

Problem 01 [5 marks]

Given an array of unsorted integers, use the divide and conquer method to sort them in ascending order. (You are not allowed to use any built-in sort function. Built-in data structures are allowed.)

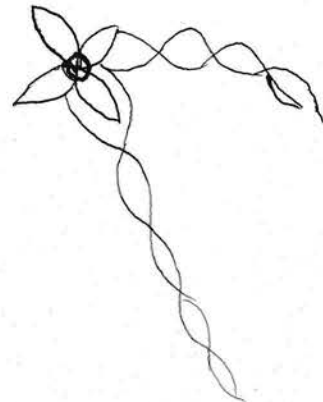
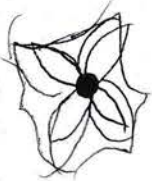
Problem 02 [3 marks]

In a stock market, there is a product with its infinite stocks. The stock prices are given for N days, where $arr[i]$ denotes the price of the stock on the i^{th} day. There is a rule that a customer can buy at most i stock on the i^{th} day. If the customer has k amount of money initially, find out the maximum number of stocks a customer can buy.

For example, for 3 days the price of a stock is given as 7, 10, 4. You can buy 1 stock worth 7 taka on day 1, 2 stocks worth 10 taka each on day 2 and 3 stock worth 4 taka each on day 3.

Pseudocode of the algorithm required to solve this problem is as follows:

1. Take input an array $price[n]$, where $price[i]$ is the price of stock on day i
2. Take input max_spend
3. $min = 99999$, $day = -1$, $total_stock = 0$
4. for $i = 0$ to $n-1$:
5. if $price[i] < min$:
6. $min = price[i]$
7. $day = i+1$
8. If $min*day \leq max_spend$:
9. $max_spend -= min*day$
10. $total_stock += day$
11. Else:
12. $max_spend -= (max_spend/min)*min$;
13. $total_stock += (max_spend/min)$
14. Break
15. $Price[day-1] = 99999$
16. $min = 99999$
17. Output t



Problem 03 [2 marks]

Given an array, find the GCD of all array elements.

Sample Input:

12 16 20 28

Sample Output:

4