Name: Rishabh Jain

Roll No: A3_54

Aim: Write a program read the file and find Minimum and Maximum using Brute force approach. Also propose and implement an improved algorithm based on Divide and Conquer Strategy. Also calculate the time taken by both the algotihm.Compute the time complexity of

1. **Brute force algorithm**
2. **Divide and Conquer based algorithm**
3. **Plot a graph to show the time comparison between algorithms**

DAA Practical 1

For data500.csv dataset

```
import time
import csv

# Function to find the minimum and maximum values in a list of numbers using brute force
def min_max (numbers):
    if not numbers:
        return None,None
    min=max=numbers[0]
    for num in numbers:
        if num<min:
            min=num
        elif num>max:
            max=num
    return min,max

# Function to find the minimum and maximum values in a list of numbers using divide and conquer (merge)
def merge(numbers,l,h):
    if(l==h):
        return numbers[l],numbers[l]
    if((h-l)==1):
        if numbers[l]< numbers[h]:
            return (numbers[l],numbers[h])
        else:
            return (numbers[h],numbers[l])
    mid =(l+h)//2
    min1,max1 = merge(numbers,l,mid)
    min2,max2 = merge(numbers,mid+1,h)
    return min(min1,min2),max(max1,max2)

# Function to read numbers from a CSV file and return them as a list
def readf(filename):
    with open(filename,'r') as file:
        reader=csv.reader(file)
        numbers =[int(row[0]) for row in reader]
    return numbers

# Specify the CSV file name
filename="/content/drive/MyDrive/DAA_Colab/data500.csv"

# Read numbers from the CSV file
numbers=readf(filename)

# Calculate the minimum and maximum using brute force
start=time.perf_counter()
min3,max3 =min_max(numbers)
end= time.perf_counter()
timetaken=end-start

# Print results for brute force method
print("Brute force")
print(f"Min number is {min3}")
print(f"Max number is {max3}")
print(f"End time is {end}")
print(f"Start time is {start}")
print(f"Total time is {timetaken}")
print(f"Total time is {timetaken}")

# Calculate the minimum and maximum using merge sort (divide and conquer)
start1=time.perf_counter()
minimum,maximum=merge(numbers,0,len(numbers)-1)
end1= time.perf_counter()
timetaken1=end1-start1

# Print results for merge sort method
print("Merge sort")
print(f"Min number is {minimum}")
print(f"Max number is {maximum}")
print(f"End time is {end1}")
print(f"Start time is {start1}")
print(f"Total time is {timetaken1}")
```

```
Brute force
Min number is 4
Max number is 998
End time is 2647.196622925
Start time is 2647.196463825
Total time is 0.00015910000001895241
Merge sort
Min number is 4
Max number is 998
End time is 2647.197923629
Start time is 2647.197357565
Total time is 0.0005660639999405248
```

For test1.csv dataset

```
import time
import csv

# Function to find the minimum and maximum values in a list of numbers using brute force
def min_max(numbers):
    if not numbers:
        return None, None
    min_val = max_val = numbers[0]
    for num in numbers:
        if num < min_val:
            min_val = num
        elif num > max_val:
            max_val = num
    return min_val, max_val

# Function to find the minimum and maximum values in a list of numbers using divide and conquer (merge)
def merge(numbers, l, h):
    if l == h:
        return numbers[l], numbers[l]
    if (h - l) == 1:
        if numbers[l] < numbers[h]:
            return numbers[l], numbers[h]
        else:
            return numbers[h], numbers[l]
    mid = (l + h) // 2
    min1, max1 = merge(numbers, l, mid)
    min2, max2 = merge(numbers, mid + 1, h)
    return min(min1, min2), max(max1, max2)

# Function to read numbers from a CSV file and return them as a list
def readf(filename):
    with open(filename, 'r') as file:
        reader = csv.reader(file)
        numbers = [float(row[4]) for row in reader]
    return numbers

# Specify the CSV file name
filename = "/content/drive/MyDrive/DAA_Colab/test1.csv"

# Read numbers from the CSV file
numbers = readf(filename)

# Calculate the minimum and maximum using brute force
start = time.perf_counter()
```

```
min3, max3 = min_max(numbers)
end = time.perf_counter()
timetaken = end - start

# Print results for brute force method
print("Brute force")
print(f"Min number is {min3}")
print(f"Max number is {max3}")
print(f"End time is {end}")
print(f"Start time is {start}")
print(f"Total time is {timetaken} seconds")

# Calculate the minimum and maximum using merge sort (divide and conquer)
start1 = time.perf_counter()
minimum, maximum = merge(numbers, 0, len(numbers) - 1)
end1 = time.perf_counter()
timetaken1 = end1 - start1

# Print results for merge sort method
print("Merge sort")
print(f"Min number is {minimum}")
print(f"Max number is {maximum}")
print(f"End time is {end1}")
print(f"Start time is {start1}")
print(f"Total time is {timetaken1} seconds")
```

```
Brute force
Min number is 34.24
Max number is 41.03
End time is 2698.021088421
Start time is 2698.020976413
Total time is 0.00011200800008737133 seconds
Merge sort
Min number is 34.24
Max number is 41.03
End time is 2698.024760911
Start time is 2698.024576233
Total time is 0.00018467800009602797 seconds
```

For test2.csv dataset

```
import time
import csv

# Function to find the minimum and maximum values in a list of numbers using brute force
def min_max (numbers):
    if not numbers:
        return None,None
    min=max=numbers[0]
    for num in numbers:
        if num<min:
            min=num
        elif num>max:
            max=num
    return min,max

# Function to find the minimum and maximum values in a list of numbers using divide and conquer (merge)
def merge(numbers,l,h):
    if(l==h):
        return numbers[l],numbers[l]
    if((h-l)==1):
        if numbers[l]< numbers[h]:
            return (numbers[l],numbers[h])
        else:
            return (numbers[h],numbers[l])
    mid =(l+h)//2
    min1,max1 = merge(numbers,l,mid)
    min2,max2 = merge(numbers,mid+1,h)
    return min(min1,min2),max(max1,max2)

# Function to read numbers from a CSV file and return them as a list
def readf(filename):
    with open(filename,'r') as file:
        reader=csv.reader(file)
        numbers =[int(row[7]) for row in reader]
    return numbers

# Specify the CSV file name
filename="/content/drive/MyDrive/DAA_Colab/test2.csv"

# Read numbers from the CSV file
numbers=readf(filename)

# Calculate the minimum and maximum using brute force
start=time.perf_counter()
min3,max3 =min_max(numbers)
end= time.perf_counter()
timetaken=end-start

# Print results for brute force method
print("Brute force")
print(f"Min number is {min3}")
print(f"Max number is {max3}")
print(f"End time is {end}")
print(f"Start time is {start}")
print(f"Total time is {timetaken} seconds")

# Calculate the minimum and maximum using merge sort (divide and conquer)
start1=time.perf_counter()
minimum,maximum=merge(numbers,0,len(numbers)-1)
end1= time.perf_counter()
timetaken1=end1-start1

# Print results for merge sort method
print("Merge sort")
print(f"Min number is {minimum}")
print(f"Max number is {maximum}")
print(f"End time is {end1}")
print(f"Start time is {start1}")
print(f"Total time is {timetaken1}")
```

```
Brute force
Min number is 0
Max number is 208
End time is 2783.424569934
Start time is 2783.291154805
Total time is 0.13341512899978625 seconds
Merge sort
Min number is 0
Max number is 208
End time is 2784.544408712
Start time is 2783.425502742
Total time is 1.1189059700000143
```

For test3.csv dataset

```
import time
import csv
import pandas as pd

# Function to find the minimum and maximum values in a list of numbers using brute force
def min_max (numbers):
    if not numbers:
        return None,None
    min=max=numbers[0]
    for num in numbers:
        if num<min:
            min=num
        elif num>max:
            max=num
    return min,max

# Function to find the minimum and maximum values in a list of numbers using divide and conquer (merge)
def merge(numbers,l,h):
    if(l==h):
        return numbers[l],numbers[l]
    if((h-l)==1):
        if numbers[l]< numbers[h]:
            return (numbers[l],numbers[h])
        else:
            return (numbers[h],numbers[l])
    mid =(l+h)//2
    min1,max1 = merge(numbers,l,mid)
    min2,max2 = merge(numbers,mid+1,h)
    return min(min1,min2),max(max1,max2)

# Function to read numbers from a CSV file and return them as a list
def readf(filename):
    with open(filename,'r') as file:
        reader=csv.reader(file)
        numbers =[float(row[4]) for row in reader]
    return numbers

# Specify the CSV file name
filename="/content/drive/MyDrive/DAA_Colab/test3.csv"

# Read numbers from the CSV file
numbers=readf(filename)

# Calculate the minimum and maximum using brute force
start=time.perf_counter()
min3,max3 =min_max(numbers)
end= time.perf_counter()
timetaken=end-start

# Print results for brute force method
print("Brute force")
print(f"Min number is {min3}")
print(f"Max number is {max3}")
print(f"End time is {end}")
print(f"Start time is {start}")
print(f"Total time is {timetaken} seconds")

# Calculate the minimum and maximum using merge sort (divide and conquer)
start1=time.perf_counter()
minimum,maximum=merge(numbers,0,len(numbers)-1)
end1= time.perf_counter()
timetaken1=end1-start1

# Print results for merge sort method
print("Merge sort")
print(f"Min number is {minimum}")
print(f"Max number is {maximum}")
print(f"End time is {end1}")
print(f"Start time is {start1}")
print(f"Total time is {timetaken1} seconds")
```

```
Brute force
Min number is 1.59
Max number is 2049.0
End time is 7042.28233422
Start time is 7042.243854394
Total time is 0.0384798260001844 seconds
Merge sort
Min number is 1.59
Max number is 2049.0
End time is 7042.610190301
Start time is 7042.283811686
Total time is 0.32637861500006693 seconds
```

For test4.csv dataset

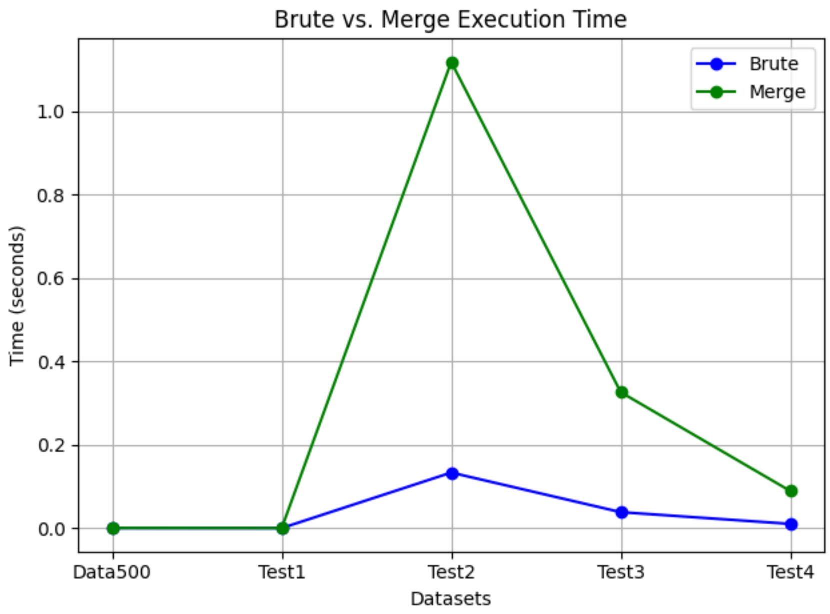
```
import time
import csv

# Function to find the minimum and maximum values in a list of numbers using brute force
def min_max (numbers):
    if not numbers:
        return None,None
    min=max=numbers[0]
    for num in numbers:
        if num<min:
            min=num
        elif num>max:
            max=num
    return min,max

# Function to find the minimum and maximum values in a list of numbers using divide and conquer (merge)
def merge(numbers,l,h):
    if(l==h):
        return numbers[l],numbers[l]
    if((h-l)==1):
        if numbers[l]< numbers[h]:
            return (numbers[l],numbers[h])
        else:
            return (numbers[h],numbers[l])
    mid =(l+h)//2
    min1,max1 = merge(numbers,l,mid)
    min2,max2 = merge(numbers,mid+1,h)
    return min(min1,min2),max(max1,max2)

# Function to read numbers from a CSV file and return them as a list
def readf(filename):
    # with open(filename,'r') as file:
    #     reader=csv.reader(file)
    #     numbers =[float(row[11]) for row in reader]
    # return numbers
    numbers = []
    with open(filename, 'r') as file:
        reader = csv.reader(file)
        for row in reader:
            try:
                # Extract data from the 12th column (index 11)
                value = float(row[11])
                numbers.append(value)
            except (ValueError, IndexError):
                # Handle non-numeric values or empty strings
                pass
    return numbers

# Specify the CSV file name
filename="/content/drive/MyDrive/DAA_Colab/test4.csv"

# Read numbers from the CSV file
numbers=readf(filename)

# Calculate the minimum and maximum using brute force
start=time.perf_counter()
min3,max3 =min_max(numbers)
end= time.perf_counter()
timetaken=end-start

# Print results for brute force method
print("Brute force")
print(f"Min number is {min3}")
print(f"Max number is {max3}")
print(f"End time is {end}")
print(f"Start time is {start}")
print(f"Total time is {timetaken}")

# Calculate the minimum and maximum using merge sort (divide and conquer)
start1=time.perf_counter()
minimum,maximum=merge(numbers,0,len(numbers)-1)
end1= time.perf_counter()
timetaken1=end1-start1

# Print results for merge sort method
print("Merge sort")
print(f"Min number is {minimum}")
print(f"Max number is {maximum}")
print(f"End time is {end1}")
print(f"Start time is {start1}")
print(f"Total time is {timetaken1}")
```

```
Brute force
Min number is 0.0
Max number is 51427.491
End time is 6904.281193495
Start time is 6904.270919045
Total time is 0.01027444999999716
Merge sort
```

```
Min number is 0.0
Max number is 51427.491
End time is 6904.371214177
Start time is 6904.281930132
Total time is 0.08928404500056786
```

Plotting the difference between the execution times of Brute force vs Merge sort

```
import matplotlib.pyplot as plt

# Data
datasets = ["Data500", "Test1", "Test2", "Test3", "Test4"]
brute = [0.00015, 0.00011, 0.13341, 0.03847, 0.01027]
merge = [0.00056, 0.00018, 1.11891, 0.32637, 0.08928]

# Create a line plot for Brute
plt.plot(datasets, brute, marker='o', label='Brute', color='b')

# Create a line plot for Merge
plt.plot(datasets, merge, marker='o', label='Merge', color='g')

# Label the x-axis
plt.xlabel('Datasets')

# Label the y-axis
plt.ylabel('Time (seconds)')

# Add a legend
plt.legend()

# Add a title
plt.title('Brute vs. Merge Execution Time')

# Show the plot
plt.grid(True)
plt.tight_layout()
plt.show()
```


Brute vs. Merge Execution Time
Brute
Merge
Time (seconds)
Data500
Test1
Test2
Test3
Test4
Datasets