Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Метрология стандартизация и сертификация в информационных технологиях (МСиСвИТ)

Отчёт по практическому занятию №1

РАСЧЕТ МЕТРИК ХОЛСТЕДА

Студенты Шукайло П. А.

Антихович. М. В.

Проверил Болтак С. В.

Минск 2024

1 АНАЛИЗИРУЕМАЯ ПРОГРАММА (ЯЗЫК JavaScript)

Листинг 1 ­– Код анализируемой программы

function sumOfNumbers(N) {

a = "ggggggg";

let sum = 0;

for (let i = 1; i <= N; i++) {

sum += i;

}

return sum;

}

function factorial(n) {

if (n === 0 || n === 1) {

return 1;

} else {

return n \* factorial(n - 1);

}

}

function isPrime(num) {

if (num <= 1) {

return false;

}

for (let i = 2; i <= Math.sqrt(num); i++) {

if (num % i === 0) {

return false;

}

}

return true;

}

function gcd(a, b) {

if (!b) {

return a;

}

return gcd(b, a % b);

}

function fibonacci(n) {

let fib = [0, 1];

for (let i = 2; i <= n; i++) {

fib[i] = fib[i - 1] + fib[i - 2];

}

return fib;

}

function isPerfectNumber(n) {

let sum = 1;

for (let i = 2; i <= Math.sqrt(n); i++) {

if (n % i === 0) {

sum += i;

if (i !== n / i) {

sum += n / i;

}

}

}

return sum === n && n !== 1;

}

function squareRoot(x) {

let guess = x / 2;

while (Math.abs(guess \* guess - x) >= 1) {

guess = (guess + x / guess) / 2;

}

return guess;

}

function generatePrimeNumbers(start, end) {

let primes = [];

for (let i = start; i <= end; i++) {

if (isPrime(i)) {

primes.push(i);

}

}

return primes;

}

function bubbleSort(arr) {

let len = arr.length;

for (let i = 0; i < len; i++) {

for (let j = 0; j < len - 1; j++) {

if (arr[j] > arr[j + 1]) {

let temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

return arr;

}

const N = 10;

const resultSum = sumOfNumbers(N);

console.log("Sum of numbers from 1 to", N, ":", resultSum);

const factorialResult = factorial(5);

console.log("Factorial of 5:", factorialResult);

const primeNumbers = generatePrimeNumbers(1, 20);

console.log("Prime numbers from:", primeNumbers);

const sortedArray = bubbleSort([3, 1, 4, 1, 5, 9, 2, 6]);

console.log("Sorted array:", sortedArray);

const fibSequence = fibonacci(10);

console.log("Fibonacci sequence (first numbers):", fibSequence);

const gcdResult = gcd(24, 36);

console.log("Greatest common divisor of numbers:", gcdResult);

const perfectNumberCheck = isPerfectNumber(28);

console.log("Is a perfect number?", perfectNumberCheck);

const squareRootResult = squareRoot(16);

console.log("Square root of:", squareRootResult);

const arraySumResult = arraySum([1, 2, 3, 4, 5]);

console.log("Sum of elements in array:", arraySumResult);

2 РЕЗУЛЬТАТ ВЫПОЛНЕНИЯ ПРОГРАММЫ

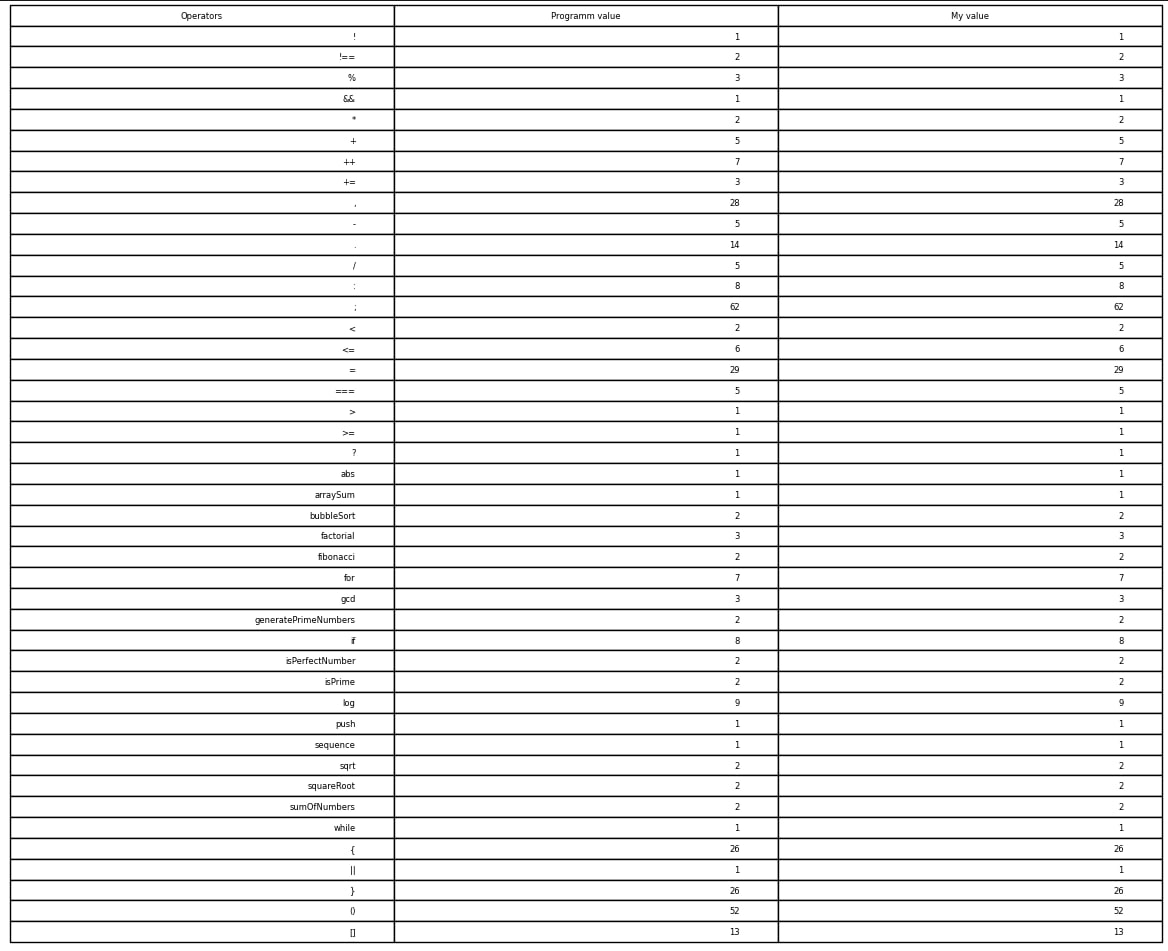
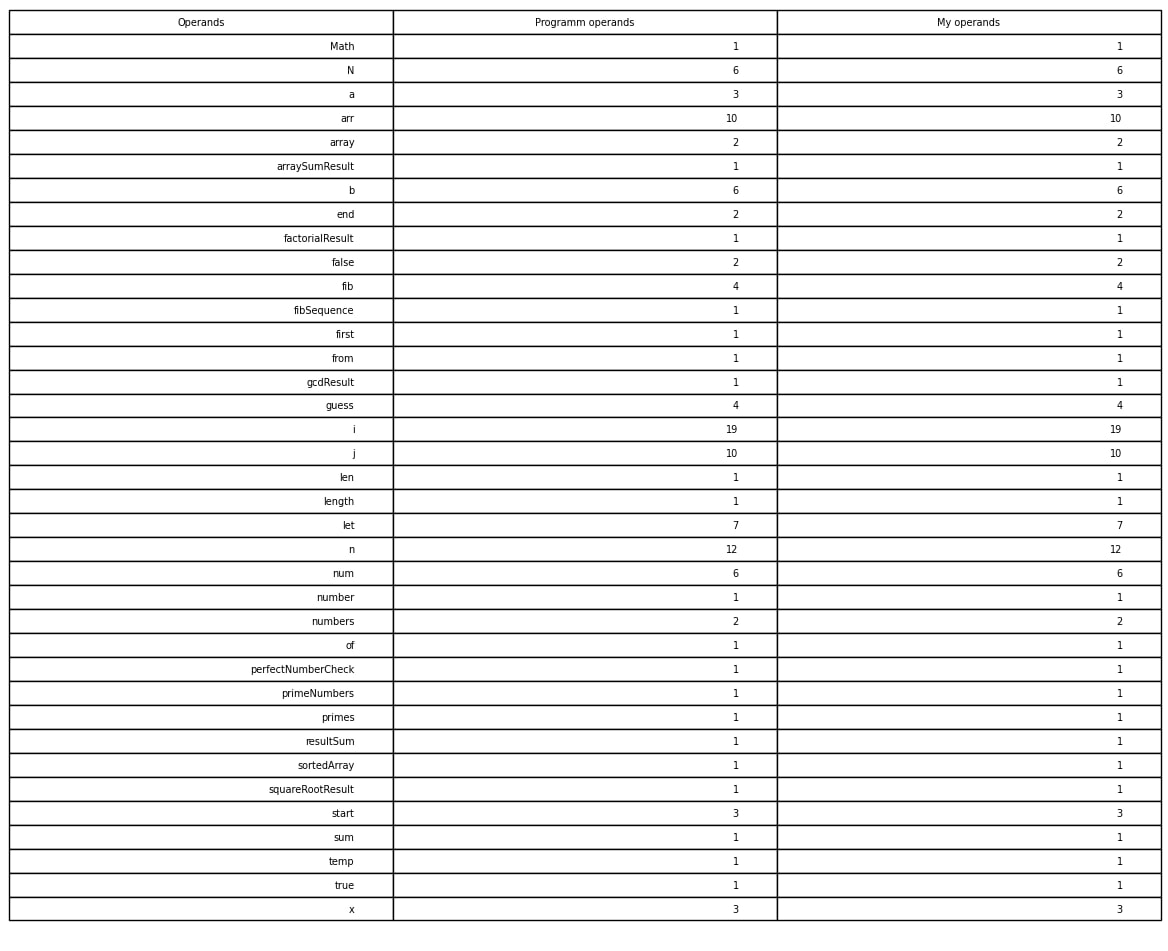


Рисунок 1 – Вывод списка операндов

Рисунок 2 — Вывод списка операторов

3 РЕЗУЛЬТАТ ПОДСЧЕТОВ ВРУЧНУЮ

|  |  |
| --- | --- |
|  | 44 |
|  | 37 |
|  | 334 |
|  | 121 |
|  | 81 = η1 + η2 |
|  | 455 = N1 + N2 |
|  | 455 \* log2(81) = 2885 |