Department of Software / Computer Engineering

SENG114 & CENG104 – Computer Programming II

Spring 2021 - 2022

Lab Guide #6 – Week 7

OBJECTIVE : Usage of Binary Search and Merge Algorithms

Instructor : Yusuf Evren AYKAÇ
Assistants : Elif ŞANLIALP, Ahmet Esad TOP, Nisanur MÜHÜRDAROĞLU MERCİMEK

BINARY SEARCH ALGORITHM:

```
1. Let top be the subscript of the initial array element.
2. Let bottom be the subscript of the last array element.
3. Repeat until top exceeds bottom, thus there are no more elements to check
  4. Let middle be the subscript of the element halfway from top to bottom.
  5. If the element at middle is the target, than return middle.
  6. else  if the element at middle is larger than the target, let bottom be middle-1,
     thus continue the search in the first half.
  7. else let top be middle+1, thus continue the search in the second half.
8. Return -1 since the loop terminated, but the number is not found.
```

1. **a)** Write a C Program which search a record of a car plate from the sorted data in **plates.txt** produced in. Use sequential search, and prints how many comparisons made to find the record.

Example Run:

```
Enter plate of a car (END for exit): 06HTC452
EFE KOROGLU 06HTC452
14 comparisons.

Enter plate of a car (END for exit): 01KL5641
BARIS COLAK 01KL5641
1 comparisons.

Enter plate of a car (END for exit): 53EF4587
DEMIRCAN COSKUN 53EF4587
37 comparisons.

Enter plate of a car (END for exit): 06GHK567
NOT FOUND
37 comparisons.

Enter plate of a car (END for exit): END
```

plates.txt

```
BARIS    COLAK        01KL5641
DENIZ    CORBACI      01RR5678
DENIZ    AGAH         06ABC453
OZAN     CERGEL       06BA1246
BARIS    ARSLAN       06BB9987
ERGIN    ERANT        06BLN027
ABDULLAH ARSLAN       06CAN063
EMRE     DINCEL       06CRN027
MUSTAFA  KAPLAN       06EMN652
FATMA    KAYA         06ERD063
BERAT    KINA         06ERK751
OZGUR    AKBABA       06ES4587
HUSEYIN  KINIKLI      06GNS697
EFE      KOROGLU      06HTC452
CELAL    ERBAY        06JAL254
ABDULLAH AKTAS        06KM3657
ONUR     BAGDADIOGLU  06KRM323
AYLIN    GURTUNA      06KRM442
BATURAY  HASER        06MNV660
MERTHAN  ILVAN        06OIL633
BARIS    INALOZ       06PRL471
ESIN     INANC        06REN694
DENIZ    INTEPE       06RMZ458
CAN      KANBAY       06VYS452
MUSTAFA  BOLAT        07ABC487
BILGEN   BILGIN       07TRK658
EDIZ     CITAK        19HJ1245
ALI      CICEK        19HT6547
DERYA    DEDEOGLU     27CNR006
AYSE     BAYHAN       34KLM365
ZEYNEP   AKANDIR      34ZZ1785
MERT     AKDEMIR      35ED5678
TAMER    CAN          35TT2369
CUNEYT   EKINCI       35ZRF006
GAMZE    BASIBUYUK    42HKN451
ABDULLAH BATTAL       42MN4178
```

b) Modify the program using Binary Search algorithm.

Example Run:

```
Enter plate of a car (END for exit): 06HTC452
EFE KOROGLU 06HTC452
3 comparisons.

Enter plate of a car (END for exit): 01KL5641
BARIS COLAK 01KL5641
5 comparisons.

Enter plate of a car (END for exit): 53EF4587
DEMIRCAN COSKUN 53EF4587
6 comparisons.

Enter plate of a car (END for exit): 06GHK567
NOT FOUND
6 comparisons.

Enter plate of a car (END for exit): END
```

2. Write a C program that gets the names of bicycles from two different sorted files named **bike1.txt** and **bike2.txt** into two arrays, and merges them and writes in a new file named **bike_list.txt.**

Write the following functions;

- **`readBikeList`** that takes the file pointer and the array which will keep the bike list as parameters, reads the bike names from the file into the specified array. The function should also return the number of bikes in the array.
- **`shiftDown`** that takes the bike list array, number of bikes and the position of the bike which will be shifted down as parameters. The function moves down all of the elements starting from specified position in the array.
- **`merge`** that takes two bike lists and their sizes as parameters, merges two array putting the bike names in the second array into the first array in a sorted form.

In main call the necessary functions.

Example Run:

bike1.txt
```
bianchi
bisan
colnago
gazelle
look
raleigh
trek
```

bike2.txt
```
beldesan
beldeyama
bisan
bmx
giant
kona
marin
pinokyo
polo
salcano
scott
```

bike_list.txt
```
beldesan
beldeyama
bianchi
bisan
bisan
bmx
colnago
gazelle
giant
kona
look
marin
pinokyo
polo
raleigh
salcano
scott
trek
```

Note that: If we want to eliminate the **duplicate** bike names, then remember the lecture notes..