**processor operations**

**Example - 1:**

**1- Assignment = 1 individual operation**

**2- Arithmetic = 1 individual operation**

**3- Comparison = 1 individual operation**

**4- return = 1 individual operation**

**function func1(x,y) {**

**let z = x + y;**

**return z \* 3**

**}**

**total processor operations = 4**

**Example - 2:**

**Binary Search এর ক্ষেত্রে**

**1- আমরা এখানে ১০ টি এলিমেন্টের একটি সর্টেড অ্যারে নিয়েছি এবং এর থেকে আমরা ৫ খোজার চেস্টা করব । যেহেতু আমাদের অ্যারের মধ্যে ৫ নেই তাহলে আমরা -১ রিটার্ন করে দিব এবং আমরা দেখব যে আমাদের কতগুলো processor operation হয়।**

**2- এখানে n মানে হচ্ছে Total Array Length**

**const sortedArray = [10, 30, 40, 80, 200, 700, 927, 1000, 1410, 1578];**

**function linearSearch(sortedArray, n, value) {**

**for (let i = 0; i < n; i++) {**

**if (sortedArray[i] == value) {**

**console.warn(sortedArray[i]);**

**return i**

**}**

**}**

**return -1**

**};**

**linearSearch(sortedArray, sortedArray.length, 5);**

**1- n = 10 হলে এবং আমরা যদি worst case নিয়ে চলি বা এমন কোন মান নিয়ে যেটা পাওয়া যাবেনা ।**

**2- let i = 0; শুরুতে এখানে একটি অপারেশন হবে = 1 operation**

**3- i < n; এই কন্ডিশন ১১ বার চলবে কারন ১০ বার ভিতরে ঢুকবে ১ বার অপারেশন হবে কিন্তু সামনে আগাবে না = 11 operations**

**4- i++; এই increment ১০ বার হবে = 10 operations**

**5- if (sortedArray[i] == value) এই কন্ডিশন ১০ বার চলবে = 10 operations**

**6- return -1; যেহেতু আমরা worst case নিয়ে চলছি তাহলে কোন মান যদি না পাওয়া যায় লুপ শেষ করে -1 return করবে এখানে = 1 operation**

**summarize the operations**

**1- let i = 0; = 1 operation**

**2- i < n; = 11 operations**

**3- i++; = 10 operations**

**4- if (sortedArray[i] == value) = 10 operations**

**5- return -1; = 1 operation**

**Total Operations = 33 Operations**

**এখন যদি N এর মান বাড়ে তাহলে Operation এর সংখ্যাও বাড়বে।**

**if (n = 10) = 33 operations (3 \* 10) + 3 = 33**

**if (n = 100) = 303 operations (3 \* 100) + 3 = 303**

**if (n = 1000) = 3003 operations (30 \* 100) + 3 = 3003**

**if (n = 10000) = 30003 operations (300 \* 100) + 3 = 30003**

**if (n = 100000) = 300003 operations (3000 \* 100) + 3 = 300003**