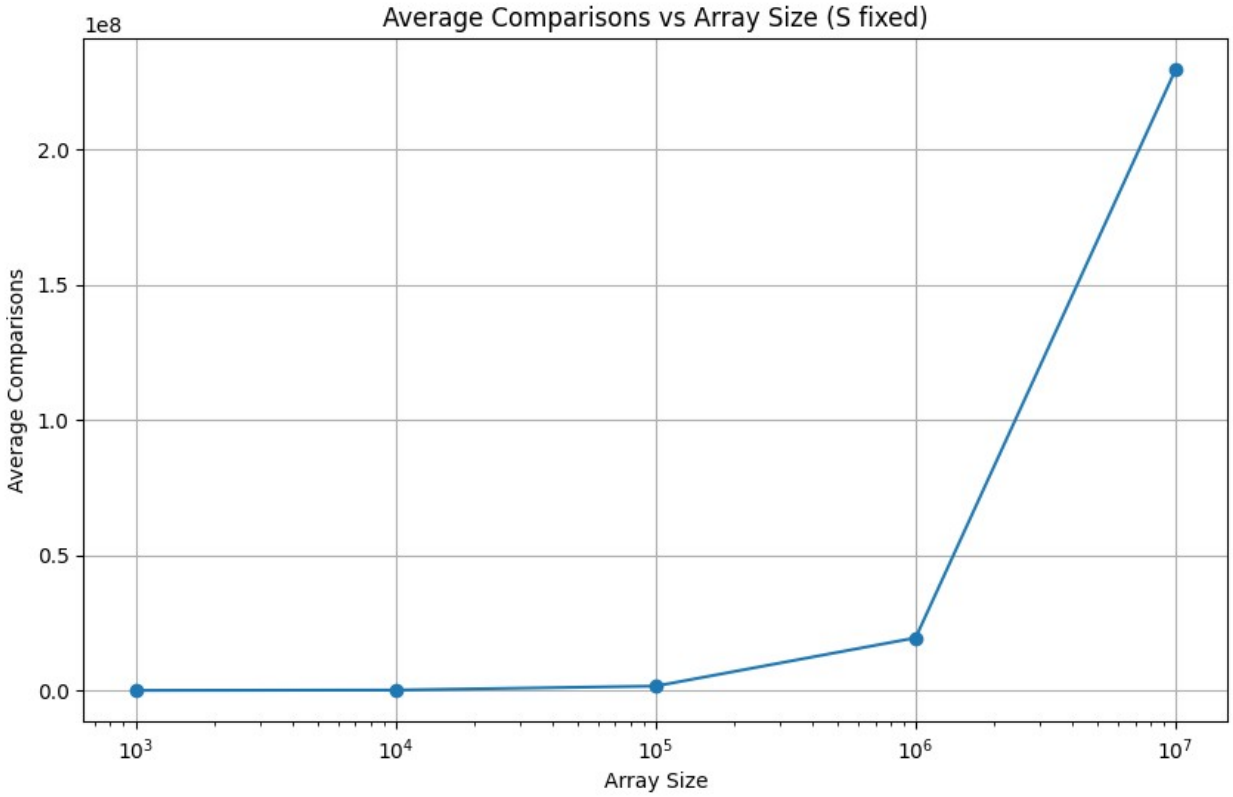
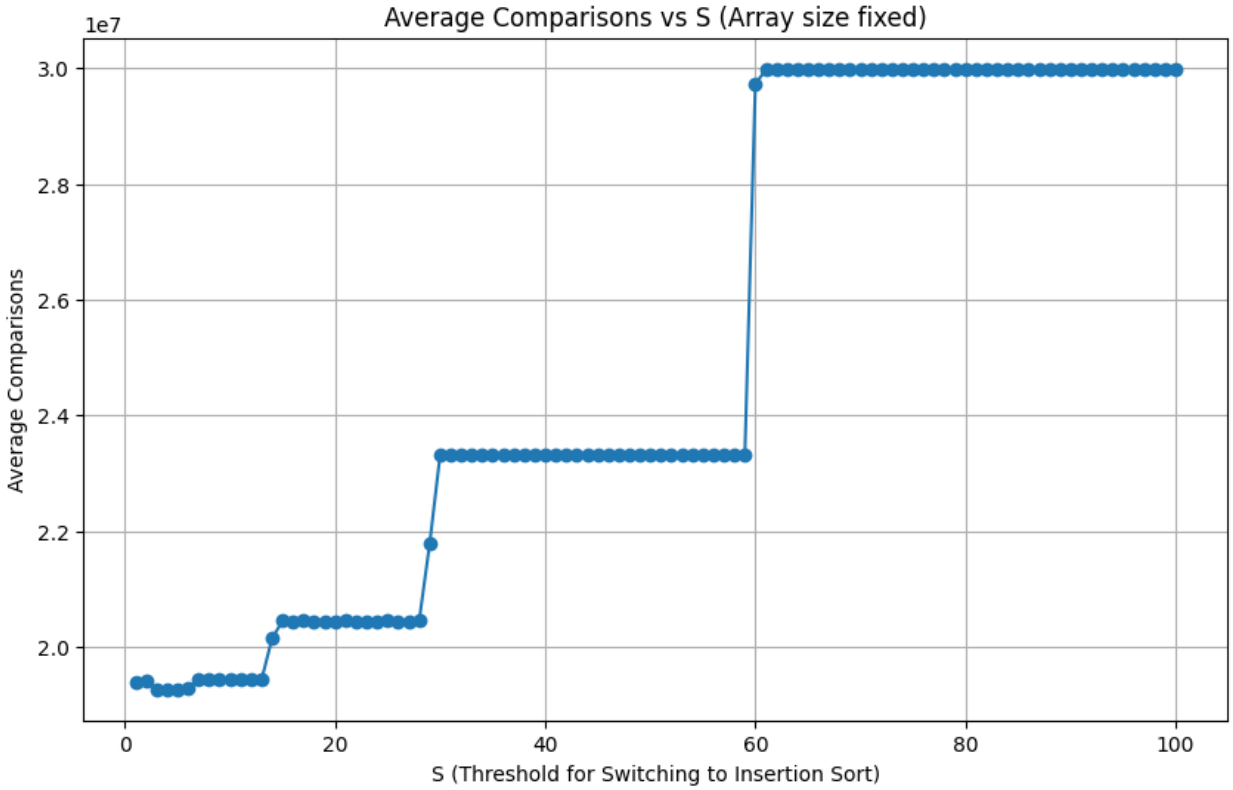
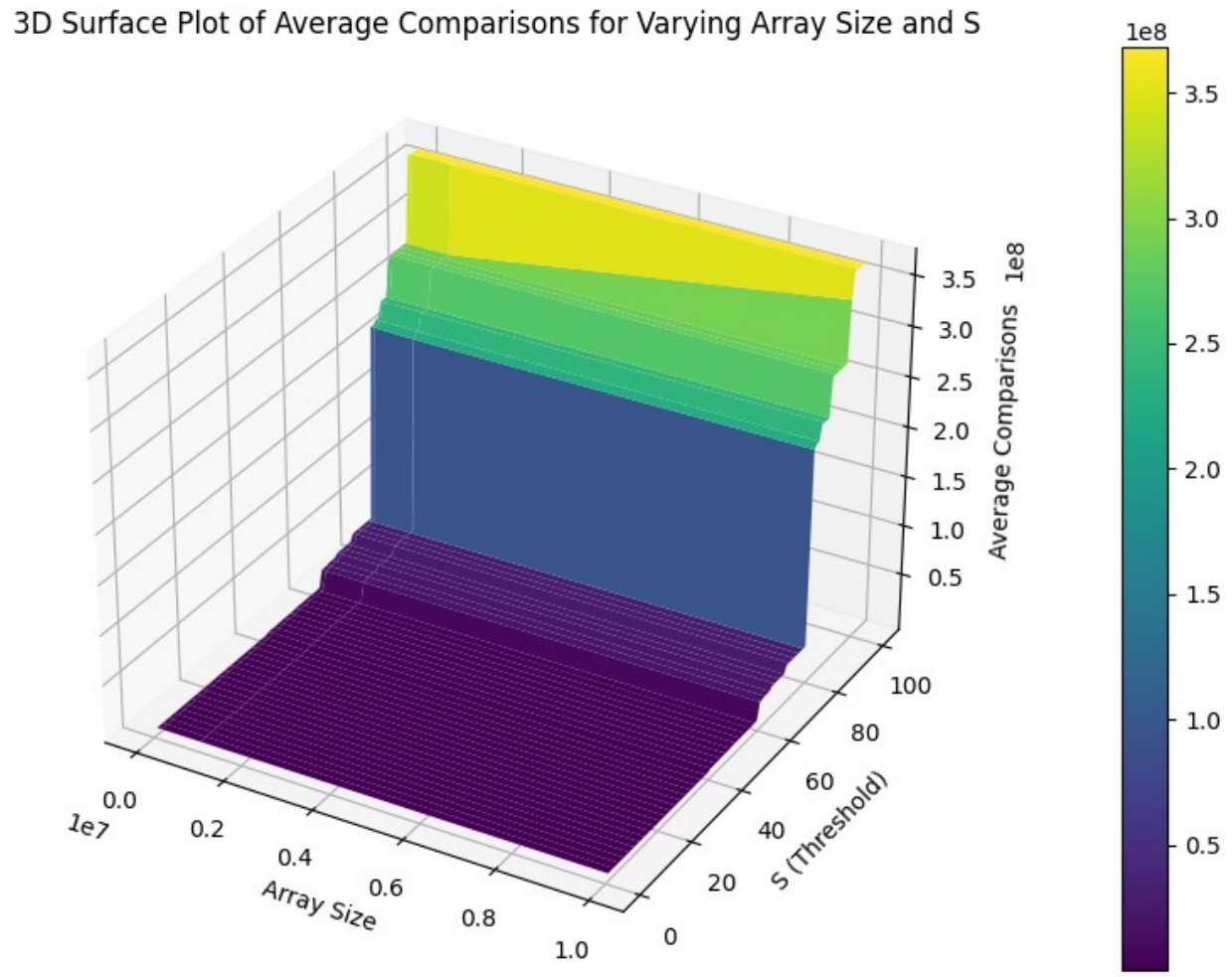
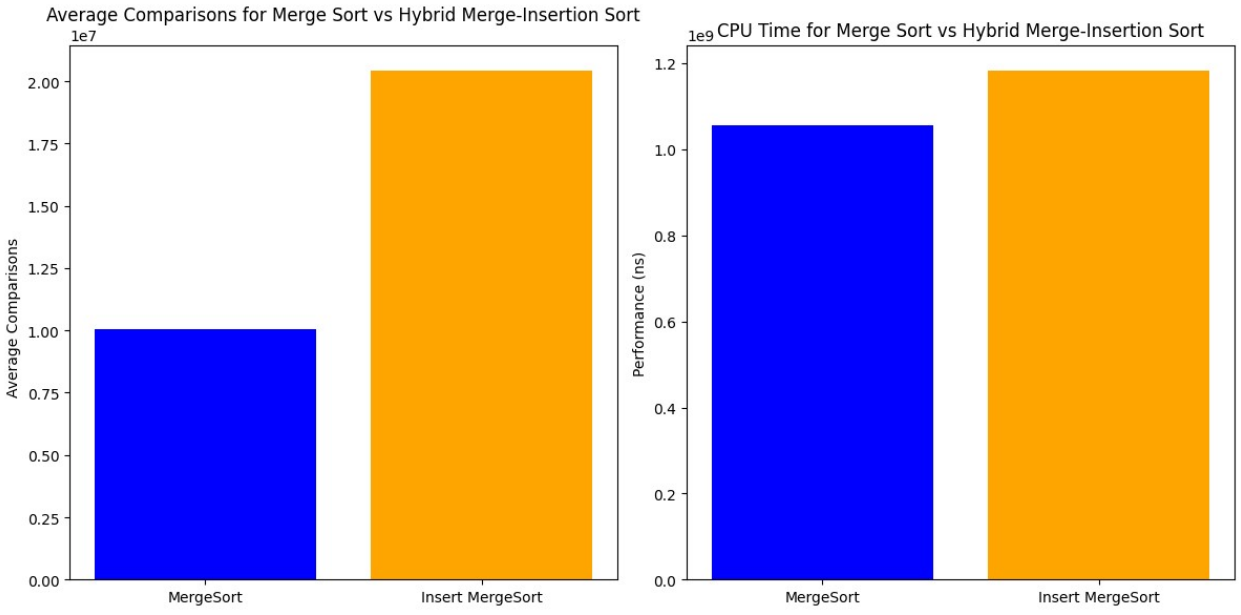
```
# Import necessary libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

# Load the datasets
new_partd_df = pd.read_csv('/work/partd.csv')
new_partci_df = pd.read_csv('/work/partci.csv')
new_partcii_df = pd.read_csv('/work/partcii.csv')
new_partciii_df = pd.read_csv('/work/partciii.csv')

# Task (c) - i: Plot key comparisons vs array size with S fixed (from
partci_df)
plt.figure(figsize=(10, 6))
plt.plot(new_partci_df['ArraySize'], new_partci_df['Average
Comparisons'], marker='o')
plt.title('Average Comparisons vs Array Size (S fixed)')
plt.xlabel('Array Size')
plt.ylabel('Average Comparisons')
plt.xscale('log')  # Log scale for better visualization
plt.grid(True)
plt.show()

# Task (c) - ii: Plot key comparisons vs S with array size fixed (from
partcii_df)
plt.figure(figsize=(10, 6))
plt.plot(new_partcii_df['S'], new_partcii_df['Average Comparisons'],
marker='o')
plt.title('Average Comparisons vs S (Array size fixed)')
plt.xlabel('S (Threshold for Switching to Insertion Sort)')
plt.ylabel('Average Comparisons')
plt.grid(True)
plt.show()

# Task (c) - iii: 3D Surface Plot of key comparisons for varying array
size and S
fig = plt.figure(figsize=(10, 6))
ax = fig.add_subplot(111, projection='3d')

# Prep data for 3D plot
X = new_partciii_df['ArraySize'].values
Y = new_partciii_df['S'].values
Z = new_partciii_df['Average Comparisons'].values

# Meshgrid for array size and S
X_unique = np.unique(X)
Y_unique = np.unique(Y)
X_mesh, Y_mesh = np.meshgrid(X_unique, Y_unique)
```

```
# Reshape Z values to fit meshgrid
Z_mesh = Z.reshape(len(Y_unique), len(X_unique))

# Plot surface
surf = ax.plot_surface(X_mesh, Y_mesh, Z_mesh, cmap='viridis',
edgecolor='none')

# Add labels and title
ax.set_title('3D Surface Plot of Average Comparisons for Varying Array
Size and S')
ax.set_xlabel('Array Size')
ax.set_ylabel('S (Threshold)')
ax.set_zlabel('Average Comparisons', labelpad=5)
ax.zaxis.get_offset_text().set_position((0, 0.05))

# Add color bar to the plot
fig.colorbar(surf, pad=0.1)
plt.tight_layout()
plt.show()

# Task (d): Compare original Merge Sort and Hybrid Merge-Insertion
Sort (from partd_df)
plt.figure(figsize=(12, 6))

# Plot Avg Comparisons for Merge Sort and Hybrid Merge-Insertion Sort
plt.subplot(1, 2, 1)
plt.bar(new_partd_df['Type'], new_partd_df['Average Comparisons'],
color=['blue', 'orange'])
plt.title('Average Comparisons for Merge Sort vs Hybrid Merge-
Insertion Sort')
plt.ylabel('Average Comparisons  ')

# Plot CPU Time (Perf) for Merge Sort and Hybrid Merge-Insertion Sort
plt.subplot(1, 2, 2)
plt.bar(new_partd_df['Type'], new_partd_df['Performance (ns)'],
color=['blue', 'orange'])
plt.title('CPU Time for Merge Sort vs Hybrid Merge-Insertion Sort')
plt.ylabel('Performance (ns)')

plt.tight_layout()
plt.show()
```


Average Comparisons vs Array Size (S fixed)
1e8
2.0
1.5
1.0
0.5
0.0
Average Comparisons
10^3
10^4
10^5
10^6
10^7
Array Size


Average Comparisons vs S (Array size fixed)
1e7
3.0
2.8
2.6
2.4
2.2
2.0
Average Comparisons
0
20
40
60
80
100
S (Threshold for Switching to Insertion Sort)





Created in Deepnote