# Knowledge-Based Assessment (KBA) Part 1

**Assessment Resources:**

|  |
| --- |
| Marking key available for lecturer via Blackboard.  Students may refer to the lecture material in formulating their answers. |

**Assessment Instructions:**

|  |
| --- |
| Students must attempt all questions. Answer succinctly using full sentences. At most two paragraphs are expected per answer.  All answers must be at the student’s own words – copying generated code or answers from ChatGPT or other AI tools is **strictly** prohibited. |

# Binary file handling techniques

Consider the following fictional image file format:

**File Structure**: A binary file containing image data.

**First 9 bytes**: Text encoded with the dimensions of the image (e.g., "1920x1080").

**Following bytes**: Series of 3 bytes representing RGB values of each of the pixels.

E,g. 1920X1080200244002

1. Given the specification, outline how you would use seek, read, write, and tell (as appropriate) to:
2. Retrieve the RGB Value of the central pixel using random access techniques
3. Retrieve all pixel values as integers from a random 100x100 grid within the image.
4. Reduce the resolution in half.

A seek to the point to work this out, find how many pixels, each pixel has three colours, each colour has three bits than off set by nine bits for the encoding prefix (1920 \* 1080 / 2 ) \* 3 \* 3 + 9. Then return the next nine bits read(9).

B Go to the starting pixel get the next one hundred pixels, to get the next line add 1920 – 100 pixels. Repeat until get the desired dimensions, in this case ninety eight times.

C Take every second pixel.

# Data Structures Fundamentals

Arrays

1. What's the difference between an array and a list in Python? How would you create an array in Python to store numerical data, and why might you choose an array over a list?

List are build in. Collection of item of any type, they can have any type because there are an index of memory address. Is an abstraction.

Arrays need a module, a collection of homogeneous type, identified by index or key. Is a data structure.

From array import array

so\_primitive = array(‘format’,[‘a’,’r’,’r’,’a’,’y’])

If you know you are working with data of all the same type, especially if they are large data sets.

Linked Lists

1. Explain how a linked list differs from a standard list or array. Are there any performance benefits to a linked list compared to an array? Why?

A link list is made up of nodes, each node has a reference to the next. Performance depends on what you are doing, if you are adding to the beginning of a list you just add a node with a reference, an array you need to shuffle the whole thing.

Stack

1. What is the difference between a stack and a queue? Outline how you would define a stack (append and pop) using a Python list. Are there any disadvantages to using a Python list as a queue or a stack?

A stack is last in first out where a queue is first in first out. If you need to add something to the stack use append() and remove and return use pop() but you need to be careful how the list is constructed, it’s mutability can lead to unexpected things happening. Because they are not made for this there Is no error handling, need to add checks. Misuse of list features when using them as a stack, which conflicts with LIFO of a stack. Less readability using dedicated stack data structure or class could improve readability.

Trees and Graphs

1. Write an example application for a Tree data structure and one for a graph. Why would you use a Tree over a Graph?

File system is an example of a tree data structure and data bases can use graphs as their data structure. Trees are hierarchical and graphs are relational so you would choose a tree if that is the most appropriate data structure.

Hash Tables

1. What built in collection in Python uses a hash table? Describe how one might store data in an array using a key and look up that item with the same key. Assume that the key is a string of characters such as a surname.

Dictionary uses a hash table. The key is hashed and then the hash corresponds with a slot in the hash table if there is a hash collision, maybe they have the same inputs such as a surname, it would be put into the next available slot in the hash table. where the data or reference to the data is.

Sets

1. What are some key use cases for sets in Python? Describe at a high-level how Python maintains the uniqueness of items in a set.

[https://docs.python.org/2/library/stdtypes.html#set](https://docs.python.org/2/library/stdtypes.html" \l "set)

“Common uses include membership testing, removing duplicates from a sequence, and computing standard math operations on sets such as intersection, union, difference, and symmetric difference”

.

Custom data structures

1. Share a real-world scenario, from your personal experience in the workplace or at home, where a user-defined data structure might be applicable. Explain the benefits and potential drawbacks of designing a custom data structure for this scenario.  
     
   **Example answer**: *When I organise my course development work, it sometimes makes sense for me to work on the same type of item (e.g. assessments for different unit of competencies) and sometimes on different types of items for the same topic (e.g. assessments and lesson plans for ICTPRG443). So if I wanted to create an application to organise my work, I would need to create a custom data structure that allows me to have both topics and task types as keys. The benefit of this data structure is that it closely aligns with the job requirements. The disadvantages are that I must build integrity checks and other constraints that don't directly map to my original needs. Using an SQL database may provide me with similar benefits without the need to create a bespoke data structure.*

Binary search

1. In a number guessing games, a player is asked to guess a number in a given range. When the player makes a guess, they are given a clue: is their guess higher, lower, or the same as the target numbers? Using the clues players adapt their answer. What is the optimal strategy for guessing the target number in the fewest guesses? How is this an example of a binary search? Why do we say that in a binary search the list we are searching in must be sorted?

Have a low, high and a middle. If the target is lower than the middle move the high to the middle minus one. If the target is higher than the middle move the low to the middle plus one. Find the middle between low and high and repeat. This is a binary search because there are two options in each decision, find no/yes, middle lower/higher. It needs to be sorted from low to high or high to low because if it is random comparing lower or higher would not provide any meaning.