1. Sum of Even Numbers in 1D Array

```
import java.util.Scanner;
public class Program1 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), sum = 0;
        for(int i = 0; i < n; i++) {
            int val = sc.nextInt();
            if(val % 2 == 0) sum += val;
        }
        System.out.println("Sum of even numbers: " + sum);
    }
}
```

2. Palindrome String Check

```
import java.util.Scanner;
public class Program2 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String s = sc.next();
        String rev = "";
        for(int i = s.length()-1; i >= 0; i--) rev += s.charAt(i);
        if(s.equals(rev)) System.out.println("Palindrome");
        else System.out.println("Not a palindrome");
    }
}
```

3. Factor of a Number (Iterative & Recursive)

```
import java.util.Scanner;
public class Program3 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        System.out.print("Iterative: ");
        for(int i = 1; i <= n; i++) if(n % i == 0) System.out.print(i + " ");
        System.out.print("\nRecursive: ");
        printFactors(n, 1);
    }
    static void printFactors(int n, int i) {
        if(i > n) return;
        if(n % i == 0) System.out.print(i + " ");
        printFactors(n, i + 1);
    }
}
```

4. Factorial (Iterative & Recursive)

```
import java.util.Scanner;
public class Program4 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), fact = 1;
        for(int i = 1; i <= n; i++) fact *= i;
        System.out.println("Iterative: " + fact);
        System.out.println("Recursive: " + factorial(n));
    }
    static int factorial(int n) {
        if(n == 0) return 1;
        return n * factorial(n - 1);
    }
}
```

5. Sum of Digits (Iterative & Recursive)

```
import java.util.Scanner;
public class Program5 {
```

```
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), sum = 0, temp = n;
        while(temp > 0) { sum += temp % 10; temp /= 10; }
        System.out.println("Iterative: " + sum);
        System.out.println("Recursive: " + sumDigits(n));
    }
    static int sumDigits(int n) {
        if(n == 0) return 0;
        return n % 10 + sumDigits(n / 10);
    }
}
```

6. Print Pattern

```
public class Program6 {
    public static void main(String[] args) {
        for(int i = 1; i <= 5; i++) {
            for(int j = 1; j <= i; j++) System.out.print(j + " ");
            System.out.println();
        }
    }
}
```

7. Fibonacci (Iterative & Recursive)

```
import java.util.Scanner;
public class Program7 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), a = 0, b = 1;
        System.out.print("Iterative: ");
        for(int i = 0; i < n; i++) {
            System.out.print(a + " ");
            int c = a + b; a = b; b = c;
        }
        System.out.print("\nRecursive: ");
        for(int i = 0; i < n; i++) System.out.print(fibo(i) + " ");
    }
    static int fibo(int n) {
        if(n <= 1) return n;
        return fibo(n - 1) + fibo(n - 2);
    }
}
```

8. Count Odd and Even Digits

```
import java.util.Scanner;
public class Program8 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), even = 0, odd = 0;
        while(n > 0) {
            int d = n % 10;
            if(d % 2 == 0) even++;
            else odd++;
            n /= 10;
        }
        System.out.println("Even: " + even + ", Odd: " + odd);
    }
}
```

9. Odd or Even Without %

```
import java.util.Scanner;
public class Program9 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
```

```
        if((n & 1) == 0) System.out.println("Even");
        else System.out.println("Odd");
    }
}
```

10. Prime Numbers in Range

```
import java.util.Scanner;
public class Program10 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int start = sc.nextInt(), end = sc.nextInt();
        for(int i = start; i <= end; i++) {
            boolean prime = true;
            if(i < 2) continue;
            for(int j = 2; j * j <= i; j++) {
                if(i % j == 0) { prime = false; break; }
            }
            if(prime) System.out.print(i + " ");
        }
    }
}
```

11. Armstrong Number

```
import java.util.Scanner;
public class Program11 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), temp = n, sum = 0, len = ("" + n).length();
        while(temp > 0) {
            int d = temp % 10;
            int pow = 1;
            for(int i = 0; i < len; i++) pow *= d;
            sum += pow;
            temp /= 10;
        }
        System.out.println(sum == n ? "Armstrong" : "Not Armstrong");
    }
}
```

12. Max, Min, Avg without Array

```
import java.util.Scanner;
public class Program12 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), val = sc.nextInt();
        int max = val, min = val, sum = val;
        for(int i = 1; i < n; i++) {
            val = sc.nextInt();
            if(val > max) max = val;
            if(val < min) min = val;
            sum += val;
        }
        System.out.println("Max: " + max + " Min: " + min + " Avg: " + (sum / n));
    }
}
```

13. Matrix Multiplication

```
import java.util.Scanner;
public class Program13 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int r1 = sc.nextInt(), c1 = sc.nextInt();
        int[][] a = new int[r1][c1];
        int r2 = sc.nextInt(), c2 = sc.nextInt();
        int[][] b = new int[r2][c2];
```

```
        if(c1 != r2) return;
        for(int i = 0; i < r1; i++) for(int j = 0; j < c1; j++) a[i][j] = sc.nextInt();
        for(int i = 0; i < r2; i++) for(int j = 0; j < c2; j++) b[i][j] = sc.nextInt();
        int[][] res = new int[r1][c2];
          for(int i = 0; i < r1; i++) for(int j = 0; j < c2; j++) for(int k = 0; k < c1; k++)
res[i][j] += a[i][k] * b[k][j];
        for(int[] row : res) {
            for(int val : row) System.out.print(val + " ");
            System.out.println();
        }
    }
}
```

14. Clock Angle

```
import java.util.Scanner;
public class Program14 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int h = sc.nextInt(), m = sc.nextInt();
        double angle = Math.abs(30*h - 5.5*m);
        angle = Math.min(angle, 360 - angle);
        System.out.println("Angle: " + angle);
    }
}
```

15. Decimal to BCD

```
import java.util.Scanner;
public class Program15 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String dec = sc.next();
        for(char d : dec.toCharArray()) {
            int digit = d - '0';
             System.out.print(String.format("%4s", Integer.toBinaryString(digit)).replace(' ', '0')
+ " ");
        }
    }
}
```

16. Bubble Sort

```
import java.util.Scanner;
public class Program16 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), a[] = new int[n];
        for(int i = 0; i < n; i++) a[i] = sc.nextInt();
        for(int i = 0; i < n-1; i++)
            for(int j = 0; j < n-i-1; j++)
                if(a[j] > a[j+1]) {
                    int t = a[j]; a[j] = a[j+1]; a[j+1] = t;
                }
        for(int val : a) System.out.print(val + " ");
    }
}
```

17. Insertion Sort

```
import java.util.Scanner;
public class Program17 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), a[] = new int[n];
        for(int i = 0; i < n; i++) a[i] = sc.nextInt();
        for(int i = 1; i < n; i++) {
            int key = a[i], j = i - 1;
            while(j >= 0 && a[j] > key) a[j+1] = a[j--];
```

```
            a[j+1] = key;
        }
        for(int val : a) System.out.print(val + " ");
    }
}
```

18. Selection Sort

```
import java.util.Scanner;
public class Program18 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), a[] = new int[n];
        for(int i = 0; i < n; i++) a[i] = sc.nextInt();
        for(int i = 0; i < n-1; i++) {
            int min = i;
            for(int j = i+1; j < n; j++) if(a[j] < a[min]) min = j;
            int t = a[i]; a[i] = a[min]; a[min] = t;
        }
        for(int val : a) System.out.print(val + " ");
    }
}
```

19. Bucket Sort (Simple)

```
// Bucket sort is best suited for 0-1 float data, use built-in for simplicity
import java.util.*;
public class Program19 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(); Float[] a = new Float[n];
        for(int i = 0; i < n; i++) a[i] = sc.nextFloat();
        Arrays.sort(a);
        for(float f : a) System.out.print(f + " ");
    }
}
```

20. Power a^b without Math.pow or *

```
import java.util.Scanner;
public class Program20 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt(), b = sc.nextInt(), res = 1;
        for(int i = 0; i < b; i++) {
            int temp = 0;
            for(int j = 0; j < res; j++) temp += a;
            res = temp;
        }
        System.out.println("a^b = " + res);
    }
}
```