AH Computing Science

STANDARD ALGORITHMS – QUICK SORT
Standard algorithms – Quick Sort

Quick sort

- Is a recursive algorithm
- More efficient with large lists than Selection, Insertion and Bubble sort
- Works on the basis that an empty list or a list with only 1 item is by definition a sorted list
- For any other list, it compares every item in it to a pivot value. The pivot value is chosen as the middle of the list, this divides the list into 2 sub-lists. One containing items bigger than the pivot one containing items smaller. Both sub-lists are then quicksorted until the lists are reduced to 1 item. This is a ‘divide and conquer’ solution
Standard algorithms –
Quick Sort

Pseudocode:

This quicksort algorithm is for a list whose index position ranges from i to listlength:

PROCEDURE quickSort(ARRAY OF INTEGER list, INTEGER i, INTEGER listLength)

DECLARE pivotPosition AS INTEGER INITIALLY 0
IF i < listLength THEN
    SET pivotPosition TO partition(list, i, listLength)
    quickSort(list, i, pivotPosition - 1)
    quickSort(list, pivotPosition + 1, listLength)
END IF

END PROCEDURE

FUNCTION partition(ARRAY list, INTEGER i, INTEGER listLength)RETURNS INTEGER
    <swaps items in list to arrange them lower than and higher than the value i>
END FUNCTION

The partition function has not been exemplified here, but it takes three parameters and
swaps items in the list passed to it so that they are separated into ones lower than and
higher than its chosen pivot value. It returns the index value of the pivot.