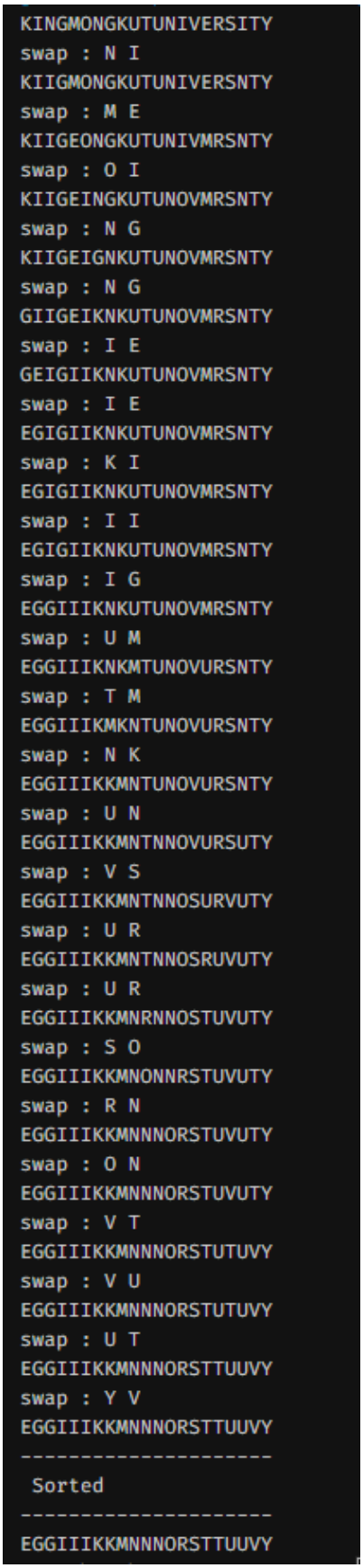
Hw sorting

1. จงเรียงลำดับข้อมูลต่อไปนี้ จากน้อยไปมาก ด้วยวิธีการ Quick Sort algorithm

```
#include <stdio.h>
#include <string.h>

void swap(char *a, char *b) {
    int temp;
    temp = *a;
    *a = *b;
    *b = temp;
}

int partition(char str[], int l, int r) {
    int p = str[l];
    int j = r;
    int i = l;

    while (1) {   // repeat
        i = i + 1;
        while(str[i] <= p && i <= j  ) {        // repeat untill A[i] >= p
            i++;
        }
        while (str[j] >= p && j >= i) {         // repeat untill A[j] <= p
            j--;
        }
        if (j < i) {       // repeat untill i >= j
            break;
        } else {
            printf("swap : %c %c\n", str[i], str[j]);
            swap(&str[i], &str[j]);  //swap(A[i], A[j])
            printf("%s\n", str);
        }
    }
    printf("swap : %c %c\n", str[i], str[j]);
    swap(&str[l], &str[j]);  //swap(A[l], A[j])
    printf("%s\n", str);
    return j;
}

void qiuckSort(char str[], int l, int r) {
    int s;
    if (l < r) {
```

```
        s = partition(str, l, r);
        qiuckSort(str, l, s - 1);
        qiuckSort(str, s+1, r);
    }
}

int main() {
    char str[64] = "KINGMONGKUTUNIVERSITY";
    int size = strlen(str);
    printf("%s\n", str);
    qiuckSort(str, 0, size - 1);        // send array to sort;
    printf("--------------------\n Sorted \n--------------------\n");
    printf("%s", str);                  // output

    return 0;
}
```

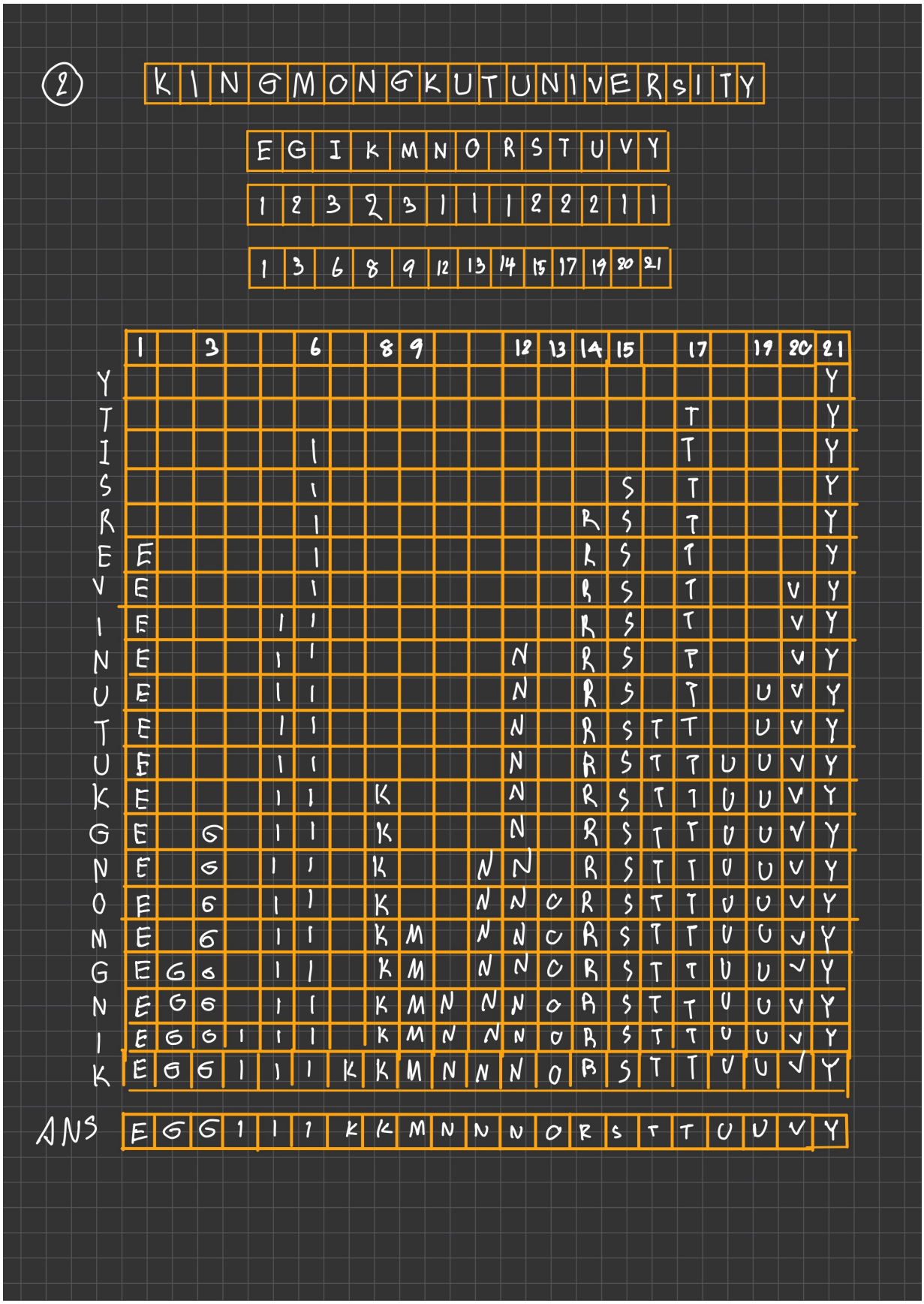
Output


KINGMONGKUTUNIVERSITY
swap : N I
KIIGMONGKUTUNIVERSNTY
swap : M E
KIIGEONGKUTUNIVMRSNTY
swap : O I
KIIGEINGKUTUNOVMRSNTY
swap : N G
KIIGEIGNKUTUNOVMRSNTY
swap : N G
GIIGEIKNKUTUNOVMRSNTY
swap : I E
GEIGIIKNKUTUNOVMRSNTY
swap : I E
EGIGIIKNKUTUNOVMRSNTY
swap : K I
EGIGIIKNKUTUNOVMRSNTY
swap : I I
EGIGIIKNKUTUNOVMRSNTY
swap : I G
EGGIIIKNKUTUNOVMRSNTY
swap : U M
EGGIIIKNKMTUNOVURSNTY
swap : T M
EGGIIIKMKNTUNOVURSNTY
swap : N K
EGGIIIKKMNTUNOVURSNTY
swap : U N
EGGIIIKKMNTNNOVURSUTY
swap : V S
EGGIIIKKMNTNNOSURVUTY
swap : U R
EGGIIIKKMNTNNOSRUVUTY
swap : U R
EGGIIIKKMNRNNOSTUVUTY
swap : S O
EGGIIIKKMNONNRSTUVUTY
swap : R N
EGGIIIKKMNNNORSTUVUTY
swap : O N
EGGIIIKKMNNNORSTUVUTY
swap : V T
EGGIIIKKMNNNORSTUTUVY
swap : V U
EGGIIIKKMNNNORSTUTUVY
swap : U T
EGGIIIKKMNNNORSTTUUVY
swap : Y V
EGGIIIKKMNNNORSTTUUVY

Sorted

EGGIIIKKMNNNORSTTUUVY

2. จงเรียงลำดับข้อมูลก่อนหน้านี้ จากน้อยไปมาก ด้วยวิธีการ Distribution Counting Sort algorithm


2
KINGMONGKUTUNIVERSITY
EGIKMNORSTUVY
1 2 3 2 3 1 1 1 2 2 2 1 1
1 3 6 8 9 12 13 14 15 17 19 20 21
YTISREVINUTUKGNOMGNIK
ANS EGG1117KKMNNNNORSTTUUVY

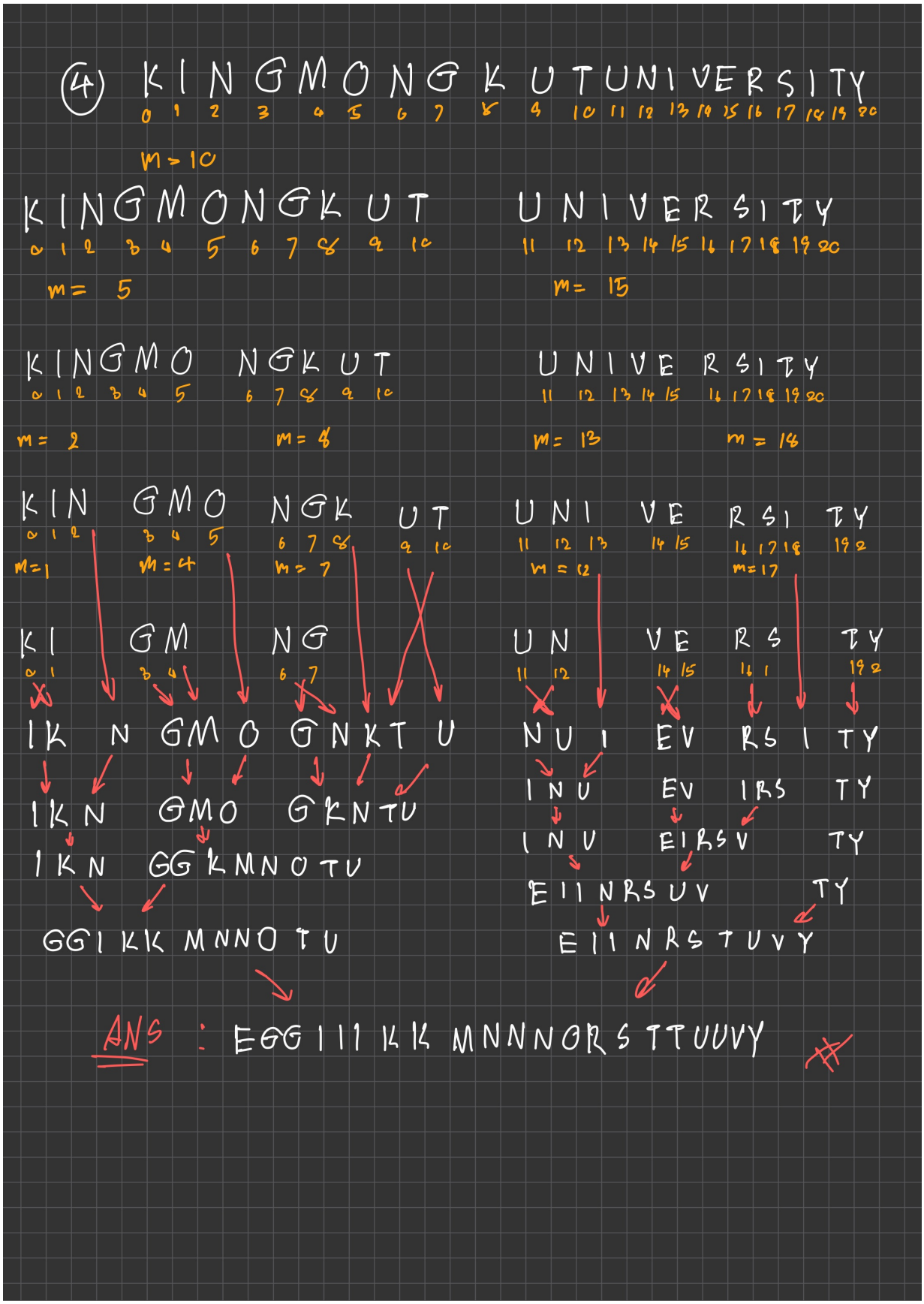
3. จง Implement Merge Sort algorithm ด้วยภาษาซี

```
#include <stdio.h>
#include <string.h>
#include <math.h>

void copyStr(char arr[], int indexArr, int endArr, char str[], int indexStr, int endStr) {
    while (indexArr <= endArr) {
        str[indexStr] = arr[indexArr];
        indexArr++;
        indexStr++;
    }
}

void merge(char arr[], int l, int m, int r) {
    int n1 = m - l + 1;
    int n2 = r - m;
    int i = 0;
    int j = 0;
    int k = 0;
    char str[64];

    while (i < n1 && j < n2) {
        if (arr[l + i] < arr[m + 1 + j]) {
            str[k] = arr[l + i];
            i = i + 1;
        } else {
            str[k] = arr[m + 1 + j];
            j = j + 1;
        }
        k = k + 1;
    }
    if (i == n1) {
        copyStr(arr, m + 1 + j, r, str, k, n1+n2-1);
    } else {
        copyStr(arr, l + i, m, str, k, n1+n2-1);
    }
    copyStr(str, 0, n1 + n2 - 1, arr, l, r);
}

void mergeSort(char str[], int l, int r) {

   if (r > l) {
        int m = (l + r) / 2;
        mergeSort(str, l, m);
        mergeSort(str, m+1, r);
        merge(str, l, m, r);
        printf("l = %d r = %d %s\n", l, r, str);
   }

}
```

```
int main() {
    char str[64] = "KINGMONGKUTUNIVERSITY";
    int size = strlen(str);
    mergeSort(str, 0, size-1);

    printf("--------------------\n Sorted \n--------------------\n");
    printf("%s", str);                  // output

    return 0;
}
```

Output

4. จงเรียงลำดับข้อมูลด้านบน จากน้อยไปมาก ด้วยวิธีการ Merge Sort algorithm


④ KINGMONGKUTUNIVERSITY
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
m = 10
KINGMONGKUT
0 1 2 3 4 5 6 7 8 9 10
m = 5
UNIVERSITY
11 12 13 14 15 16 17 18 19 20
m = 15
KINGMO
0 1 2 3 4 5
m = 2
NGKUT
6 7 8 9 10
m = 8
UNIVE
11 12 13 14 15
m = 13
RSITY
16 17 18 19 20
m = 16
KIN
0 1 2
m = 1
GMO
3 4 5
m = 4
NGK
6 7 8
m = 7
UT
9 10
UNI
11 12 13
m = 12
VE
14 15
RSI
16 17 18
m = 17
TY
19 20
KI
0 1
GM
3 4
NG
6 7
UN
11 12
VE
14 15
RS
16 1
TY
19 2
IK N GMO GNKT U
NU I EV RS I TY
IKN GMO GKNTU
INU EV IRS TY
IKN GGKMNOTU
INU EIRSV TY
GGIKK MNNOTU
EIINRSUV TY
EIINRSTUVY
ANS : EGGIIIKK MNNNORSTTUUVY