

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
“ВЫСШАЯ ШКОЛА ЭКОНОМИКИ”»

**ДОМАШНЕЕ ЗАДАНИЕ №3**

Работа студента 2 курса бакалавриата группы БПИ-195

по предмету «Архитектура вычислительных систем»

**Выполнил:**

Карабаш Радимир Сейранович,

БПИ195-1

**Преподаватель:**

Доктор технических наук,

Профессор

Легалов А. И

**Москва 2020**

### **Задание:**

Определить множество индексов  $i$ , для которых  $(A[i] - B[i])$  или  $(A[i] + B[i])$  являются простыми числами. Входные данные: массивы целых положительных чисел  $A$  и  $B$ , произвольной длины  $\geq 1000$ . Количество потоков является входным параметром.

### **Решение:**

При запуске программы пользователь вводит приветствующее сообщение с информацией об исполнителе и задании, а затем программа просит ввести пользователя длину двух массивов. После получения информации о длине массивов программа просит пользователя ввести количество потоков для выполнения задачи. Два массива генерируются с помощью генератора псевдослучайных чисел, и логируются в файл input.txt для дальнейшей проверки корректности алгоритма.

Далее в зависимости от количества потоков программа равномерно разделяет данные для обработки двух массивов. Метод, используемый потоками, принимает в себя границу обработки массивов, соответствующие два массива, и ссылку на массив с индексами простых чисел. Далее после отработки потоков список индексов сортируется, логируются в файл output.txt для дальнейшей проверки корректности алгоритма, и показывается пользователю.

Вывод данных осуществляется в консоль.

Для проверки правильности работы алгоритма реализована логирование входных и выходных данных! (файлы input.txt и output.txt).

Входным параметром для количества потоков является целое число в диапазоне от 1 до 10 включительно.

### **Описание методов:**

- **vector<int> generate\_array(int size)** – генерация массива длиной size
- **void is\_prime(int n, bool& flag)** – проверка числа  $n$  на простое

- **bool try\_parse(string n)** – возможно ли привести строку n в целочисленный тип
- **int introduce(int& thr\_count)** – получение входных параметров и вывод информативного сообщения
- **void log(vector<int> a, vector<int> b)** – логирование входных данных
- **void log(vector<int> a)** – логирование выходных данных
- **void get\_indexes(int left\_border, int right\_border, vector<int> a, vector<int> b, vector<int>& indexes)** – получение индексов простых чисел

Текст программы приведен ниже:

```
#include <iostream>
#include<vector>
#include<string>
#include<thread>
#include<mutex>
#include<fstream>
#include<algorithm>

using namespace std;
mutex my_mutex;
//Генерация массива
vector<int> generate_array(int size) {
    vector<int> v1(size);
    for (size_t i = 0; i < v1.size(); i++)
    {
        v1[i] = 1 + rand();
    }
    return v1;
}
// Проверка на простое число
void is_prime(int n, bool& flag) {
    for (int i = 2; i <= sqrt(n); i++) {
        if (n % i == 0) {
            flag = false;
            return;
        }
    }
    flag = true;
}
// Возможно ли записать значение в int
bool try_parse(string n) {
```

```

    try
    {
        int a = stoi(n);
        return true;
    }
    catch (const std::exception&)
    {
        return false;
    }
}
// Ознакомительные сообщения и получение данных от пользователя
int introduce(int& thr_count) {
    std::cout << "Karabash Radimir BSE195\n\n";
    std::cout << "Variant 13. Determine the set of indices i for
which (A [i] - B [i]) or (A [i] + B [i]) are prime
numbers.\nInput data: arrays of positive integers A and B,
arbitrary length must be equal or more than 1000. \nThe number
of streams is an input parameter.\n";
    cout << "Input number of elements in array: ";
    string n;
    cin >> n;
    int size;
    if (try_parse(n))
    {
        size = stoi(n);
        if (size < 1000)
        {
            throw new exception("Size must be more or equal
1000!\n");
        }
    }
    else
    {
        throw new exception("Invalid size of array!\n");
    }
    cout << "Input number of threads: ";
    n = "";
    cin >> n;
    if (try_parse(n))
    {
        thr_count = stoi(n);
        if (thr_count < 1 || thr_count > 10)
        {
            throw new exception("Quantity of threads must be
more or equal than 1 and less than 11!\n");
        }
    }
    else
    {
        throw new exception("Invalid quantity of threads!\n");
    }
    return size;
}

```

```

// Логирование входных и выходных данных
void log(vector<int> a, vector<int> b) {
    fstream file("../input.txt", fstream::out);
    file << "Array A: ";
    for (size_t i = 0; i < a.size(); i++)
    {
        file << to_string(a[i]) + " ";
    }
    file << "\n\nArray B: ";
    for (size_t i = 0; i < b.size(); i++)
    {
        file << to_string(b[i]) + " ";
    }
}

void log(vector<int> a) {
    fstream file("../output.txt", fstream::out);
    file << "Indexes: ";
    for (size_t i = 0; i < a.size(); i++)
    {
        file << to_string(a[i]) + " ";
    }
}

// Получение индексов простых чисел
void get_indexes(int left_border, int right_border, vector<int>
a, vector<int> b, vector<int>& indexes) {
    my_mutex.lock();
    bool left, right;
    for (size_t i = left_border; i < right_border; i++)
    {
        is_prime(abs(a[i] - b[i]), left);
        is_prime(abs(a[i] + b[i]), right);
        if (left || right)
        {
            indexes.push_back(i);
        }
    }
    my_mutex.unlock();
}

int main()
{
    try {
        int thr_count;
        int size = introduce(thr_count);
        vector<int> a = generate_array(size);
        vector<int> b = generate_array(size);
        log(a, b);
        vector<thread> threads;
        vector<int> indexes;
        int det = size / thr_count;
        get_indexes(0, det, a, b, indexes);
        for (size_t i = 1; i < thr_count; i++)
        {
            if (i != thr_count - 1)


```

```

        threads.push_back(thread(get_indexes, det * i,
det * (i + 1), a, b, ref(indexes)));
    else
        threads.push_back(thread(get_indexes, det * i,
det * (i + 1) + size % thr_count, a, b, ref(indexes)));
    }
    for (size_t i = 0; i < threads.size(); i++)
    {
        threads[i].join();
    }
    sort(indexes.begin(), indexes.end());
    log(indexes);
    for (size_t i = 0; i < indexes.size(); i++)
    {
        cout << indexes[i] << endl;
    }
    cout << "Logging of input and output data is implemented
to check the correctness of the algorithm! (files input.txt and
output.txt)";
}
catch (exception* e) {
    cout << "Error occured: " << (*e).what() << endl;
}
}

```

## Результаты разработки:



Консоль отладки Microsoft Visual Studio

Karabash Radimir BSE195

Variant 13. Determine the set of indices  $i$  for which  $(A[i] - B[i])$  or  $(A[i] + B[i])$  are prime numbers.  
Input data: arrays of positive integers  $A$  and  $B$ , arbitrary length must be equal or more than 1000.  
The number of streams is an input parameter.  
Input number of elements in array: 45  
Error occured: Size must be more or equal 1000!

C:\Users\Радимир\source\repos\MultiThread\Debug\MultiThread.exe (процесс 68052) завершил работу с кодом 0.  
Нажмите любую клавишу, чтобы закрыть это окно...

TEST 1

```
Консоль отладки Microsoft Visual Studio
Karabash Radimir BSE195

Variant 13. Determine the set of indices i for which (A [i] - B [i]) or (A [i] + B [i]) are prime numbers.
Input data: arrays of positive integers A and B, arbitrary length must be equal or more than 1000.
The number of streams is an input parameter.
Input number of elements in array: 1000
Input number of threads: 15
Error occured: Quantity of threads must be more or equal than 1 and less than 11!

C:\Users\Радимир\source\repos\MultiThread\Debug\MultiThread.exe (процесс 69836) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно...
```

## TEST 2

```
C:\Users\Радимир\source\repos\MultiThread\Debug\MultiThread.exe
Karabash Radimir BSE195

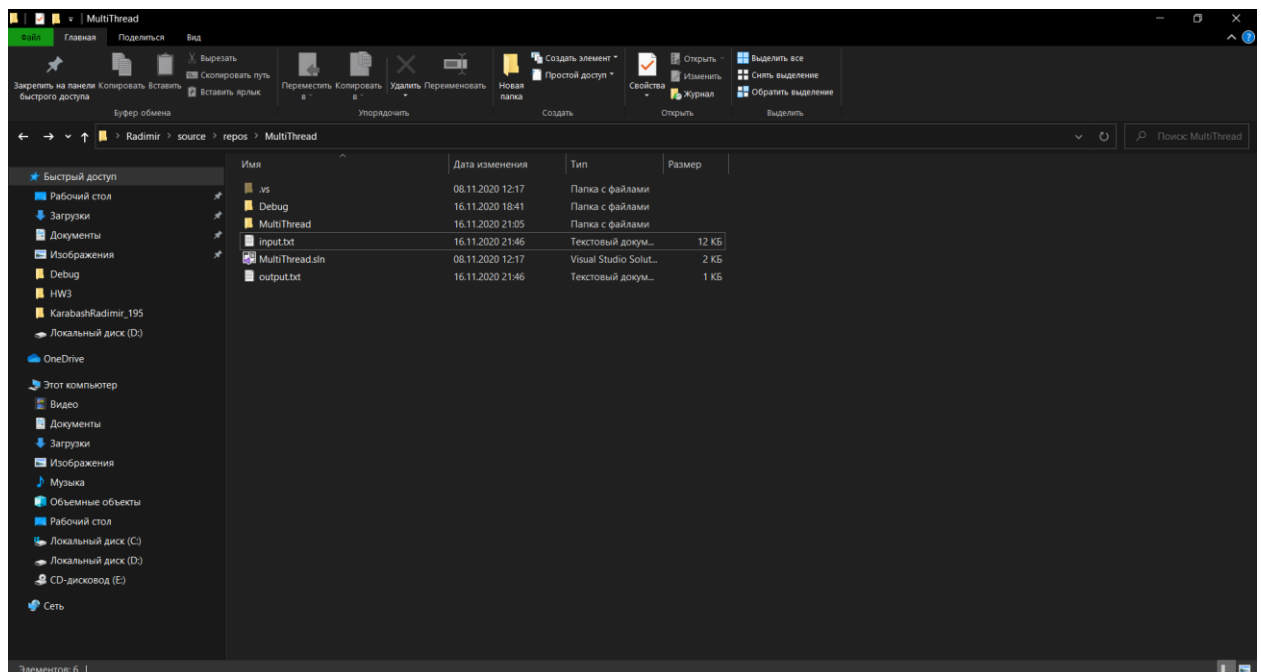
Variant 13. Determine the set of indices i for which (A [i] - B [i]) or (A [i] + B [i]) are prime numbers.
Input data: arrays of positive integers A and B, arbitrary length must be equal or more than 1000.
The number of streams is an input parameter.
Input number of elements in array: 1000
Input number of threads: 5
```

## TEST 3.1

```
Консоль отладки Microsoft Visual Studio

859
862
870
873
880
883
884
886
901
904
906
907
931
945
949
955
959
966
970
975
976
978
991
993
996
Logging of input and output data is implemented to check the correctness of the algorithm! (files input.txt and output.t
xt)
C:\Users\Радимир\source\repos\MultiThread\Debug\MultiThread.exe (процесс 70684) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно...
```

## TEST 3.2



## TEST 3.3



```
input.txt - Блокнот
Файл  Правка  Формат  Вид  Справка
Array A: 42 18468 6335 26501 19170 15725 11479 29359 26963 24465 5706 28146 23282 16828 9962 492 2996 11943 4828 5437 32392 14605 3903 154
85 8282 4735 54 2000 26419 27939 6901 3789 18128 468 3729 14894 24649 22484 17808 2422 14311 6618 22814 9515 14310 7617 18936 17452 20601 5
5536 20451 11174 10467 12045 21660 26293 26440 17254 20025 26155 29511 4746 20650 13187 8314 4475 28023 2169 14019 18788 9906 17959 7392 10
12994 27385 19406 6541 31112 28705 12836 32357 6073 29351 18824 14486 20557 23217 1627 9358 8527 13358 29338 23272 23870 29362 12897 13023
43 7620 20914 7592 6705 31819 9233 751 25206 4976 1540 304 11423 21099 11248 13585 13649 2972 17865 22914 11076 21546 28713 17547 18679 177
1283 13654 30675 2221 5403 6924 3832 19370 3879 20260 19009 22620 23972 30004 21946 9782 26505 12393 32686 25314 6699 5590 12723 5939 19038

Array B: 26668 1529 26680 7865 29422 8406 8827 6817 7517 27727 28667 29088 27682 19965 1341 5687 6022 11663 14722 6065 29310 20416 17903 29
73 6250 14641 29175 18820 944 32612 1942 20290 8420 5566 3806 7586 16217 31451 1616 2610 11065 9167 16894 16075 13510 30301 19696 9574 1559
743 8611 4473 7129 18435 5842 20719 3504 14868 24866 10939 1882 9258 22751 28615 18599 28459 2662 26064 32757 20808 20279 19490 19436 6366
27754 20900 11785 15566 3094 13609 6173 11244 29930 7515 10169 5056 11192 5974 8923 6749 5652 10987 2145 16447 31578 26518 14630 29917 5875
11970 28979 21618 4016 16627 3685 9169 17907 25929 12098 28119 24391 15200 11786 14487 19200 12421 20711 18272 15814 27416 6086 319 3581 13
043 26254 20084 11421 15815 32719 12245 31064 7230 20653 18865 4770 30471 15006 21048 1595 21488 24327 3277 21324 6541 7680 23991 32589 247
```

## TEST 3.4

```
output.txt - Блокнот
Файл  Правка  Формат  Вид  Справка
Indexes: 1 10 13 30 32 35 43 46 52 53 57 58 62 63 65 71 81 83 88 91 95 98 102 109 126 130 131 144 150 157 158 166 185 197 204 206 217 221 2
```

## TEST 3.5

```
C:\Users\Радимир\source\repos\MultiThread\Debug\MultiThread.exe
Karabash Radimir BSE195

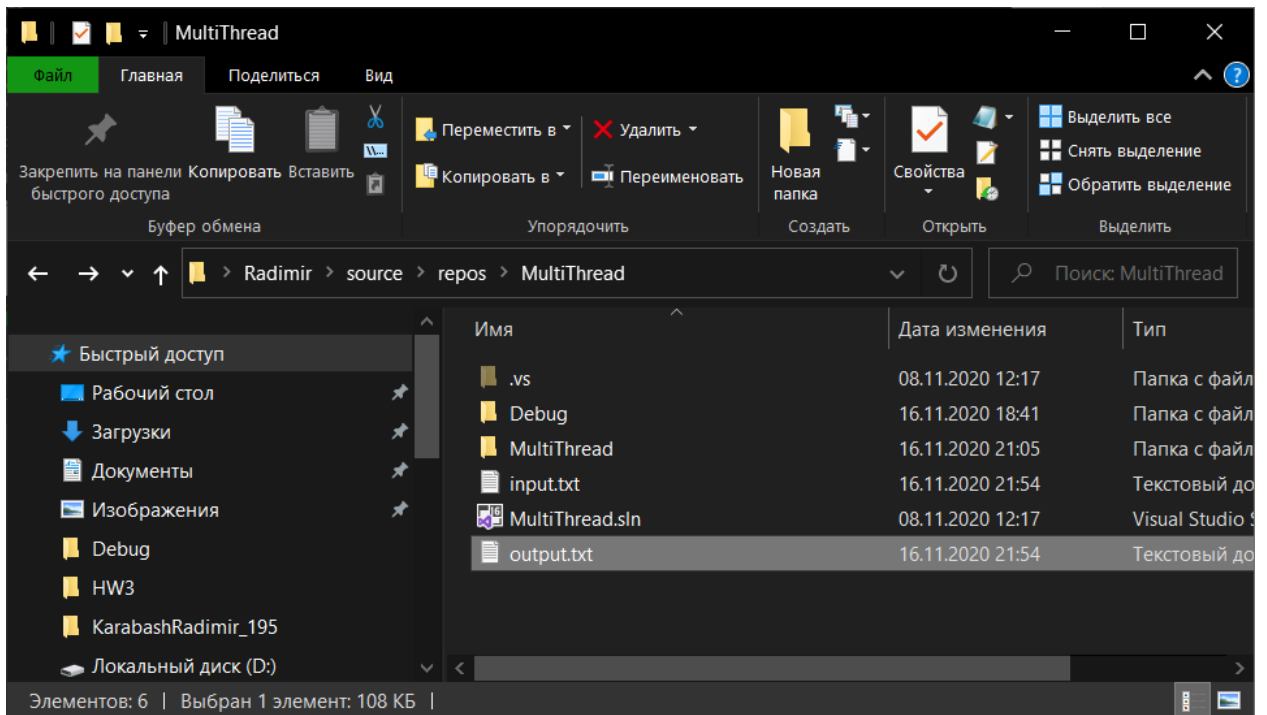
Variant 13. Determine the set of indices i for which (A [i] - B [i]) or (A [i] + B [i]) are prime numbers.
Input data: arrays of positive integers A and B, arbitrary length must be equal or more than 1000.
The number of streams is an input parameter.
Input number of elements in array: 100000
Input number of threads: 10_
```

## TEST 4.1

```
Консоль отладки Microsoft Visual Studio

99870
99881
99886
99894
99901
99916
99917
99918
99919
99925
99926
99932
99934
99942
99944
99950
99962
99969
99970
99987
99988
99990
99997
99998
99999
Logging of input and output data is implemented to check the correctness of the algorithm! (files input.txt and output.txt)
C:\Users\Радимир\source\repos\MultiThread\Debug\MultiThread.exe (процесс 71208) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно...
```

## TEST 4.2



## TEST 4.3

[illegible]

## TEST 4.4

```
File Edit Format View Help
Indexes: 0 1 8 15 16 19 31 37 40 45 47 52 53 70 83 99 104 106 109 111 112 115 119 132 133 141 154 155 156 195 198 204 228 230 237 239 242 2
1387 1393 1395 1396 1401 1403 1405 1408 1415 1416 1420 1421 1428 1431 1435 1444 1445 1446 1447 1448 1450 1451 1461 1473 1478 1496 1499 1502
2550 2562 2565 2566 2568 2574 2581 2583 2587 2600 2604 2611 2615 2617 2618 2620 2621 2640 2646 2649 2654 2656 2657 2658 2666 2672 2677 2686
3613 3617 3620 3622 3625 3640 3641 3642 3643 3644 3646 3651 3652 3655 3660 3671 3680 3685 3687 3689 3692 3715 3724 3730 3731 3738 3741 3749
4617 4622 4628 4631 4634 4635 4649 4652 4657 4659 4663 4668 4670 4672 4675 4682 4686 4689 4700 4704 4710 4723 4726 4728 4741 4754 4755 4758
5740 5745 5748 5758 5771 5773 5780 5787 5789 5790 5792 5805 5806 5808 5816 5817 5824 5825 5828 5829 5834 5842 5848 5854 5857 5860 5863
6738 6739 6742 6744 6750 6764 6766 6775 6776 6780 6785 6786 6795 6798 6800 6802 6806 6816 6820 6823 6829 6832 6844 6846 6848 6849 6850 6858 6866
7862 7870 7871 7874 7877 7879 7886 7901 7903 7905 7916 7931 7938 7946 7949 7953 7955 7956 7957 7959 7960 7963 7967 7968 7971 7984 7987 7991
8927 8936 8937 8942 8945 8948 8949 8950 8956 8957 8962 8986 8989 8991 8993 8995 9010 9014 9017 9021 9026 9038 9040 9042 9046 9053 9054 9057 9059
4 10022 10025 10027 10032 10037 10040 10043 10058 10063 10070 10076 10078 10081 10084 10092 10094 10098 10101 10108 10113 10115 10117 10122
939 10942 10946 10948 10960 10962 10966 10967 10970 10983 10987 10994 10995 11001 11004 11005 11010 11013 11018 11024 11029 11030 11031 111
11806 11814 11817 11819 11827 11832 11838 11855 11859 11863 11867 11868 11877 11880 11882 11901 11912 11915 11918 11921 11923 11940 11948 1
0 12703 12708 12711 12712 12715 12719 12730 12740 12759 12765 12770 12771 12778 12779 12780 12784 12787 12788 12791 12794 12798 12809 12813
645 13647 13656 13663 13667 13670 13674 13679 13684 13694 13702 13704 13709 13711 13713 13714 13724 13727 13738 13739 13740 13744 13747 137
14532 14542 14555 14565 14567 14579 14584 14590 14593 14595 14597 14604 14611 14618 14621 14625 14629 14630 14633 14641 14655 14659 14662 1
2 15417 15464 15467 15471 15473 15477 15491 15493 15498 15511 15513 15514 15523 15527 15529 15540 15542 15549 15560 15565 15566 15572
15573 15574 15575 15576 15577 15578 15579 15580 15581 15582 15583 15584 15585 15586 15587 15588 15589 15590 15591 15592 15593 15594 15595 15596 15597 15598 15599 15600 15601 15602 15603 15604 15605 15606 15607 15608 15609 15610 15611 15612 15613 15614 15615 15616 15617 15618 15619 15620 15621 15622 15623 15624 15625 15626 15627 15628 15629 15630 15631 15632 15633 15634 15635 15636 15637 15638 15639 15640 15641 15642 15643 15644 15645 15646 15647 15648 15649 15650 15651 15652 15653 15654 15655 15656 15657 15658 15659 15660 15661 15662 15663 15664 15665 15666 15667 15668 15669 15670 15671 15672 15673 15674 15675 15676 15677 15678 15679 15680 15681 15682 15683 15684 15685 15686 15687 15688 15689 15690 15691 15692 15693 15694 15695 15696 15697 15698 15699 15700 15701 15702 15703 15704 15705 15706 15707 15708 15709 15710 15711 15712 15713 15714 15715 15716 15717 15718 15719 15720 15721 15722 15723 15724 15725 15726 15727 15728 15729 15730 15731 15732 15733 15734 15735 15736 15737 15738 15739 15740 15741 15742 15743 15744 15745 15746 15747 15748 15749 15750 15751 15752 15753 15754 15755 15756 15757 15758 15759 15760 15761 15762 15763 15764 15765 15766 15767 15768 15769 15770 15771 15772 15773 15774 15775 15776 15777 15778 15779 15780 15781 15782 15783 15784 15785 15786 15787 15788 15789 15790 15791 15792 15793 15794 15795 15796 15797 15798 15799 15800 15801 15802 15803 15804 15805 15806 15807 15808 15809 15810 15811 15812 15813 15814 15815 15816 15817 15818 15819 15820 15821 15822 15823 15824 15825 15826 15827 15828 15829 15830 15831 15832 15833 15834 15835 15836 15837 15838 15839 15840 15841 15842 15843 15844 15845 15846 15847 15848 15849 15850 15851 15852 15853 15854 15855 15856 15857 15858 15859 15860 15861 15862 15863 15864 15865 15866 15867 15868 15869 15870 15871 15872 15873 15874 15875 15876 15
```

## TEST 4.5