



OCR A Level Computer Science



Your notes

6.2 Thinking Ahead

Contents

- * Computational Thinking: Thinking Ahead



Your notes

Computational Thinking: Thinking Ahead

Inputs & Outputs in Computational Thinking

- In order to create programs that are efficient and robust, the program must be decomposed into its component parts
- Developers need to consider an algorithm's inputs and outputs and define each of them before implementing the processes that turn inputs into outputs
- Determining the inputs and outputs factor into an algorithm's preconditions for operation, that is the conditions necessary for the algorithm to successfully complete

How do you Identify Inputs and Outputs?

- Most algorithms require inputs **in order** to produce outputs
- An input can be any piece of data that an algorithm requires in order to run, usually passed as a parameter to a subroutine
- An output can be any piece of data returned to the calling subroutine, or data provided to an output device such as a monitor or speaker
- When designing an algorithm, the inputs and outputs must be explicitly defined, including their type, size and format
 - If not defined explicitly, creating an algorithm becomes much more difficult and can cause errors and problems later on when unexpected events occur
- Consider an algorithm such as a search algorithm, the inputs and outputs must be defined in advance before writing the program:
 - Inputs: What are the data types? How many elements are in the array? Is the array sorted?
 - Outputs: Is a message returned? Is the value returned? Is the index of the element in the array returned? What is the data type? Is a boolean value True or False returned? Is nothing returned?
- To define the inputs and outputs of a problem, the following process can be used:
 - Name: LinearSearch
 - Input: A list of integers $my_list = (i_1, i_2, i_3, \dots, i_n)$
 - Output: The index of the element as an integer

Preconditions in Computational Thinking



Your notes

Determining preconditions

- Preconditions are conditions that must be true for an algorithm to complete successfully, without errors or crashing
- An example of a precondition is that the binary search algorithm must be supplied with an ordered list
 - If an unordered list is passed to the subroutine, it will fail to execute correctly
- Other preconditions could include:
 - The parameter list must not be empty, otherwise index out of bounds errors will occur
 - The data may need to be all the same data type, otherwise invalid results may be returned or invalid comparisons made
 - The target data (the element being searched for) must be the same data type as the list and also cannot be empty
- A bubble sort, being a $O(n^2)$ algorithm may require a limit on the size of the input list to prevent the program taking a very long time to complete
- Preconditions, once determined, can be programmed into the subroutine or can be supplied in documentation with the subroutine to ensure the program successfully completes
- To define the preconditions of a problem, the following process can be used:
 - Name: BinarySearch
 - Inputs: A list of integers my_list ($i_1, i_2, i_3, i_4, \dots, i_n$)
 - Outputs: The index of the element as an integer
 - Preconditions: length of my_list > 0, type(my_list) = int
- The advantages of specifying preconditions are:
 - If the preconditions are provided in documentation, then the developer is aware of what checks are required before calling the subroutine
 - If no preconditions are specified then the program itself will carry out any validation checks, preventing the developer from writing additional code
 - Defining preconditions, including inputs and outputs, allows the subroutine to be reusable and put into a library and called on at any time



Worked Example



Your notes

Janet is designing a piece of software for a furniture company.

The software will allow a user to plan the position of furniture in a room. Users will be able to set the size and shape of a room, and then choose furniture from a library of furniture items. These pieces of furniture will have set sizes and designs and the user will be able to view the room in 3D to see how it looks from a variety of angles.

The program allows the user to enter dimensions of the room and the furniture. There are preconditions that must be met before the software will draw the room and furniture.

Suggest two preconditions that must be met before the software will run.

2 marks

Answer:

All room dimensions entered must be greater than 0. [1]

Width, length and height of the room must have been entered. [1]

Stating “room dimensions” is not enough, you must be clear and provide a condition that can be evaluated e.g. greater than 0.

Caching in Computational Thinking

The nature and benefits of caching

- Cache is the tiny storage on a processor that is used to temporarily store frequently used data and instructions while a program is running
 - The act of storing data and instructions is called caching
- Caching is usually performed automatically by the operating system, though some languages such as C allow manipulation of the underlying hardware
- Other uses of cache include web caching where HTML pages and images most recently looked at are stored, allowing for faster access and saving bandwidth by not requiring them to be redownloaded each time

Reusable Program Components

The need for reusable program components

- Commonly used subroutines are bundled into packages called libraries in most programming languages
- Examples of such subroutines are printing, casting, finding the maximum and minimum value of a list, generating random numbers, etc

- Developers may also create their own libraries of complex subroutines and use them in their own or other projects
 - Examples could be bespoke subroutines to calculate the length or difference in values of a linked list, or be implementations of data structures such as stacks and queues or standard algorithms such as quick sort or merge sort
- Creating and using subroutines that have already been thoroughly tested and documented will save time when working on new projects



Your notes