type safety

- B) Reduced boxing/unboxing

- C) Late binding

- D) Code reusability

\*\*Answer\*\*: C) Late binding

---

### 7. What is the purpose of the `T` in `List<T>`?

- A) It is a predefined type

- B) It is a generic type parameter

- C) It stands for a template

- D) It is a keyword for integers

\*\*Answer\*\*: B) It is a generic type parameter

---

### 8. Which collection can be used to store a unique set of elements?

- A) List<T>

- B) HashSet<T>

- C) Dictionary<TKey, TValue>

- D) Stack<T>

\*\*Answer\*\*: B) HashSet<T>

---

### 9. Which of the following is true about the `Queue<T>` collection?

- A) It operates on a LIFO basis

- B) It operates on a FIFO basis

- C) It allows key-value pairs

- D) It supports indexed access

\*\*Answer\*\*: B) It operates on a FIFO basis

---

### 10. How do you add an item to a `Dictionary<TKey, TValue>` in C#?

- A) `dictionary.Insert(key, value)`

- B) `dictionary.Add(key, value)`

- C) `dictionary.Push(key, value)`

- D) `dictionary.Append(key, value)`

\*\*Answer\*\*: B) `dictionary.Add(key, value)`

---

### 11. What does the `Count` property in a collection return?

- A) The maximum capacity of the collection

- B) The number of items in the collection

- C) The index of the last element

- D) The memory used by the collection

\*\*Answer\*\*: B) The number of items in the collection

---

### 12. What is the primary advantage of using generic collections over non-generic collections?

- A) Improved performance

- B) Type safety at compile time

- C) Better user interface

- D) Faster memory allocation

\*\*Answer\*\*: B) Type safety at compile time

---

### 13. Which of the following statements is true about `Stack<T>`?

- A) It allows random access

- B) It operates on a FIFO principle

- C) It is similar to a Queue<T>

- D) It operates on a LIFO principle

\*\*Answer\*\*: D) It operates on a LIFO principle

---

### 14. Which method is used to remove an element from a `Queue<T>` in C#?

- A) `Dequeue()`

- B) `Pop()`

- C) `Remove()`

- D) `Peek()`

\*\*Answer\*\*: A) `Dequeue()`

---

### 15. What happens when you call `stack.Peek()` on a `Stack<T>`?

- A) Removes and returns the top element

- B) Returns the top element without removing it

- C) Returns the number of elements in the stack

- D) Returns null

\*\*Answer\*\*: B) Returns the top element without removing it

---

### 16. Which of the following can hold duplicate keys?

- A) Dictionary<TKey, TValue>

- B) SortedList<TKey, TValue>

- C) HashSet<T>

- D) None of the above

\*\*Answer\*\*: D) None of the above

---

### 17. Which of these collections is indexed?

- A) Stack<T>

- B) Queue<T>

- C) Dictionary<TKey, TValue>

- D) List<T>

\*\*Answer\*\*: D) List<T>

---

### 18. What method is used to sort a `List<T>` in C#?

- A) `list.OrderBy()`

- B) `list.Sort()`

- C) `list.Arrange()`

- D) `list.Order()`

\*\*Answer\*\*: B) `list.Sort()`

---

### 19. What is the difference between `ArrayList` and `List<T>`?

- A) `ArrayList` is non-generic, `List<T>` is generic

- B) `ArrayList` is faster

- C) `List<T>` does not allow indexing

- D) There is no difference

\*\*Answer\*\*: A) `ArrayList` is non-generic, `List<T>` is generic

---

### 20. What is the default capacity of a `List<T>` when it is initialized?

- A) 0

- B) 10

- C) 4

- D) 1

\*\*Answer\*\*: C) 4

---

### 21. What will happen if you add a duplicate key to a `Dictionary<TKey, TValue>`?

- A) It will overwrite the existing key

- B) It will throw an exception

- C) It will add the new key

- D) It will remove the previous key

\*\*Answer\*\*: B) It will throw an exception

---

### 22. Which of the following is an example of a non-generic collection in C#?

- A) List<T>

- B) Dictionary<TKey, TValue>

- C) Stack

- D) Queue<T>

\*\*Answer\*\*: C) Stack

---

### 23. What method is used to remove an element from a `List<T>` by index?

- A) `Delete()`

- B) `RemoveAt()`

- C) `RemoveByIndex()`

- D) `DeleteAt()`

\*\*Answer\*\*: B) `RemoveAt()`

---

### 24. How do you access an item in a `Dictionary<TKey, TValue>` by key?

- A) `dictionary.GetValue(key)`

- B) `dictionary[key]`

- C) `dictionary.Find(key)`

- D) `dictionary.Lookup(key)`

\*\*Answer\*\*: B) `dictionary[key]`

---

### 25. Which method checks if a `Dictionary<TKey, TValue>` contains a specific key?

- A) `ContainsKey()`

- B) `KeyExists()`

- C) `HasKey()`

- D) `FindKey()`

\*\*Answer\*\*: A) `ContainsKey()`

---

### 26. What does the `Where` method in LINQ do when applied to a collection?

- A) Filters elements based on a condition

- B) Sorts the collection

- C) Modifies elements in the collection

- D) Converts the collection to another type

\*\*Answer\*\*: A) Filters elements based on a condition

---

### 27. Which of the following collection types maintains elements in sorted order based on the keys?

- A) HashSet<T>

- B) SortedList<TKey, TValue>

- C) List<T>

- D) Queue<T>

\*\*Answer\*\*: B) SortedList<TKey, TValue>

---

### 28. What will `stack.Pop()` do on an empty `Stack<T>`?

- A) Return null

- B) Throw an `InvalidOperationException`

- C) Return a default value

- D) Do nothing

\*\*Answer\*\*: B) Throw an `InvalidOperationException`

---

### 29. Which of the following operations is not allowed on a `HashSet<T>`?

- A) Adding duplicate elements

- B) Removing an element

- C) Checking if an element exists

- D) Enumerating through elements

\*\*Answer\*\*: A) Adding duplicate elements

---

### 30. What is the output of the following code?

```csharp

List<string> fruits = new List<string> { "apple", "banana", "cherry" };

Console.WriteLine(fruits.Contains("banana"));

```

- A) True

- B) False

- C) InvalidOperationException

- D) Syntax error

\*\*Answer\*\*: A) True

---

### 31. Which collection class does not allow duplicate values in C#?

- A) List<T>

- B) Dictionary<TKey, TValue>

- C) HashSet<T>

- D) Queue<T>

\*\*Answer\*\*: C) HashSet<T>

---

### 32. Which method is used to check if a `List<T>` contains a specific element?

- A) `list.HasElement()`

- B) `list.Contains()`

- C) `list.Exists()`

- D) `list.CheckElement()

`

\*\*Answer\*\*: B) `list.Contains()`

---

### 33. Which of the following is true for `Dictionary<TKey, TValue>`?

- A) It is a FIFO collection

- B) Keys can be duplicate

- C) It stores elements in key-value pairs

- D) It is a stack-based collection

\*\*Answer\*\*: C) It stores elements in key-value pairs

---

### 34. What method is used to remove all elements from a `List<T>`?

- A) `Clear()`

- B) `Empty()`

- C) `RemoveAll()`

- D) `DeleteAll()`

\*\*Answer\*\*: A) `Clear()`

---

### 35. Which of the following is the most efficient collection for quick lookups by key?

- A) List<T>

- B) Dictionary<TKey, TValue>

- C) Stack<T>

- D) Queue<T>

\*\*Answer\*\*: B) Dictionary<TKey, TValue>

---

### 36. What will `queue.Peek()` do in a `Queue<T>`?

- A) Remove and return the front element

- B) Return the front element without removing it

- C) Return the count of elements

- D) Return the last element

\*\*Answer\*\*: B) Return the front element without removing it

---

### 37. What is the default initial size of a `Queue<T>` when initialized?

- A) 4

- B) 0

- C) 10

- D) 16

\*\*Answer\*\*: B) 0

---

### 38. Which collection automatically increases its size when required?

- A) Stack<T>

- B) ArrayList

- C) Dictionary<TKey, TValue>

- D) List<T>

\*\*Answer\*\*: D) List<T>

---

### 39. How is a new element added to the top of a `Stack<T>`?

- A) `Push()`

- B) `Insert()`

- C) `Add()`

- D) `Enqueue()`

\*\*Answer\*\*: A) `Push()`

---

### 40. Which interface ensures that a collection can be enumerated?

- A) IEnumerable

- B) ICollection

- C) IList

- D) IEnum

\*\*Answer\*\*: A) IEnumerable

---

### 41. What type of collection is used for LIFO (Last In First Out) operations?

- A) Queue<T>

- B) Stack<T>

- C) Dictionary<TKey, TValue>

- D) List<T>

\*\*Answer\*\*: B) Stack<T>

---

### 42. Which class is a non-generic version of `List<T>`?

- A) Stack

- B) ArrayList

- C) Queue

- D) HashSet

\*\*Answer\*\*: B) ArrayList

---

### 43. How do you retrieve the number of elements in a `Dictionary<TKey, TValue>`?

- A) `dictionary.Size`

- B) `dictionary.Length`

- C) `dictionary.Count`

- D) `dictionary.Elements`

\*\*Answer\*\*: C) `dictionary.Count`

---

### 44. Which of the following collection types is best for FIFO (First In First Out) operations?

- A) List<T>

- B) Queue<T>

- C) Stack<T>

- D) Dictionary<TKey, TValue>

\*\*Answer\*\*: B) Queue<T>

---

### 45. What is the primary feature of a `HashSet<T>`?

- A) It preserves the order of elements

- B) It allows duplicate elements

- C) It provides fast lookups and does not allow duplicates

- D) It is used for LIFO operations

\*\*Answer\*\*: C) It provides fast lookups and does not allow duplicates

---

### 46. What will happen if you try to remove an item from a `Queue<T>` when it is empty?

- A) Return null

- B) Throw `InvalidOperationException`

- C) Return a default value

- D) Do nothing

\*\*Answer\*\*: B) Throw `InvalidOperationException`

---

### 47. Which collection does not allow access by index?

- A) List<T>

- B) ArrayList

- C) Queue<T>

- D) Dictionary<TKey, TValue>

\*\*Answer\*\*: C) Queue<T>

---

### 48. What type of collection is a `KeyValuePair<TKey, TValue>` used in?

- A) List<T>

- B) Dictionary<TKey, TValue>

- C) Stack<T>

- D) Queue<T>

\*\*Answer\*\*: B) Dictionary<TKey, TValue>

---

### 49. How do you remove an element from a `HashSet<T>`?

- A) `hashSet.Delete()`

- B) `hashSet.Remove()`

- C) `hashSet.Clear()`

- D) `hashSet.Pop()`

\*\*Answer\*\*: B) `hashSet.Remove()`

---

### 50. Which method is used to add an element to a `List<T>`?

- A) `list.Add()`

- B) `list.Insert()`

- C) `list.Append()`

- D) `list.Push()`

\*\*Answer\*\*: A) `list.Add()`

---