

Mir Hossain

CS477

HW3

1A.) The enigma algorithm checks

If there is an equivalent amount
of even and odd elements in
the array.

Ex: $[1, 3, 4, 6] \rightarrow \text{False}$; $[1, 3, 5, 4, 6] \rightarrow \text{True}$

1B.) $\sum_{i=0}^{n-1} ; \sum_{i=1}^{n-1} 1$

$$\hookrightarrow \sum_{i=0}^{n-2} [f(n-1) - (i+1) + 1] = \sum_{i=0}^{n-2} (n-i-1)$$

$$= [(n-1) + (n-2) + \dots + \frac{(n-1)n}{2}] \Rightarrow \frac{n^2 - n}{2} \Rightarrow n^2 \text{ is the}$$

dominating term, so the running time
is $O(n^2)$

Ans: $O(n^2)$

2A.) Code for Bubble Sort on attached file & Screenshot
below.

Also 2B will be under the screenshot

3.) Mergesort Also Attached

hw3-2_bubbleSort

Before Bubble Sort :

6 5 2 8 3 1

Element 5 was swapped with 6
Element 2 was swapped with 6
Element 3 was swapped with 8
Element 1 was swapped with 8
Element 2 was swapped with 5
Element 3 was swapped with 6
Element 1 was swapped with 6
Element 3 was swapped with 5
Element 1 was swapped with 5
Element 1 was swapped with 3
Element 1 was swapped with 2

After Bubble Sort :

1 2 3 5 6 8

Testing BubbleSort on Char Array

Before Bubble Sort :

E A S Y Q U E S T I O N

Element A was swapped with E
Element Q was swapped with Y
Element U was swapped with Y
Element E was swapped with Y
Element S was swapped with Y
Element T was swapped with Y
Element I was swapped with Y
Element O was swapped with Y
Element N was swapped with Y
Element Q was swapped with S
Element E was swapped with U
Element S was swapped with U
Element T was swapped with U
Element I was swapped with U
Element O was swapped with U
Element N was swapped with U
Element E was swapped with S
Element I was swapped with T
Element O was swapped with T
Element N was swapped with T
Element E was swapped with Q
Element I was swapped with S
Element O was swapped with S
Element N was swapped with S
Element I was swapped with S
Element O was swapped with S
Element N was swapped with S
Element I was swapped with Q
Element O was swapped with Q
Element N was swapped with Q
Element N was swapped with O

After Bubble Sort :

A E E I N O Q S S T U Y

2B.) 11 swaps for int array

31 swaps for char array

3.)MergeSort

```
problem3
Array before merge sort:A S O R T I N G E X A M P L E
Merged Array: A
Merged Array: A S O
Merged Array: A S O R I
Merged Array: A S O R I T G
Merged Array: A S O R I T G N E
Merged Array: A S O R I T G N E X A
Merged Array: A S O R I T G N E X A M L
Merged Array: A O R
Merged Array: A O R S G I N
Merged Array: A O R S G I N T A E M
Merged Array: A O R S G I N T A E M X E L
Merged Array: A G I N O R S
Merged Array: A G I N O R S T A E E L M P
Merged Array: A A E E G I L M N O P R S T
Press any key to continue . . .
```

4) Use Loop Invariant to show a^n is computed

Basis $i=1 \Rightarrow$ before the loop starts, $i=1 \Rightarrow$

$i-1=0$, so $a^{i-1}=1$, \Rightarrow Basis is correct

Inductive Step: For any i , $i \leq n$, so a^{i-1}

$\Rightarrow i$ will never be greater than n , so $\text{pow} = (a^{i-1})a = a^i$. Value of i increase by 1, on when

$i=n$, we have $\text{pow} = a^n$.

\downarrow For any Value of n , we get $\text{pow} = a^n$.