

LAB 5: Working with collections

Task 1: Examining abstract data types

Examining an array list:

Question: What do you think is the purpose of each of the three instance variables of `SimpleArrayList`?

Array to hold items, maximum number of items, current number of items stored

Question: How does a `SimpleArrayList` store objects?

In an array

Question: What kind of objects can it store?

Any type which is derived from `System.Object`, i.e. any type

Question: How many objects can it store?

10

Question: How many objects does it currently contain?

6

Examining a linked list:

Question: How does a `SimpleLinkedList` store objects?

In nodes, each of which contains an item and a reference to the next node

Question: How many objects can it store?

No fixed limit, can keep adding nodes

Question: How many objects does it currently contain?

6

Examining a dictionary:

Question: How does a Dictionary store objects?

An array of keys and an array of values

Question: What does 'keys' contain?

The ID values for the objects

Question: What do you notice about the order in which keys and objects are stored?

Order of keys matches order of values, but not stored in order of adding, and array has gaps – storage position depends on hash code calculated from key

Task 2: Using .NET Framework collections

Questions:

- *How does the List class store the objects it holds?*

In an array

- *What is the advantage of using a generic collection such as List<Patient>?*

Don't need to cast to get object out as instance of Patient

- *What types of object can be stored in a list of type List<Patient>?*

Patient and any type derived from Patient

Use Dictionary instead of List:

Questions:

- *How does the Dictionary class store the objects it holds?*

Array of keys and array of values

- *Which version of HealthCentre will be more efficient when you need to find a patient if there is a very large number of patients? Why?*

Dictionary – fast lookup by key rather than searching through list