Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Кафедра электронных вычислительных машин

Отчёт

по лабораторной работе № 3

по дисциплине КиДСВТ

Выполнил ст. гр. 950502 Раговский И.Д.

Проверил: Татур М.М.

Минск 2023

**1. Выполнить тестирование оперативной памяти с использованием встроенной в ОС Windows программы, в различных режимах.**

Запуск программы осуществляется нажатием клавиш «Win» + «R», затем, дать согласие на перезагрузку компьютера. Далее необходимо ввести команду **mdsched.exe в появившемся окне (см. Рисунок 1.1).**

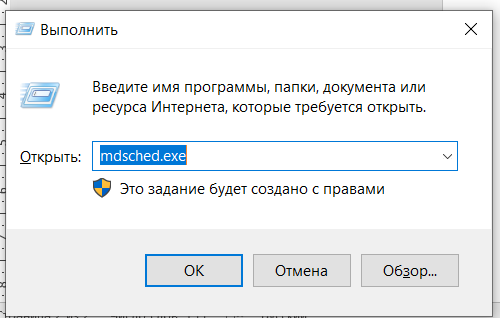


Рисунок 1.1 – Запуск программы для проверки памяти

Далее нажимаем «ОК», откроется окно «Средство проверки памяти Windows» (см. Рисунок 1.2).

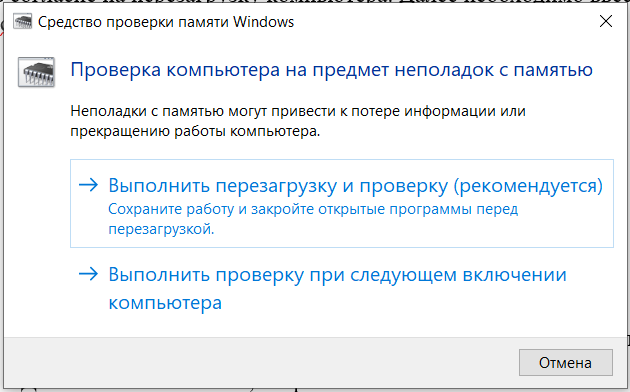


Рисунок 1.2 – Окно «Средство проверки памяти Windows»

Далее выбираем пункт «Выполнить перезагрузку и проверку (рекомендуется)». Компьютер выключится и начнет проверку оперативной памяти (см. Рисунок 1.3), результаты которой можно будет посмотреть после прохождения проверки и входа в операционную систему либо кликнув по оповещению, либо перейдя в журнал операций.

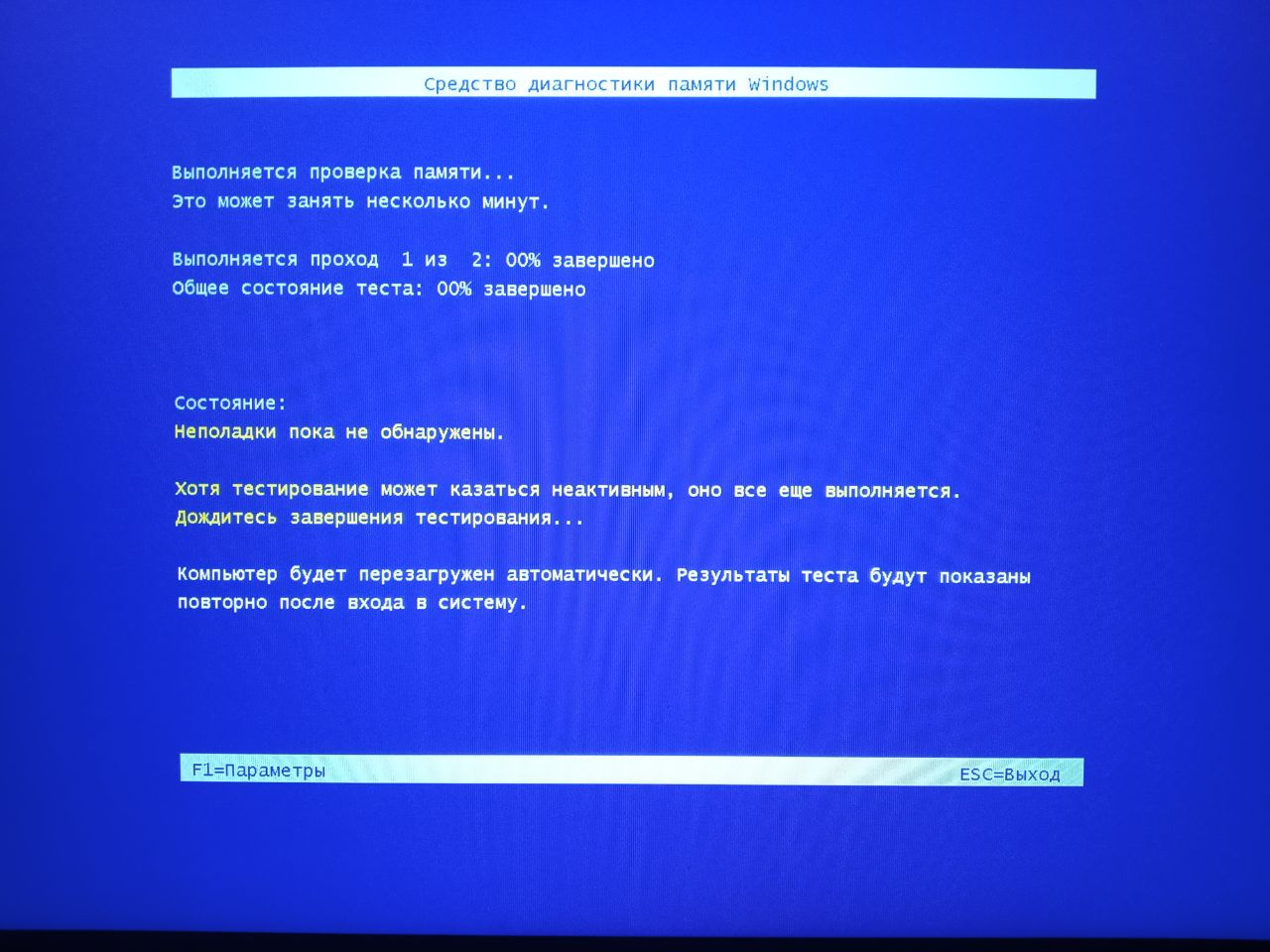


Рисунок 1.3 – Проверка оперативной памяти программным средством OC Windows

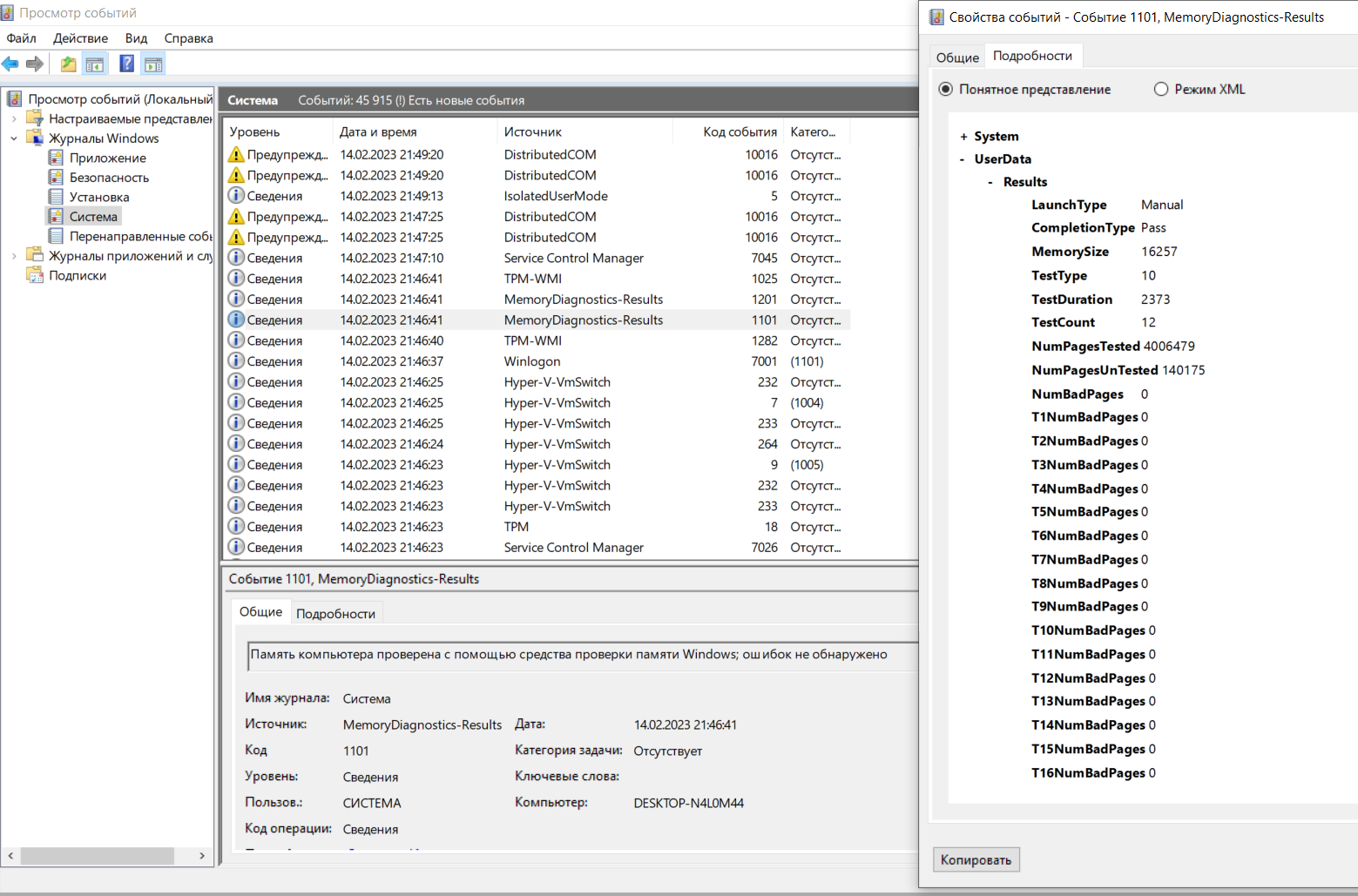


Рисунок 1.4 – Событие с кодом 1101 для теста 1

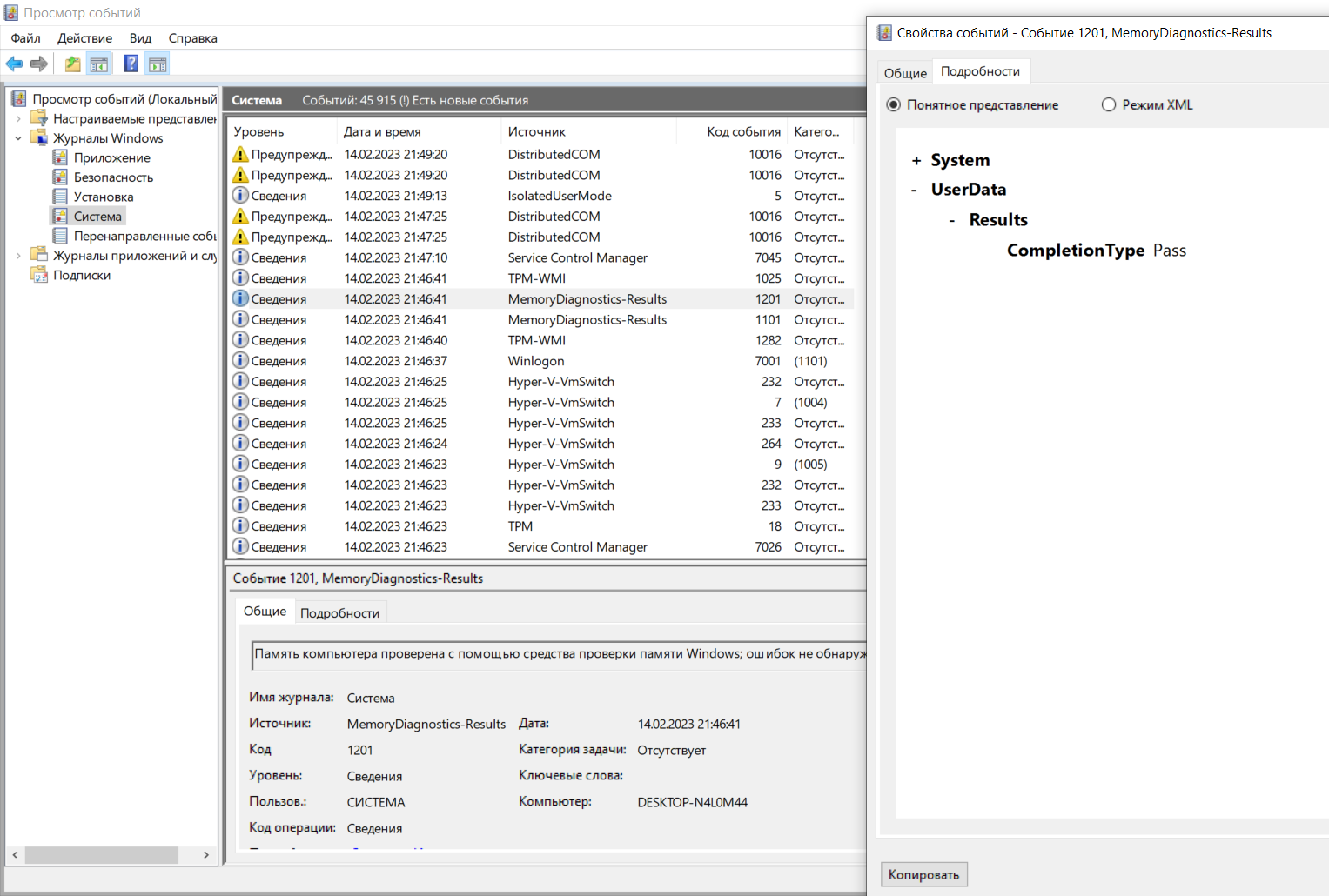


Рисунок 1.5 – Событие с кодом 1201 для теста 1

В результате проверки тестом 1 в журнале системы появились события с кодами 1101 и 1201 соответственно (см. Рисунок 1.4 и 1.5). Первый тест прошел нормально и не выявил никаких ошибок в оперативной памяти. В результате тест определил размер оперативной памяти: 16257 Мб; время теста указано в непонятной величине, но по нему можно определить длительность того или иного теста: 2373; кол-во тестов – 12; кол-во протестированных страниц – 4006479; кол-во не протестированных страниц – 140175.

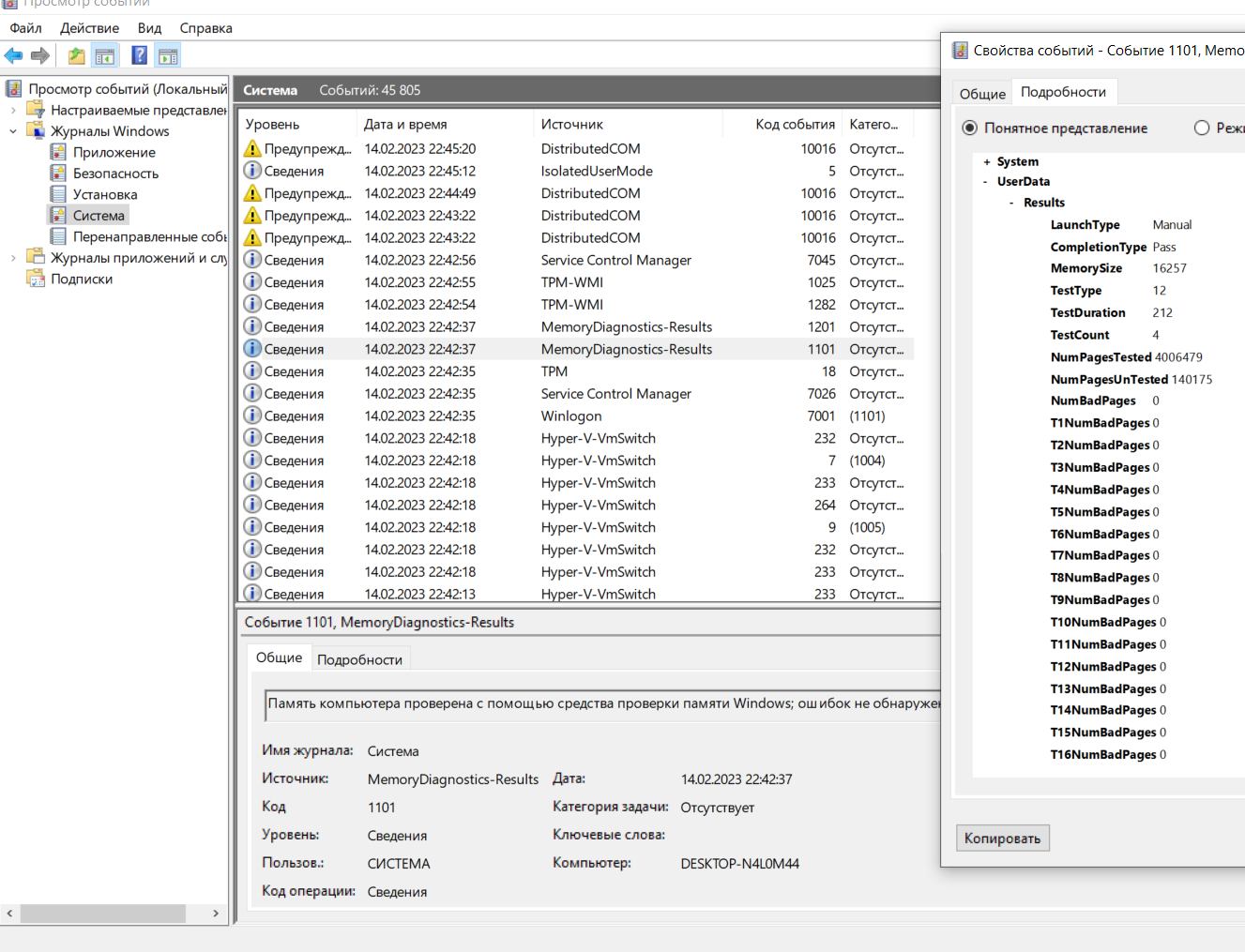


Рисунок 1.6 – Событие с кодом 1101 для теста 2

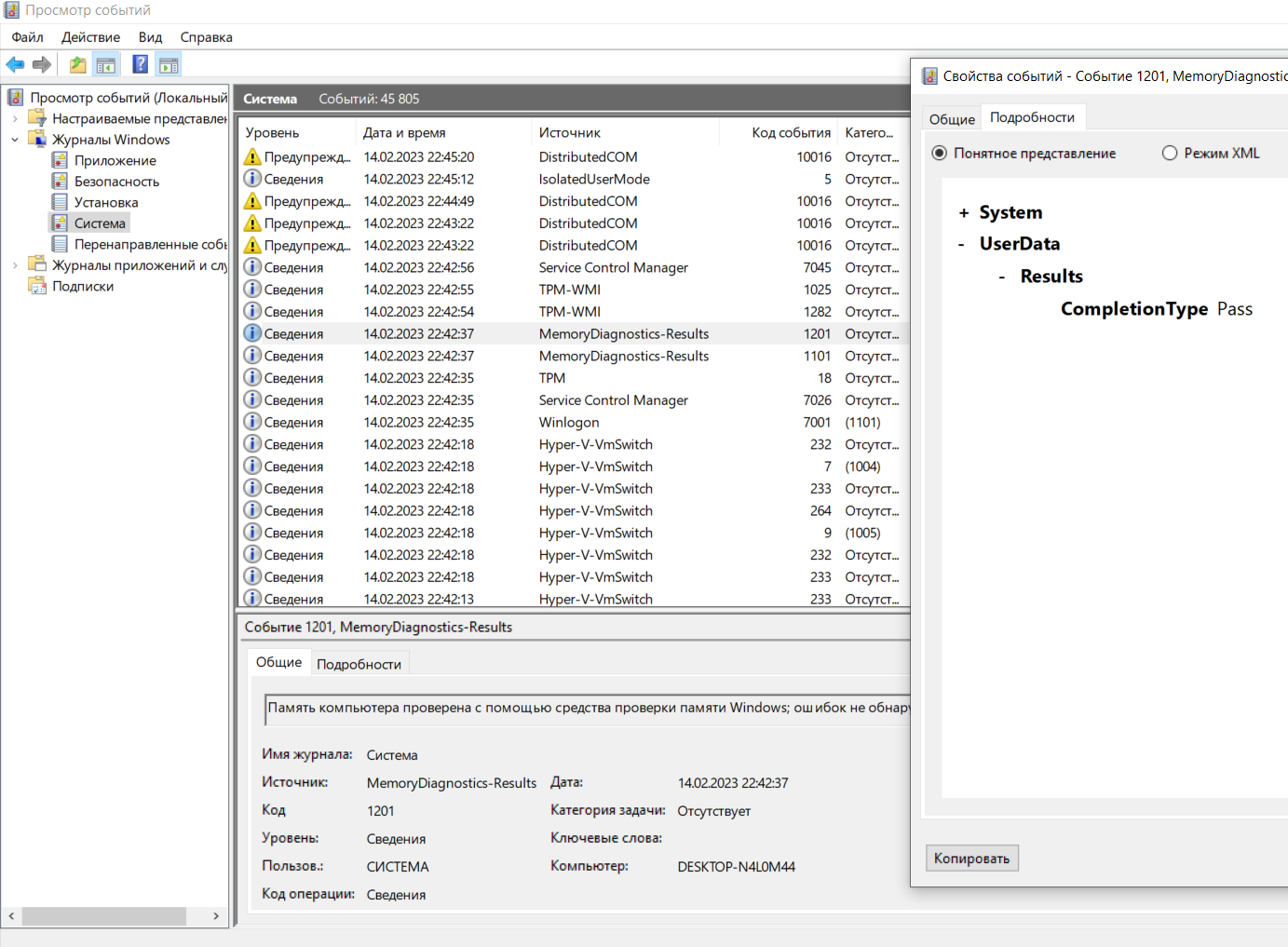


Рисунок 1.7 – Событие с кодом 1201 для теста 2

В результате проверки тестом 2 в журнале системы появились события с кодами 1101 и 1201 соответственно (см. Рисунок 1.6 и 1.7). Второй тест прошел нормально и не выявил никаких ошибок в оперативной памяти. В результате тест определил размер оперативной памяти: 16257 Мб; время теста указано в непонятной величине, но по нему можно определить длительность того или иного теста: 212; кол-во тестов – 4; кол-во протестированных страниц – 4006479; кол-во не протестированных страниц – 140175.

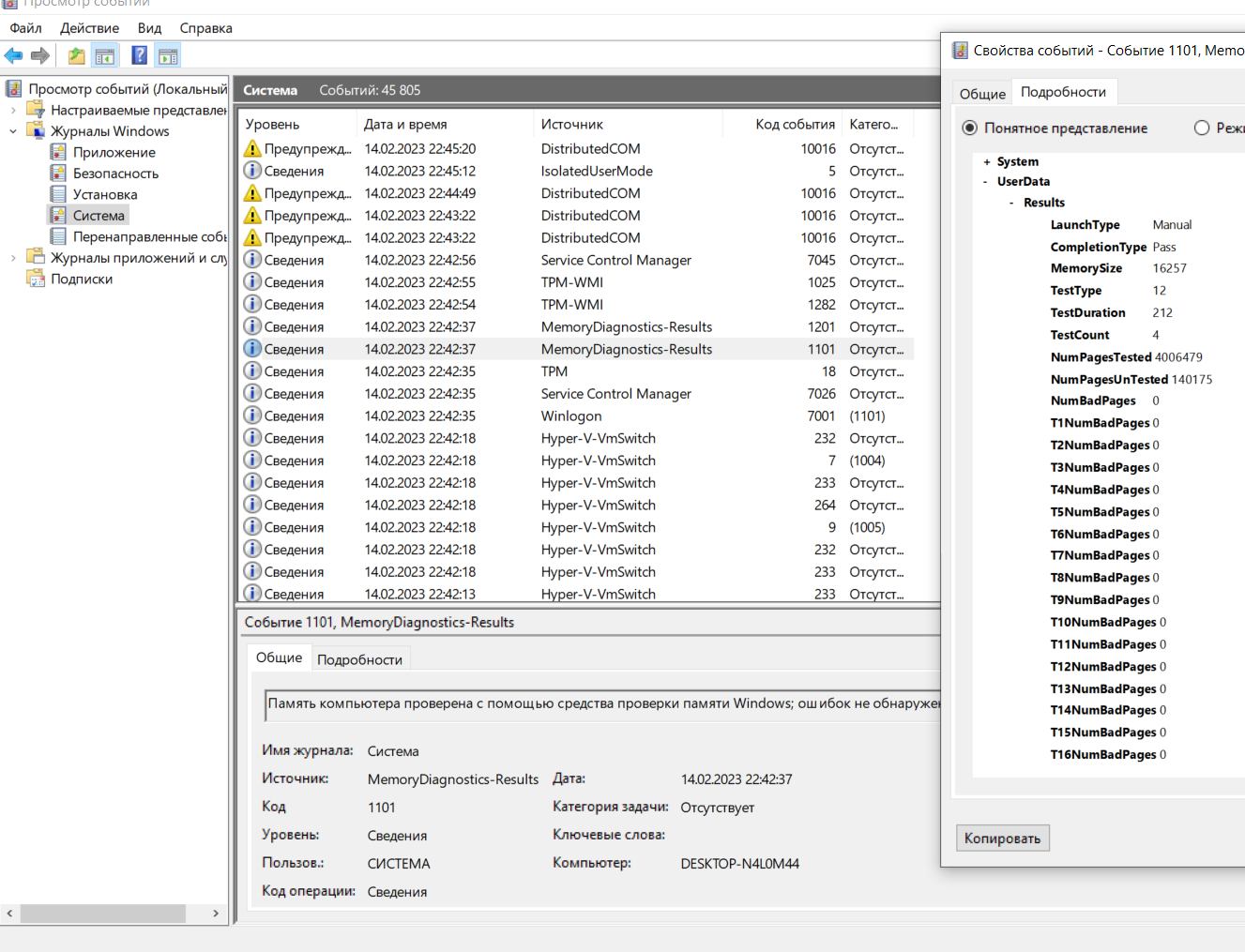


Рисунок 1.6 – Событие с кодом 1101 для теста 2

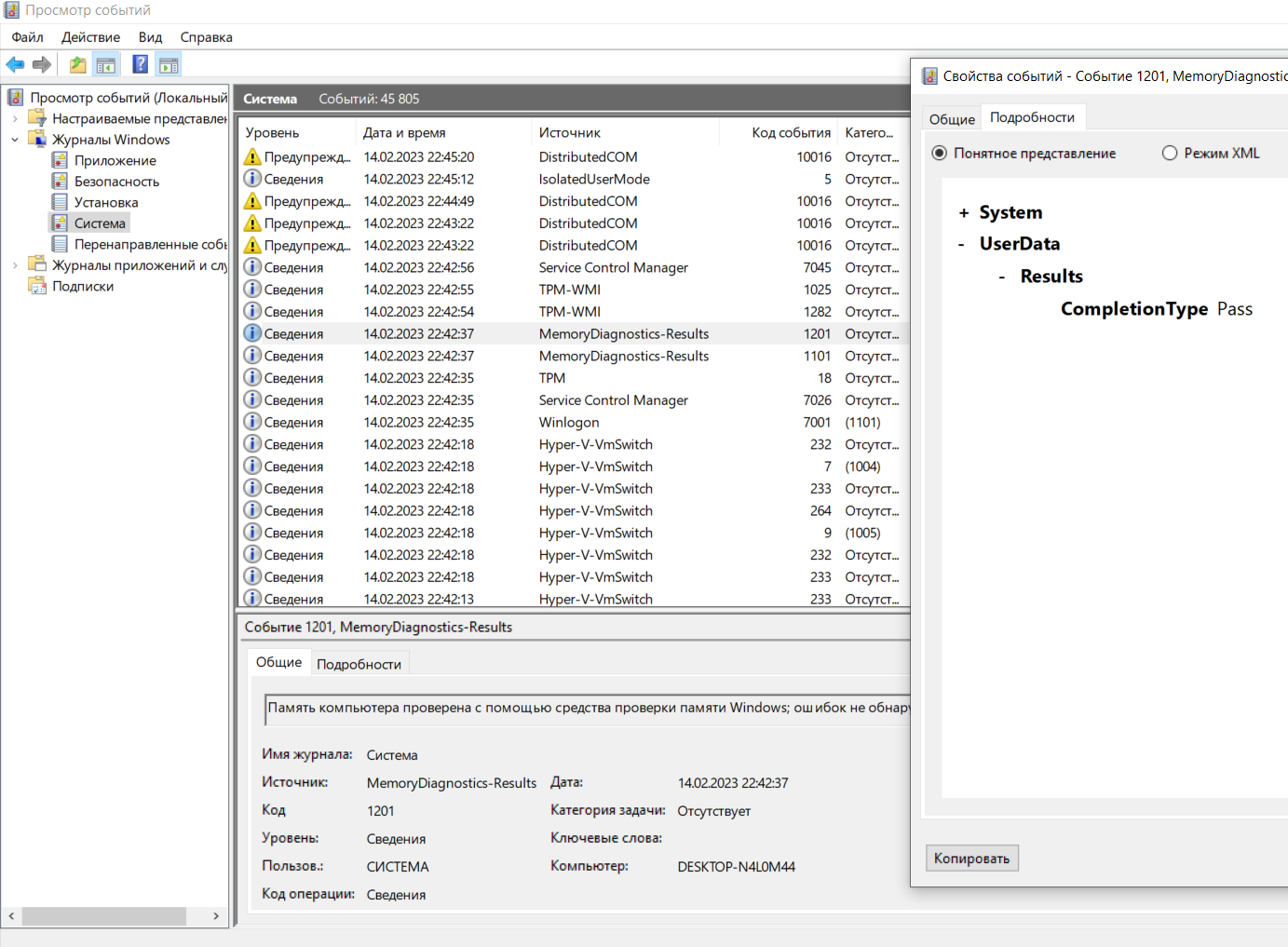


Рисунок 1.7 – Событие с кодом 1201 для теста 2

Тестом 3 проверить не удалось. Запускался тест 2 раза, первый раз 14.02.2023 примерно в 23:00, второй – утром 15.02.2023 в 9:17. Тест был прерван в первый раз и перезапущен. Точно так же он был прерван во второй раз в 18:11. Прошло более 9 часов и никаких продвижений не было. Средство диагностики зависло на отметке в 21%. По длительности теста можно предположить, что он просто завис в определенный момент. Это может произойти по причине проблемы с оперативной памятью, проблемы с ПО. Хочу заметить, что в ноутбуке стоит оперативная память с разными таймингами. Но она определяется и работают обе плашки памяти. В журнале системы появилось событие с кодами 1103 (см. Рисунок 1.7). В результате тест определил размер оперативной памяти: 16257 Мб; время теста: 1149; кол-во тестов – 6; кол-во протестированных страниц – 4006487; кол-во не протестированных страниц – 140177.

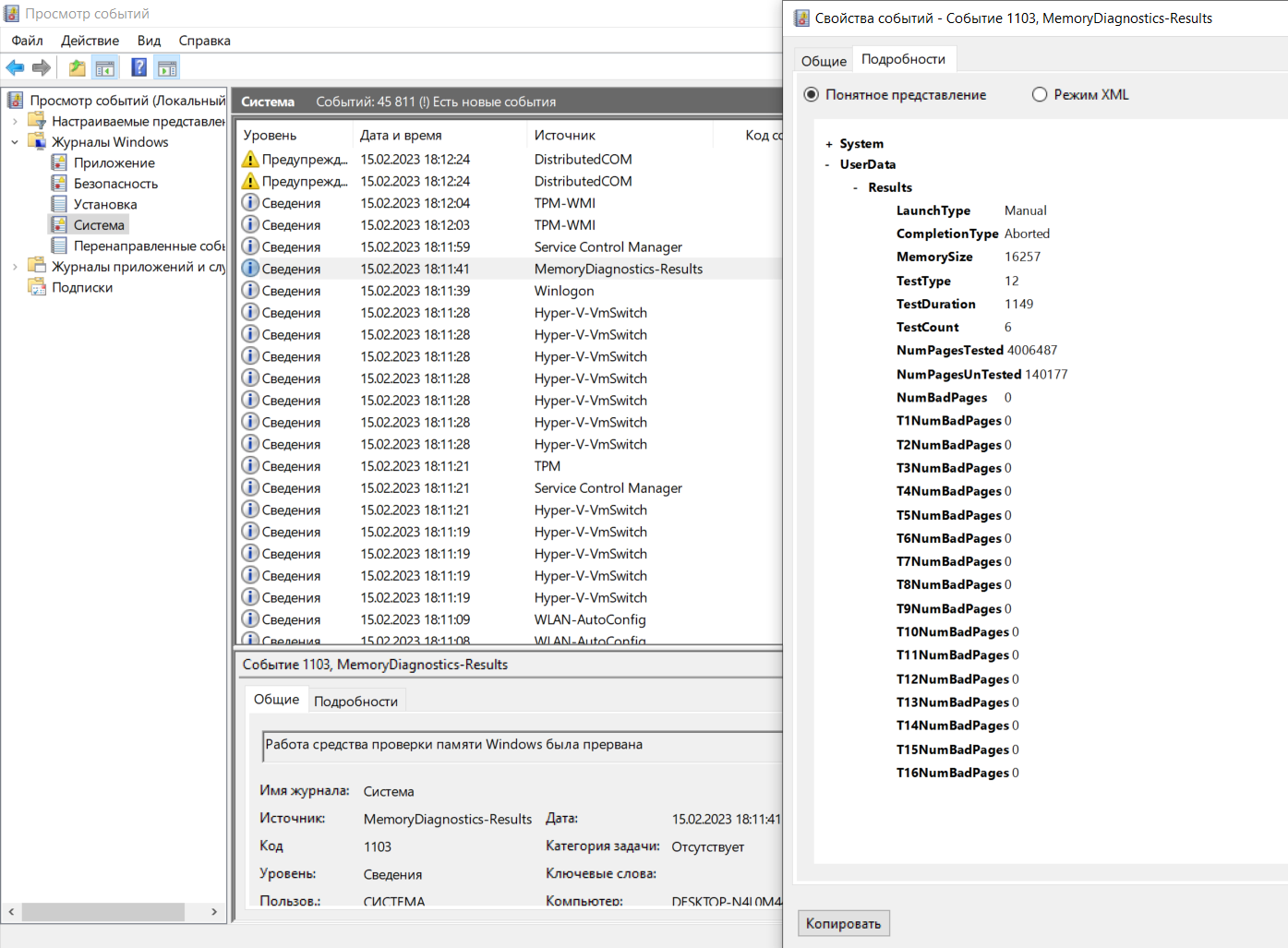


Рисунок 1.8 – Событие с кодом 1301 для теста 3

**2. Выписать названия алгоритмов тестирования для каждого режима, оценить время выполнения тестирования в каждом из режимов.**

Тест 1 начался в 21:05 закончился в 21:46. Набор тестов – Обычный: все базовые тесты, плюс LRAND, Stride6 (с включенным кэшем), CHCKR3, WMATS+ и WINVC. Время теста около 41 минуты. В условных единицах время – 2373.

Тест 2 начался в 22:36 закончился в 22:42. Набор тестов – Базовый: MATS+, INVC и SCHCKR (с включенным кэшем). Время теста около 6 минут. В условных единицах время - 212

Тест 3 был отменен из-за слишком большого времени тестирования. Набор тестов – Широкий: Все стандартные тесты, плюс MATS+ (с отключенным кэшем), Stride38, WStride-6, CHCKR4, WCHCKR3, ERAND, Stride6 (с отключенным кэшем) и CHCKR8. Время, через которое был отменен тест – 9 часов.

**3. Найти в литературе, сделать формальное описание реализуемых алгоритмов, оценить объем выполняемых операций запись/чтение для каждого из них. Сделать вывод, на какие модели неисправностей ориентирован каждый из алгоритмов тестирования.**

Маршевый тест MATS (modified algorithmic test sequence − MATS) имеет следующую запись: {⇑⇓(w0); ⇑(r0,w1); ⇓(r1)}, согласно которой данный тест состоит из трех маршевых элементов и имеет сложность 4N. Первый маршевый элемент ⇑⇓(w0) называется фазой инициализации (initialization phase), применяемой для записи начального состояния (background) запоминающего устройства. Эта фаза, как правило, выполняется при изменении адресов от младшего адреса к старшему адресу или, наоборот, путем записи во 61 все ячейки нулевого значения. Вторая фаза теста MATS ⇑(r0, w1) определяет возрастающий порядок адресов и для каждого ЗЭ состоит из операции чтения (r0), когда ожидаемое значение содержимого ЗЭ 0, и записи в данную ячейку 1 (w1). В третьей фазе (⇓(r1)) содержимое ЗУ считывается последовательно, при этом порядок изменения адресов обратный по отношению к предыдущей фазе, а ожидаемое считываемое значение равно 1. March C- является лучшем маршевым тестом, способным распознать SAF, AF, TF, CF ошибки и имеет сложность 10N.

MATS+ имеет сложность 5N. Используется для решения AF (неисправности дешифратора адреса (address decoder faults – AF), или адресные неисправности) и SAF (Константные неисправности (stuck-at faults – SAF)).

INVC – …

Stride6 – Этот алгоритм оценивает только простые комбинации присутствия и отсутствия аномалий с соблюдением требования высокой скорости обработки и низкой требуемая пропускной способности данных, которые имитируют аппаратные ограничения встроенных устройств. В слове «Stride» каждая буква обозначает тип проблемы: S – Spoffing, T – Tampering, R – Repudiation, I – Information disclosure, D – Denial of service, E – Evelation of privilege [1]. Сложность теста log2N.

CHCKR – **Checkerboard Algorithm. Алгоритм делит ячейки на две чередующиеся группы, так что каждая соседняя ячейка находится в другой группе. Шаблон шахматной доски в основном используется для активации отказов, возникающих в результате утечек, коротких замыканий между ячейками и SAF [2]. W в названии, предположительно, означает, что тест только на запись. Цифра – окно, или размер «шахматной доски». Сложность теста 4N.**

RAND – это алгоритм, который использует некоторую степень случайности как часть своей логики или процедуры. Алгоритм обычно использует равномерно случайные биты в качестве вспомогательных входных данных для управления своим поведением в надежде на достижение хорошей производительности в «среднем случае» среди всех возможных вариантов случайного выбора, определяемых случайными битами; таким образом, либо время работы, либо результат (или и то, и другое) являются случайными величинами.

[1] <https://www.omron.com/global/en/assets/file/technology/omrontechnics/vol52>

/OMT\_Vol52\_013EN.pdf

[2] <https://www.design-reuse.com/articles/45915/memory-testing-self-repair-mec>

hanism.html#:~:text=Checkerboard%20Algorithm&text=The%20algorithm%20divides%20the%20cells,shorts%20between%20cells%2C%20and%20SAF.

**4. Скачать одну из программ тестирования компьютера, изучить ее интерфейс, применить для тестирования компьютера. Привести описание программы и порядок тестирования.**

После загрузки архива необходимо создать на флеш-накопителе образ программы (см. Рисунок 4.1)

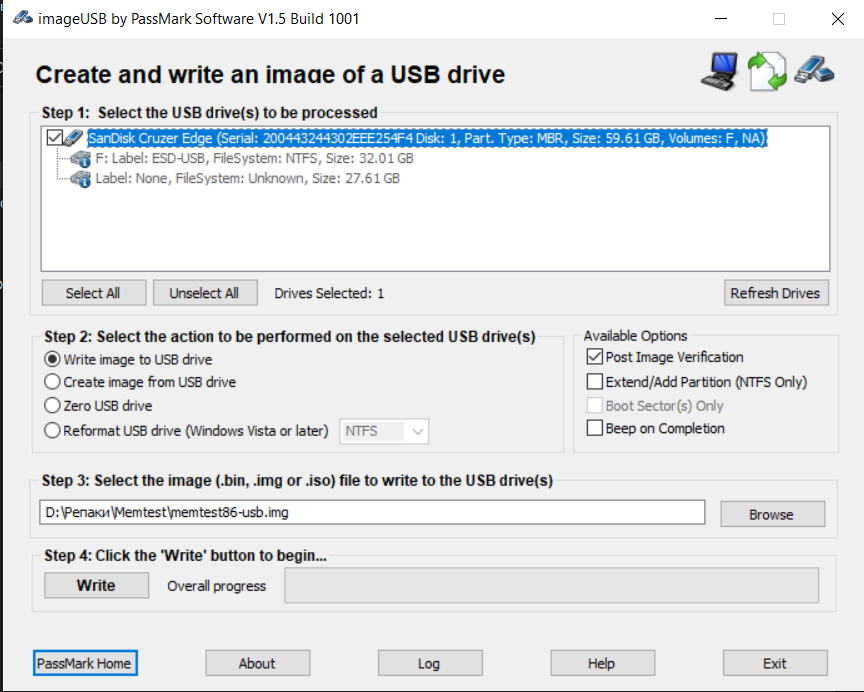


Рисунок 4.1 – Создание образа на флеш-накопителе.

На рисунке 4.1 видно, что создание флеш-накопителя разделено на 4 шага:

1. Выбор диска/флешки на который установится образ

2. Выбор действия: Записать программу, Создать программу, Очистить накопитель, Изменить формат накопителя. Так же есть возможность включить дополнительные опции.

3. Выбор образа.

4. Загрузка.

Для начала загрузки необходимо поочередно убедиться, что все шаги заполнены так, как вам требуется и нажать кнопку «Write».

К сожалению, memtest86 не запустился. Проблема, скорее всего, в совместимости материнской платы.

Так как эта программа не подошла попробую другую, выберу OCCT. Её можно скачать с сайта разработчика (см. Рисунок 4.2).

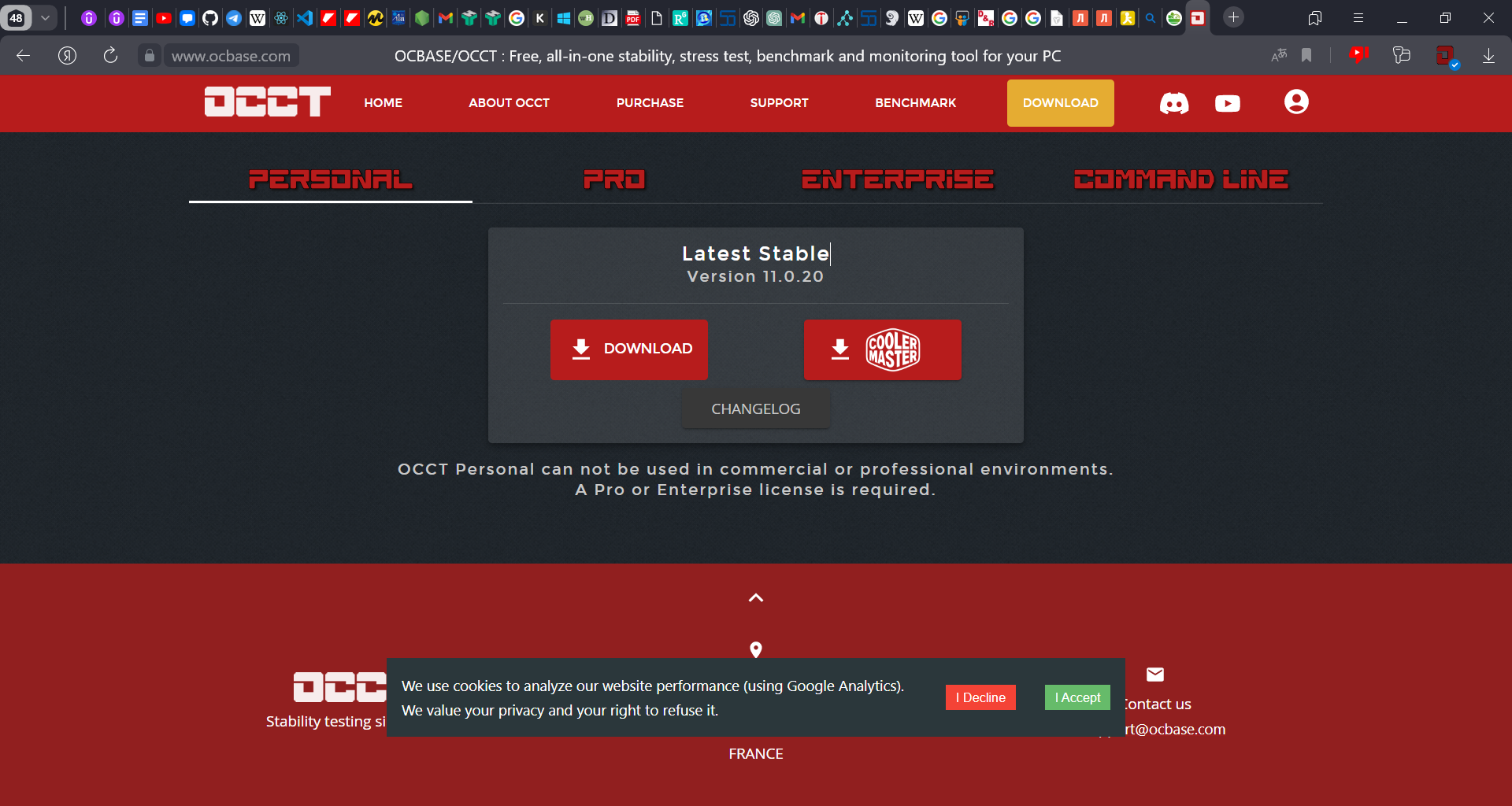


Рисунок 4.2 – OCCT

Есть 4 версии ПО: Personal, PRO, Enterprise, Command Line. Предпочтительной для меня является версия Personal (см. Рисунок 4.3), потому что PRO платная, Enterprise для крупных предприятий, Commandline будет с не интуитивным командным интерфейсом, что мне не нравится. К тому же, только Personal можно получить без предварительной аутентификации.

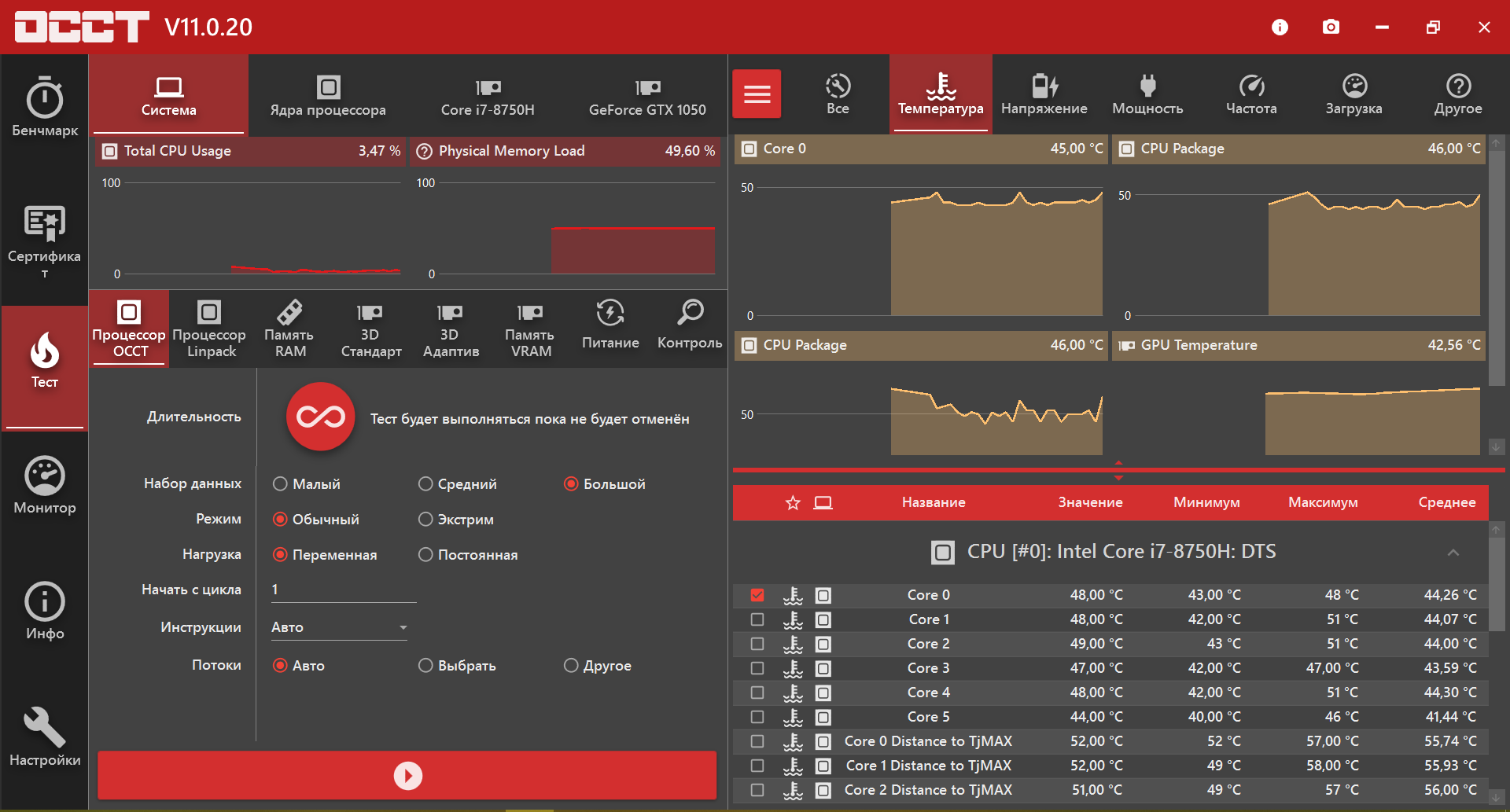
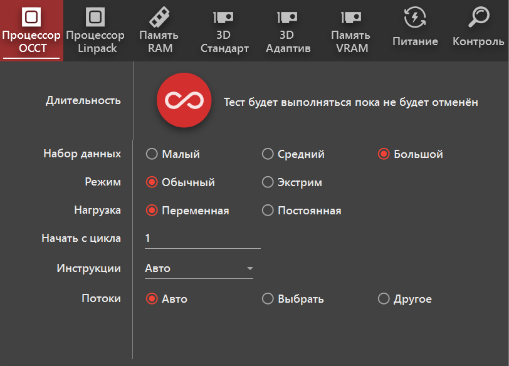
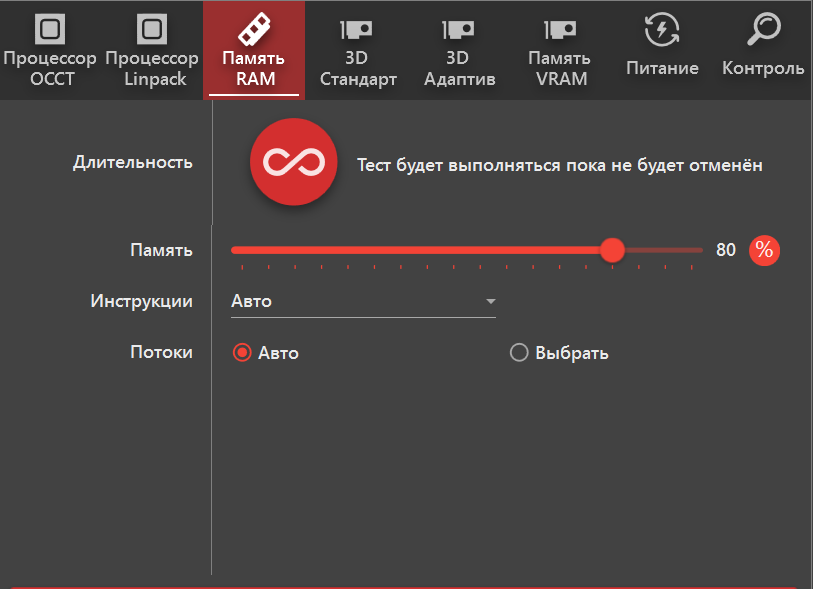
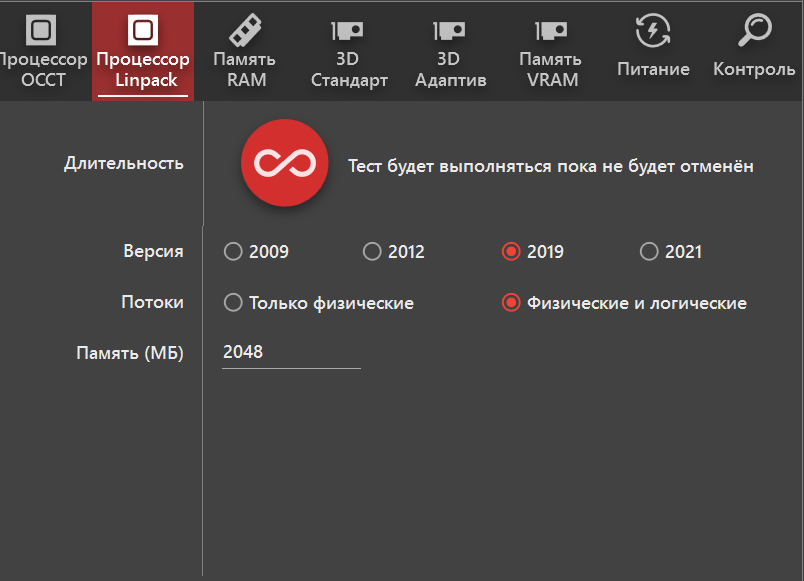


Рисунок 4.2 – Интерфейс OCCT Personal. Вкладка «Тест»

На рисунке 4.2 открыта вкладка «Тест». Справа в ней дублирована вкладка «Монитор». Слева вверху расположены отдельные «важные» пункты для мониторинга, по которым судят о производительности/загруженности системы в тот или иной момент. Слева, под дополнительными пунктами мониторинга есть возможные тесты: «Процессор OCCT», «Процессор Linpack», «Память RAM», «3D Адаптив» (доступен только с подпиской), «3D Стандарт», «Память VRAM», «Питание», «Контроль». «Процессор OCCT» (см. Рисунок 4.3) используется для тестирования процессора, можно выбрать набор данных (малый/средний/большой), режим (обычный/экстрим), нагрузку (переменная/постоянная), выбрать начальный цикл, инструкции и выбрать потоки. «Процессор Linpack» (см. Рисунок 4.3) используется для тестирования процессора, можно выбрать версию (2009/2012/2019/2021), какие потоки тестировать, выбрать объем доступной памяти. «Память RAM» (см. Рисунок 4.3) используются для тестирования оперативной памяти, можно изменить кол-во используемой памяти, инструкции и сколько потоки памяти тестировать. «3D Адаптив» и «3D Стандарт» (см. Рисунок 4.3) используются для тестирования видеокарты, можно выбрать какой чип для теста использовать, можно включить/выключить обнаружение ошибок, ограничить нагрузку, выбрать сложность шейдера и длительность теста. «Память VRAM» (см. Рисунок 4.3) используются для тестирования памяти видеокарт, можно изменить кол-во используемой памяти, и видеокарту, к которой привязана память. «Питание» (см. Рисунок 4.3) используются для тестирования питания видеокарт, можно изменить видеокарту и инструкции. Тест «Контроль» необходимо запускать при запущенном стороннем приложении, для контроля поведения системы.





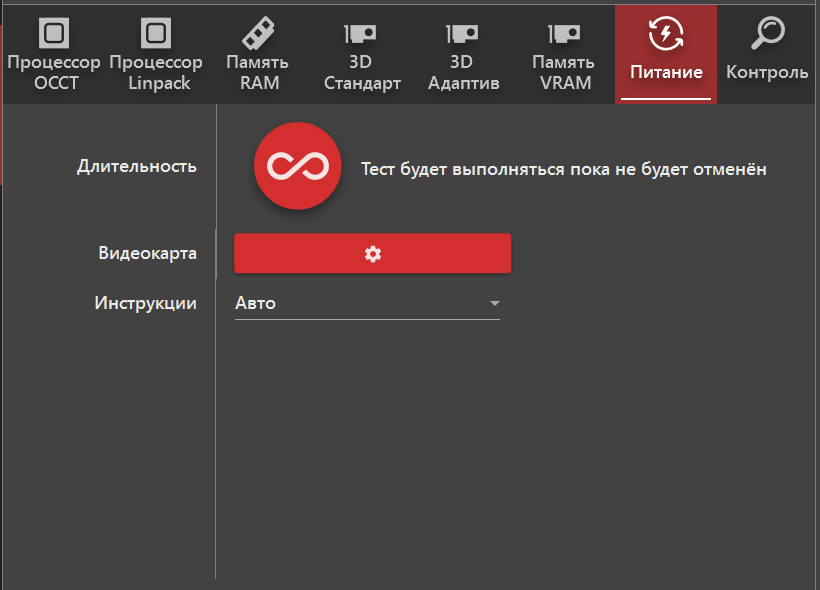
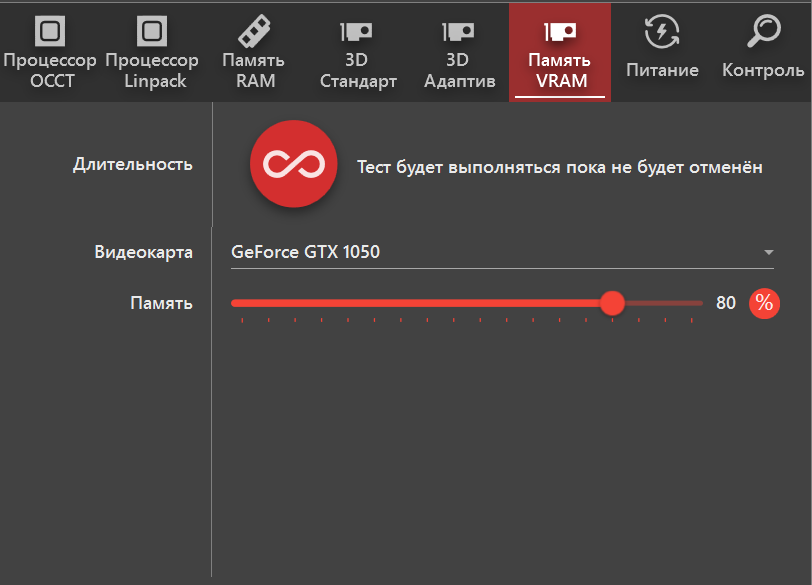


Рисунок 4.3 – Все тесты.

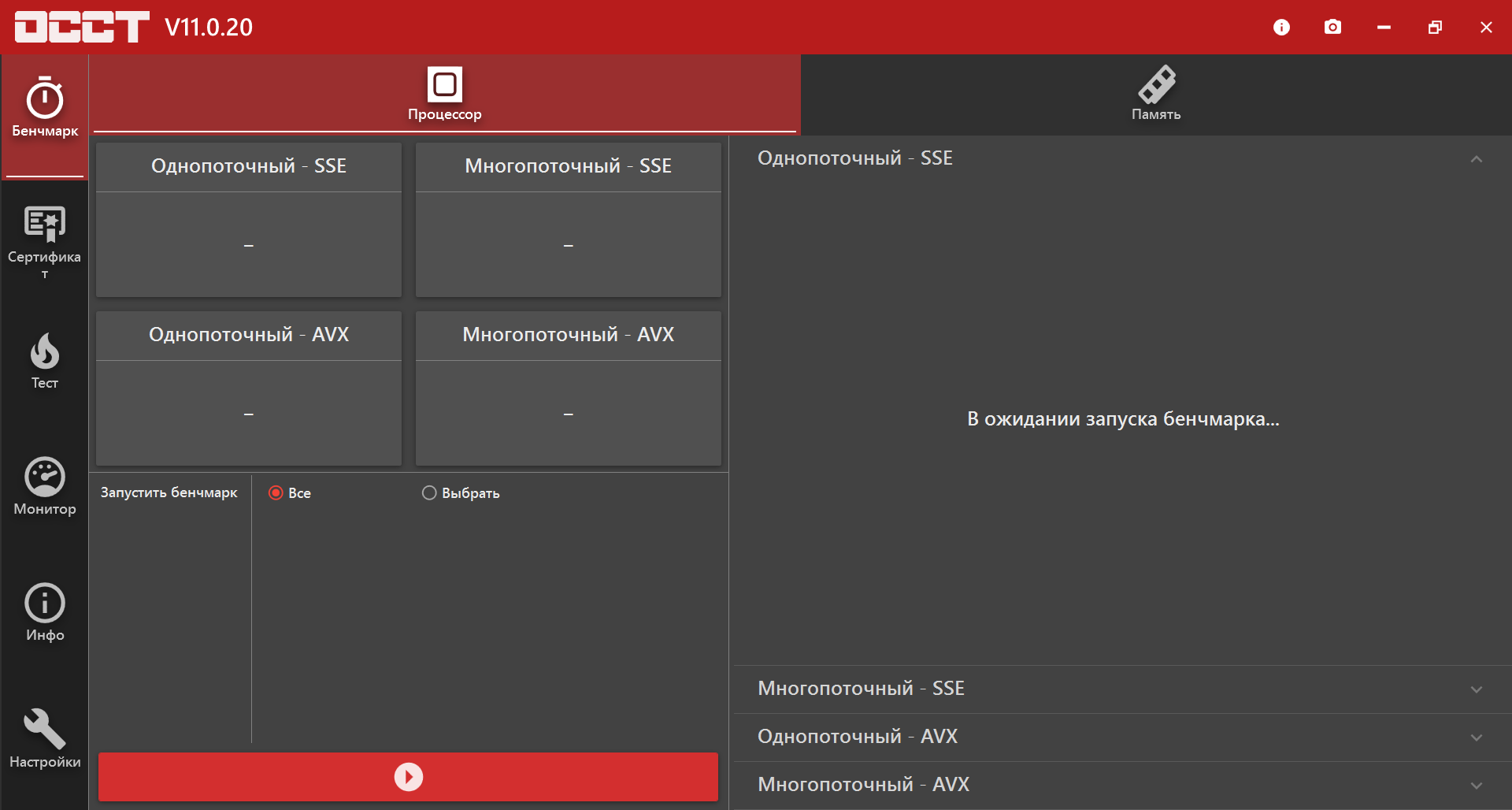


Рисунок 4.4 – Вкладка «Бенчмарк»

На рисунке 4.4 представлена вкладка «Бенчмарк». На ней можно запустить тесты для процессора и оперативной памяти. Имеет пункты тестирования «Процессор» и «Память», информационные окна и одну кнопку пуск. Для запуска тестов необходимо нажать на единственную кнопку пуск.

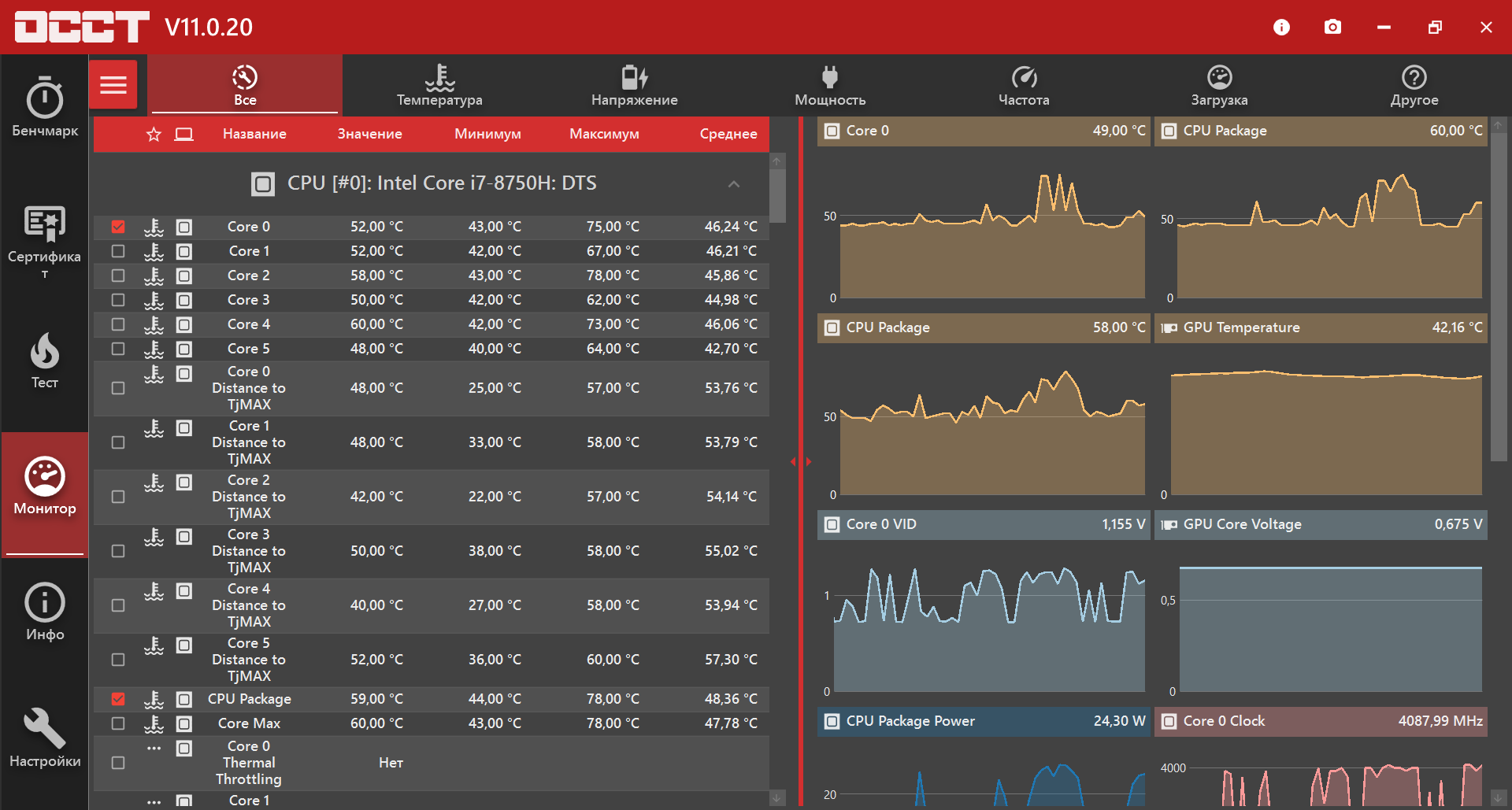


Рисунок 4.5 – Вкладка «Монитор»

На рисунке 4.5 представлена вкладка «Монитор». В которой в реальном времени можно наблюдать за состоянием системы. Есть возможность отдельно следить за температурой, напряжением, мощностью, частотой, загрузкой. В пункте «Другое» находится информация, которую неуместно выделять в отдельные крупные группы, и которую нельзя причислить к уже существующим группам для наблюдения

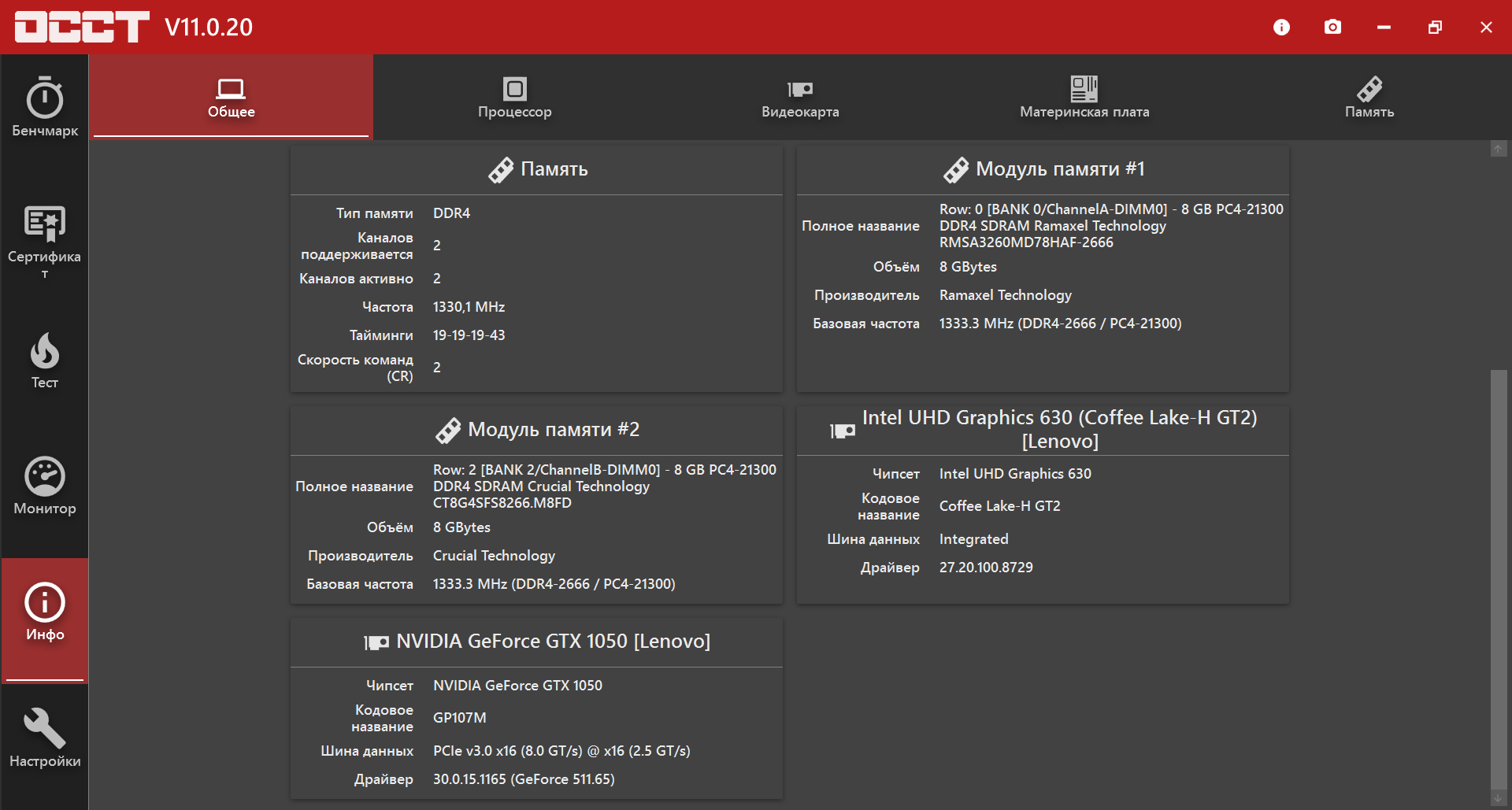


Рисунок 4.6 – Вкладка «Инфо»

На рисунке 4.6 показана вкладка «Инфо». На ней можно узнать информацию о системе: максимальные частоты отдельных модулей, тайминги, производителей, поставщиков драйверов, название ус-ва. К слову, очень раздосадовал факт, что память, почему-то, работает в половину максимальных частот.

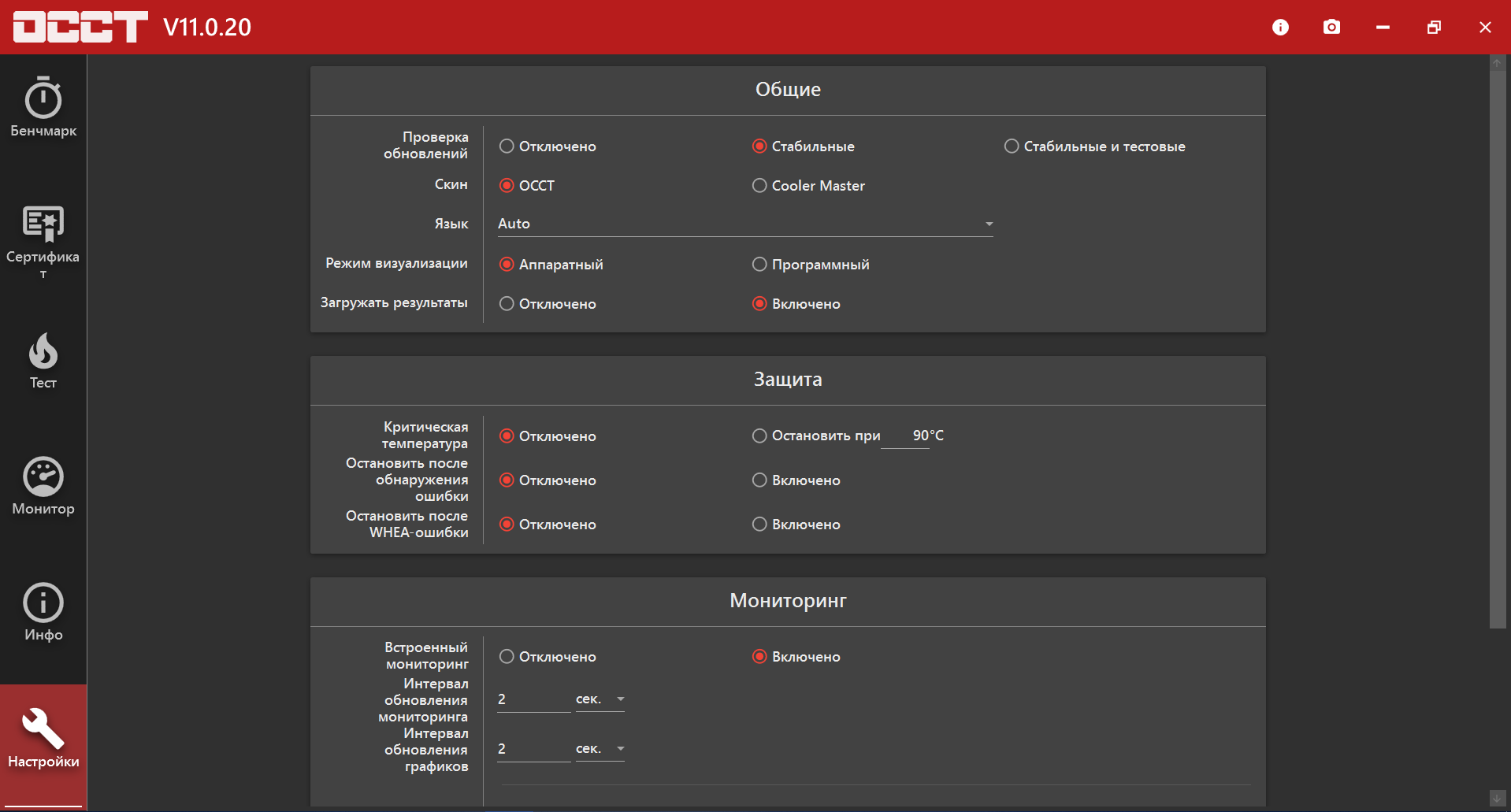
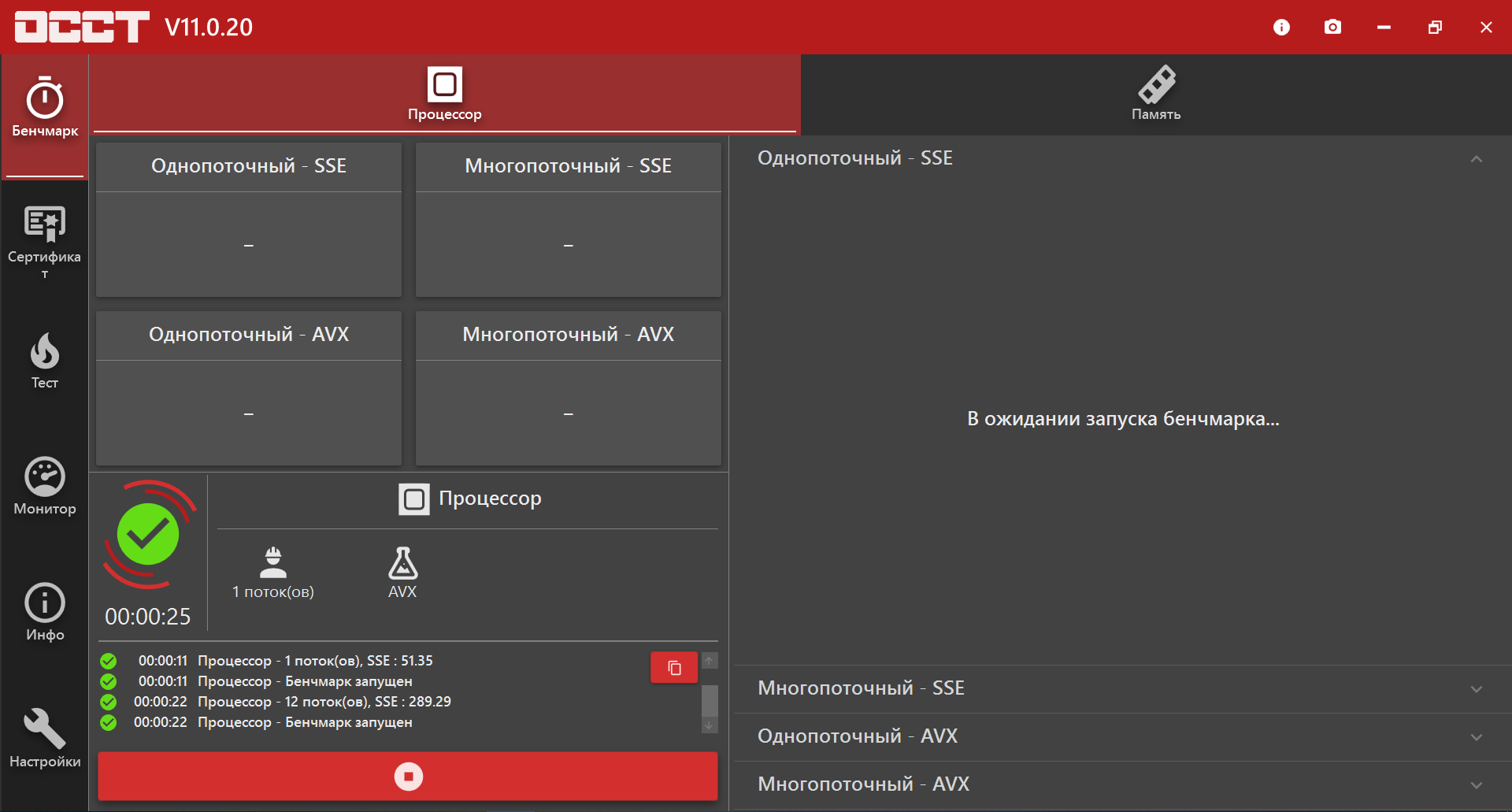


Рисунок 4.7 – Вкладка «Настройки»

На рисунке 4.7 показана вкладка «Настройки». На ней можно настроить отображение параметров системы в приложении. Так же отдельно стоит отметить настройки защиты при тестах. Можно настроить так, чтобы тест отключался при критических температурах или после возникновения ошибки. В параметрах мониторинга так же можно включить дополнительные устройства для наблюдения.



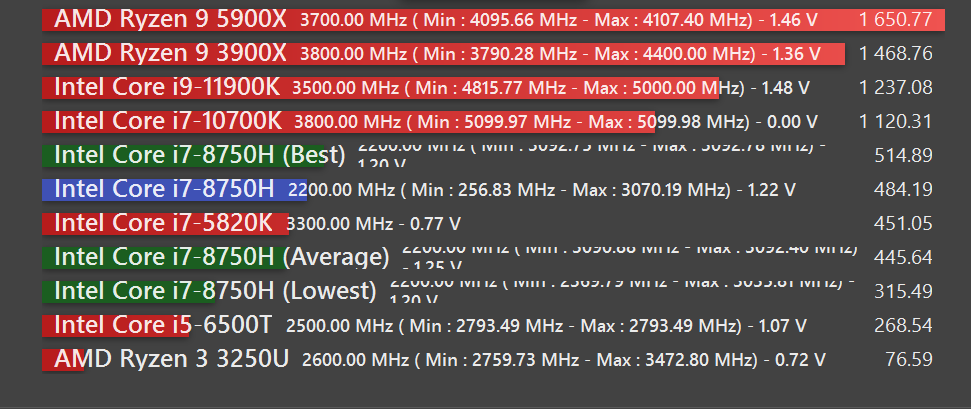


Рисунок 4.8 – Результаты бенчмарка процессора.

В качестве опыта был запущен бенчмарк тест процессора (см. Рисунок 4.8). Так же был запущен тест оперативной памяти (см. Рисунок 4.9). Результат не очень: минимум информации, сильные нагрузки, не понятно каким образом происходит тест памяти, сколько тестов применяется. К тому же в данном тесте процессор задействован на полную, такого не было при тестировании встроенным ПО от Windows. К тому же, таким тестом нельзя полностью проверить память, потому что часть памяти используется системой, по всей видимости по использованной памяти не проходятся. Ещё, я установил ползунок на 80% проверяемой памяти, а проверяло всего 7440 МБ, когда всего памяти 16 ГБ.

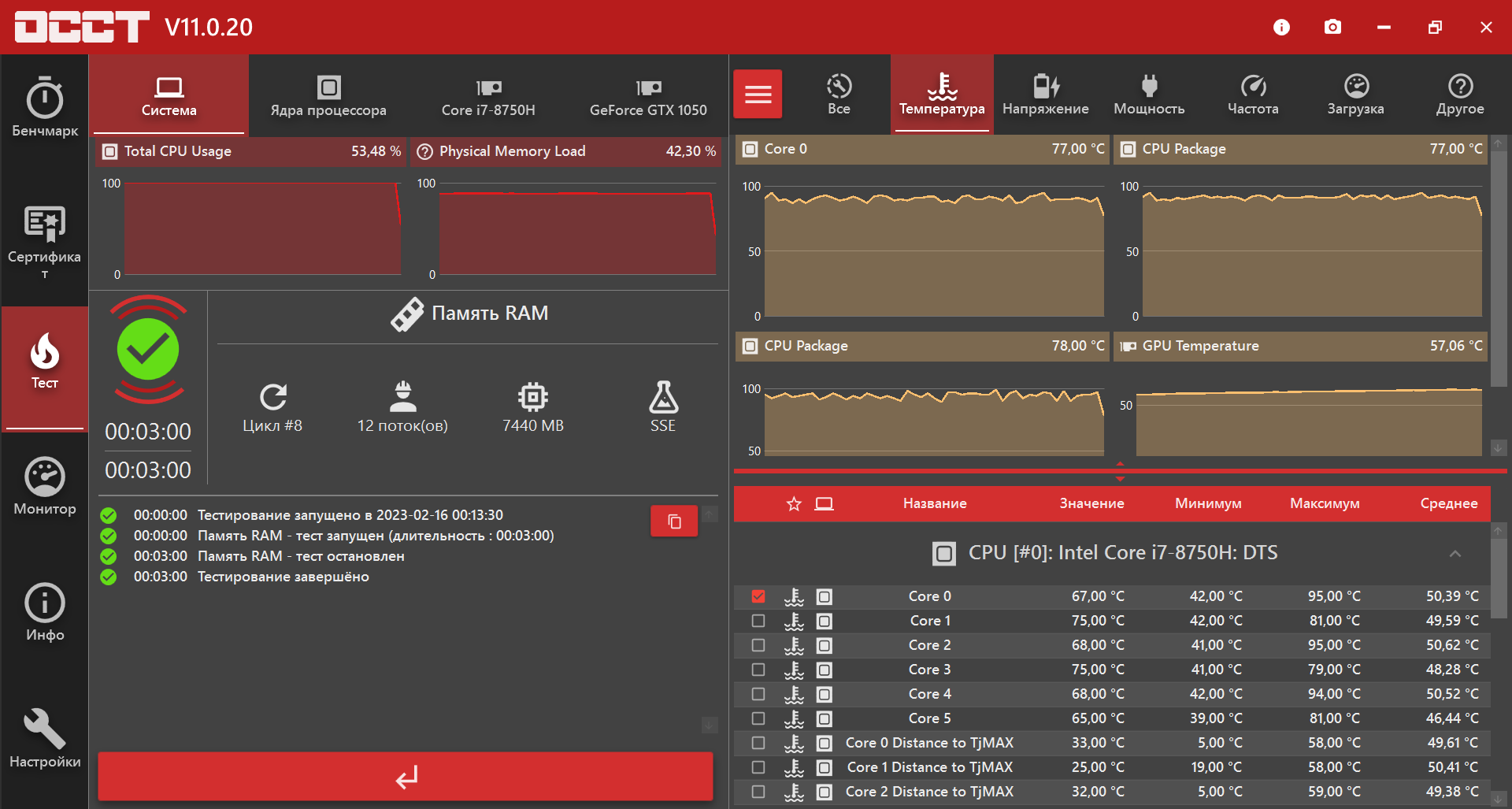


Рисунок 4.9 – Тестирование памяти

В итоге, программа OCCT может быть использована для мониторинга, потому что в ней удобно следить за параметрами, и не более. Тесты и бенчмарк очень слабые и ограниченные. Так же у программы существуют ограничения по времени тестирования и ограничения в использовании связанные с платной подпиской. Как инструмент для тестирования и диагностики с дальнейшим выявлением проблем данное ПО – спорное решение.

В качестве бенчмарка для видеокарты я бы выбрал FurMark. В качестве теста для процессора, видеокарты, памяти, кэша – Aida 64. Оперативную память очень удобно проверять встроенной утилитой от Windows или memtest86.

**4.1.1 Общее описание алгоритма Linkpack**

**Linpack benchmark** является тестом, разработанным как приложение к библиотеке Linpack для тестирования производительности компьютерных систем. По сути это генерирование некоторой системы линейных алгебраических уравнений (СЛАУ)    со случайной плотной квадратной матрицей A и вектором   таким, чтобы решением был вектор , у которого все элементы равны 1, с последующим решением этой СЛАУ. Решение в силу неспецифичности (с рядом оговорок) матрицы делается в несколько этапов. Сначала для матрицы системы выполняется PLU-разложение (P — матрица перестановок, L — левая треугольная с единичными диагональными элементами, U — правая треугольная матрица) методом Гаусса с выбором ведущего элемента по столбцу. На втором этапе с помощью полученного разложения последовательно решается СЛАУ вида   . На третьем этапе вычисляется невязка    и её характеристики, после чего выводятся данные о полученных точности и производительности вычислений. Интересно, что производительность вычисляется только для основной части алгоритма, т. е. туда не включены ни вычисление невязки, ни вычисление норм.

**4.1.2 Математическое описание алгоритма**

Исходные данные: невырожденная (ГСЧ подобран так, чтобы это обеспечивалось, в версии от 1992 г.) квадратная матрица A (элементы ), вектор правой части (элементы bi). В реальности они не вполне входные (элементы матрицы генерирует псевдослучайно сама программа, все они находятся в диапазоне (-1/2, +1/2), ), но для разбора общей схемы алгоритма удобно считать их входными данными.

Вычисляемые данные: левая треугольная матрица L (элементы ), правая треугольная матрица U (элементы ), матрица перестановок P (вычисляется не в развёрнутом виде, а как набор номеров ведущих элементов), вектор решения x (элементы ).

Детальные формулы отдельных частей алгоритма здесь описывать не будем, поскольку они являются самостоятельными алгоритмами и их детальные описания должны быть выполнены раздельно. Поэтому здесь будет приведена только общая схема алгоритма.

Первая часть — разложение матрицы в произведение двух треугольных и матрицы перестановок: A=PLU — выполняется методом Гаусса с выбором ведущего элемента по столбцу. При этом матрица перестановок хранится с помощью вектора, характеризующего выполненные перестановки. Матрица L левая треугольная с единичными диагональными элементами, матрица U — правая треугольная.

Вторая часть состоит из двух решений треугольных СЛАУ. Сначала c помощью прямого хода решается система , затем с помощью обратной подстановки — система  . При этом в тесте треугольные матрицы хранятся в соответствующих местах матрицы A. Поэтому перед началом третьей части её генерируют заново (с теми же стартовыми параметрами для ГСЧ).

Третья часть (для которой не измеряется скорость работы) состоит в вычислении сначала невязки решения  и затем её равномерной нормы, а также нормы решения. По окончании этой части алгоритм выдаёт данные по точности полученного решения, а также вычисляет производительность системы.

**4.1.3 Вычислительное ядро алгоритма**

Основная часть алгоритма — разложение матрицы методом Гаусса с выбором ведущих элементов по столбцу — определяет основную часть операций. Следующая по значимости часть — вычисление невязки и её нормы — реализовано подпрограммой вычисления суммы вектора и произведения матрицы на другой вектор и функцией вычисления равномерной нормы. Третьи по значимости части — прямой ход и обратная подстановка в методе Гаусса.

**4.1.4 Макроструктура алгоритма**

Как уже записано в описании ядра алгоритма, разложение матрицы методом Гаусса с выбором ведущих элементов по столбцу — определяет основную часть операций и при этом является первой макрооперацией. После неё идёт прямой ход метода Гаусса и сразу за ним обратная подстановка. После повторного заполнения матрицы следующей макрооперацией является вычисление невязки решения СЛАУ, а затем — вычисление её и вектора решения равномерных норм.

**4.1.5 Схема реализации последовательного алгоритма**

Математическое описание алгоритма дано выше. При этом в стандартном тесте из-за того, что разложение Гаусса вычисляется на месте исходной матрицы, а также из-за того, что при прямом и обратном ходе решение занимает место правой части, после вычисления решения приходится заново перевычислять как матрицу системы, так и правую часть — для последующего вычисления невязки. После вычисления невязки её норма в принципе может быть вычислена в любом порядке с нормой решения, это не влияет на алгоритм и его результаты.

**Вывод**

В ходе лабораторной работы использовал Windows Memory Diagnostic Tool, для тестирования оперативной памяти. Провел три разных теста: два удачных, один – нет. Кратко описал различные методы тестирования оперативной памяти. Попробовал использовать memtest86, к сожалению, возник конфлик с материнской платой, программа просто не запускалась. Разобрался в работе программы для тестирования и мониторинга системных ресурсов OCCT.