DSL Practical 5

Mayur Zope Comp A 75

Write a Python program to store second year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using

a) Insertion sort.

b) Shell Sort and display top five scores.

```
def insertion_sort(arr):
    """Sorts an array using insertion sort algorithm."""
    n = len(arr)
    for i in range(1, n):
        key = arr[i]
        j = i - 1
        while j >= 0 and arr[j] > key:
            arr[j + 1] = arr[j]
            j -= 1
            arr[j + 1] = key


def shell_sort(arr):
        """Sorts an array using shell sort algorithm."""
        n = len(arr)
        gap = n // 2
        while gap > 0:
                for i in range(gap, n):
                        temp = arr[i]
                        j = i
                        while j >= gap and arr[j - gap] > temp:
                                arr[j] = arr[j - gap]
                                j -= gap
                        arr[j] = temp
                gap //= 2
```

DSL Practical 5

Mayur Zope Comp A 75

```
def display_top_five_scores(arr):
    """Displays the top five scores from a sorted array."""
    if len(arr) > 5:
        top_five = arr[-5:] # Get the top 5 scores
    else:
        top_five = arr # If less than 5 elements, display all
    print("Top five scores:")
    for score in top_five:
        print(f"{score:.2f}")
    print("\n")

# Example usage
def main():

    # Input: Second year percentages of students
    percentages = [85.5, 78.2, 90.1, 88.7, 76.4, 92.3, 69.5, 81.2, 87.3, 79.8]
    print("Before Sorting : ", percentages,"\n")

    # Make a copy of the list for each sort
    percentages_for_insertion_sort = percentages.copy()
    percentages_for_shell_sort = percentages.copy()

    # Sort using Insertion Sort
    print("Sorting using Insertion Sort...")
    insertion_sort(percentages_for_insertion_sort)
    display_top_five_scores(percentages_for_insertion_sort)



    # Sort using Shell Sort
    print("Sorting using Shell Sort...")
```

```
    shell_sort(percentages_for_shell_sort)
    display_top_five_scores(percentages_for_shell_sort)

if __name__ == "__main__":
    main()
```

// OUTPUT.

```
Before Sorting :  [85.5, 78.2, 90.1, 88.7, 76.4, 92.3, 69.5, 81.2, 87.3, 79.8]

Sorting using Insertion Sort...
Top five scores:
85.50
87.30
88.70
90.10
92.30


Sorting using Shell Sort...
Top five scores:
85.50
87.30
88.70
90.10
92.30



=== Code Execution Successful ===
```