МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования

**«Дальневосточный федеральный университет»**

**ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ**

**Департамент программной инженерии и искусственного интеллекта**

МНОГОПОТОЧНАЯ РЕАЛИЗАЦИЯ ГЕНЕРАТОРА ПСЕВОДОСЛУЧАЙНЫХ ЧИСЕЛ С ПОМОЩЬЮ ЛИНЕЙНОГО КОНГРУЭНТНОГО АЛГОРИТМА DRBG: ЗАВИ

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ**

по дисциплине «Параллельное программирование»

по образовательной программе подготовки бакалавров по направлению 09.03.04 Программная инженерия

Выполнил студент гр. Б9118-09.03.04прогин

\_\_\_\_\_\_\_\_\_\_\_ Скопецкий А.Г.

Проверил: к.т.н., доцент Чусов А. А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

(подпись)

г. Владивосток

2022 г.

# Входные данные

Количество элементов: 100.000.000.

A = 134775813

B = 1

C = 4294967295 (или 2^32 - 1)

Seed = 228

## Таблица результатов параллельных вычислений

|  |  |  |
| --- | --- | --- |
| Результат вычисления | Время вычисления | Ускорение |
| 500,05 | 1,07725 | 1 |
| 500,04 | 0,720922 | 1,494267 |
| 500,05 | 0,507968 | 2,120704 |
| 500,05 | 0,472213 | 2,28128 |
| 500,04 | 0,355925 | 3,026621 |
| 500,05 | 0,323151 | 3,333581 |
| 500,05 | 0,336927 | 3,19728 |
| 500,04 | 0,304243 | 3,540755 |
| 500,05 | 0,263209 | 4,092755 |
| 500,05 | 0,238097 | 4,524417 |
| 500,04 | 0,273198 | 3,943111 |
| 500,05 | 0,215633 | 4,995757 |
| 500,05 | 0,283617 | 3,798256 |
| 500,04 | 0,204856 | 5,258572 |
| 500,05 | 0,257906 | 4,176909 |
| 500,05 | 0,217547 | 4,951804 |
| 500,04 | 0,188815 | 5,70532 |
| 500,05 | 0,192048 | 5,609275 |
| 500,05 | 0,16662 | 6,46531 |
| 500,04 | 0,150241 | 7,170147 |
| 500,05 | 0,169731 | 6,346808 |
| 500,05 | 0,206533 | 5,215873 |
| 500,04 | 0,229964 | 4,684429 |
| 500,05 | 0,182722 | 5,895568 |
| 500,05 | 0,278191 | 3,87234 |
| 500,04 | 0,360875 | 2,985106 |
| 500,05 | 0,273762 | 3,934987 |
| 500,05 | 0,284846 | 3,781868 |
| 500,04 | 0,253792 | 4,244618 |
| 500,05 | 0,307905 | 3,498644 |
| 500,05 | 0,356676 | 3,020248 |
| 500,04 | 0,472401 | 2,280372 |
| 500,05 | 0,299042 | 3,602337 |
| 500,05 | 0,415556 | 2,59231 |
| 500,04 | 0,319926 | 3,367185 |
| 500,05 | 0,51278 | 2,100803 |
| 500,05 | 0,487712 | 2,208783 |
| 500,04 | 0,549972 | 1,958736 |
| 500,05 | 0,540282 | 1,993866 |
| 500,05 | 0,563824 | 1,910614 |
| 500,04 | 0,595178 | 1,809963 |
| 500,05 | 0,551058 | 1,954876 |
| 500,05 | 0,538201 | 2,001576 |
| 500,04 | 0,531859 | 2,025443 |
| 500,05 | 0,547578 | 1,9673 |
| 500,05 | 0,619349 | 1,739326 |
| 500,04 | 0,564384 | 1,908718 |
| 500,05 | 0,71464 | 1,507402 |



## Код алгоритма заполнения массива

#define localOffsetB offsetB \* (vecMulA[t] - 1) / (vecMulA[0] - 1)

#define parSeedI vecMulA[t + 1] \* seedI + localOffsetB

void vecRandParallel(unsigned seedGlobal, vector<vectorType>\* vec, unsigned min, unsigned max) {

//---> Создаем массив множителей A

vector<int> vecMulA(numOfThreads + 1);

vecMulA[0] = mulA;

for (int i = 1; i < vecMulA.size(); i++) {

vecMulA[i] = (vecMulA[i - 1] \* mulA) % mod;

}

//<---

//---> Создаем массив сумм всех элементов массива vec

accInit;

accResInit;

//<---

//---> Процесс каждого из ядер

auto thread\_proc = [=, &vec accThreadProc](unsigned t) {

unsigned seedI = vecMulA[t] \* seedGlobal + localOffsetB;

for (unsigned i = t; i < vec->size(); i += numOfThreads) {

vec->at(i).value = seedI % valBounds;

accInc;

seedI = parSeedI;

}

};

//<---

//---> Создаем массив исполняющих потоков и инициализируем исполнение каждого из потоков

vector<thread> threads;

for (unsigned t = 1; t < numOfThreads; t++)

threads.emplace\_back(thread\_proc, t);

thread\_proc(0);

for (auto& thread : threads)

thread.join();

//<---

accSum;

accOutput;

}