

Name: Rajdeep Bandopadhyay

M12871145

Answer 1:

```
int gcd(int a, int b)
{
    if (a == 0)
        return b;
    return gcd(b % a, a);
}
```

using the function above I was able to compute the GCD for the 2 given set of numbers:

1. 124 and 730 -> GCD is 2
2. 11235 and 89350 -> GCD is 5

Answer 2:

```
int poly[] = {2, -6, 2, -1}; // List of co-efficients
int x = 3; // So this -V
x = x * (-1);
int n = sizeof(poly)/sizeof(poly[0]);
int result = poly[0];
```

```
for (int i=1; i<n; i++)  
result = result*x + poly[i];  
return result; // returns f(-v) using Horner's rule
```

Answer 3:

> O (log (N))

> O (log (N))

> O (1)

Answer 4:

```
void selectionSort(int arr[], int n)  
{  
    int i, j, min_idx;  
    // One by one move boundary of unsorted subarray  
    for (i = 0; i < n-1; i++)  
    {  
        // Find the minimum element in unsorted array  
        min_idx = i;  
        for (j = i+1; j < n; j++)  
            if (arr[j] < arr[min_idx])  
                min_idx = j;
```

```
// Swap the found minimum element with the first element
```

```
swap(&arr[min_idx], &arr[i]);
```

```
}
```

```
}
```

```
int main () {
```

```
int arr[] = {64, 25, 12, 22, 11};
```

```
int n = sizeof(arr)/sizeof(arr[0]);
```

```
selectionSort(arr, n);
```

```
cout << arr [0] ", " << arr [1];
```

```
}
```

Worst complexity: n^2

Average complexity: n^2

Best complexity: n^2