SUMMARY: LAB_5

```java
public class Lab2021_5 {
    public static void main(String[] args) {
        //Fibonacci(10);
        System.out.println(Horner(new int[] {4,3,-2,1,2},2));
        System.out.println(Bezuot(4,2,new int[] {4,3,-2,1,2}));
    }
    //BINOMIAL COEFFICIENT
    public static int binomial_coeff(int n, int k){
        //create method that will return factorial of the number n
        int val1 = Lab2021_3.factorialFOR(n);
        int val2 = Lab2021_3.factorialFOR(k);
        int val3 = Lab2021_3.factorialFOR(n-k);
        int res = val1/(val2*val3);
        return res;
    }
    //FIBONACCI SEQUENCE
    public static void Fibonacci(int n){
        int fib1 = 0, fib2 = 1;
        int[] fibArr = new int[n];
        for (int i=2; i<n; i++){
            fibArr[0]=fib1;
            fibArr[1]=fib2;
            fibArr[i]=fibArr[i-1]+fibArr[i-2];
        }
        //Now print an array (create method to print an array)
        ArrayManipulation.printArr(fibArr);
    }
    //HORNER's METHOD and BEZOUT's THEOREM
    public static int Horner(int[] coeff, int alpha){
        int n = coeff.length;
        int[] resArr = new int[n];
        resArr[0]=coeff[0];
        for (int i=1; i<n; i++){
            resArr[i] = alpha*resArr[i-1]+coeff[i];
        }
        return resArr[n-1];
    }
    public static int Bezuot(int n, int alpha, int[] coeff){
        int sum=0;
        for (int i=0; i<=n; i++){
            sum = (int) (sum+coeff[i]*Math.pow(alpha,n-i));
        }
        return sum;
    }
}
```

$$\binom{n}{k} = \frac{n!}{k!\cdot(n-k)!}, \binom{n}{0} = \binom{n}{1}=1$$

$$F_n = F_{n-1}+F_{n-2}, F_0 = 0, F_1 = 1$$

	a_n	a_{n-1}	a_{n-2}	$\cdots$	a_0
α	b_n	b_{n-1}	b_{n-2}		b_0
	$=$	$=$	$=$		$=$
	a_n	$\alpha \cdot b_n + a_{n-1}$	$\alpha \cdot b_{n-1} + a_{n-2}$		$\underbrace{\alpha \cdot b_1 + a_0}_{R-Remainder}$