

Складення та відлагодження циклічної програми мовою асемблера процесорів Cortex- M3/M4

МЕТА РОБОТИ: ознайомитись на прикладі циклічної програми з основними командами асемблера процесорів Cortex- M3/M4; розвинути навички складання програми з вкладеними циклами; відтранслювати і виконати покроково в режимі відлагодження програму, складену відповідно до свого варіанту; перевірити виконання тесту.

1. Теоретична частина

Для того, щоб створити новий проект в середовищі розроблення програм Keil μ Vision в меню Project треба вибрати команду «New μ Vision Project». Відкриється вікно створення проекту «Create New Project», в якому необхідно, використовуючи дерево каталогів, вказати потрібну папку. Якщо папка для проекту ще не створена, середовище μ Vision дозволяє зробити це. Не виходячи з вікна створення проекту, потрібно звичайними засобами операційної системи перейти у відповідну папку і створити в ній нову папку. Про всяк випадок ім'я доцільніше вводити латинськими буквами. Ім'я проекту може співпадати з іменем папки.

Тут же появиться вікно «Select Device for Target 'Target 1' » («Виберіть цільовий пристрій, 'Мета 1' »), в якому необхідно вказати для якого ядра мікропроцесора буде призначена програма. Розкрийте список ARM і виберіть ARMCM4_FP – процесор ARM Cortex M4 з арифметичним співпроцесором.

Після вибору процесорного ядра відкриється вікно менеджера оточення реального часу виконання Manage Run-Time Environment. Перша позиція містить компоненти програмного інтерфейсу мікроконтролерів Cortex, які помітно прискорюють і полегшують програмування мовою високого рівня C/C++. Серед них: заголовочні файли з визначеннями регістрів периферійних пристроїв; засоби абстрактного доступу до них; приклади. Друга позиція дозволяє під'єднати уніфіковані драйвери інтерфейсів, такі як драйвер Ethernet, флеш-пам'яті тощо. Інші позиції також надають додаткові опції програмістам мовою C/C++, починаючи від стартових файлів і закінчуючи підтримкою графіки, USB-інтерфейсів та інше.

Для програмування мовою асемблера, тим більше на перших порах, можна опустити всі додаткові можливості, натиснувши відразу (внизу вікна) клавішу «OK». Новий проект буде створено і відображено у вікні проектів з ім'ям «LR11» (рис. 1). Для даного проекту зазначено один цільовий пристрій Target 1 на базі вибраного процесора Cortex-M4F. У вікні проектів показана перша група початкових файлів проекту - «Source Group 1». Зазвичай в проекті достатньо однієї групи початкових файлів, однак в окремих випадках їх може бути кілька. Крім того, для кожної з груп початкових файлів можна встановлювати свої опції (налаштування) середовища.

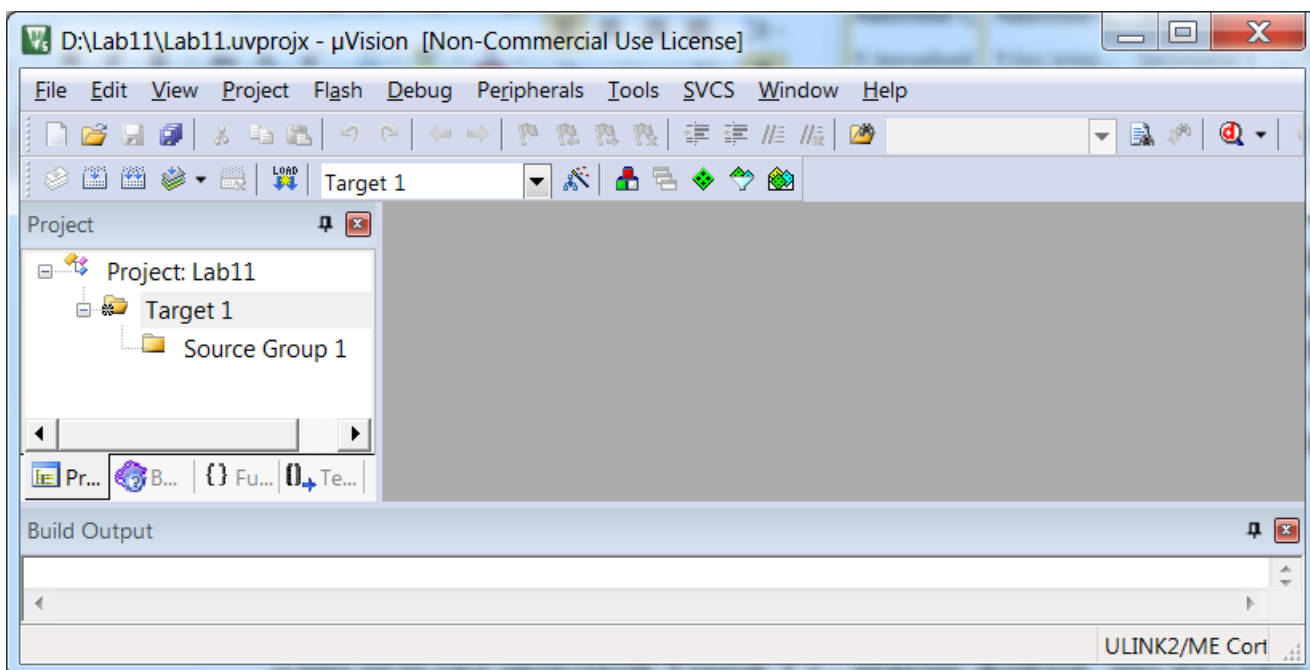


Рис. 1. Вікно середовища µVision з створеним проектом

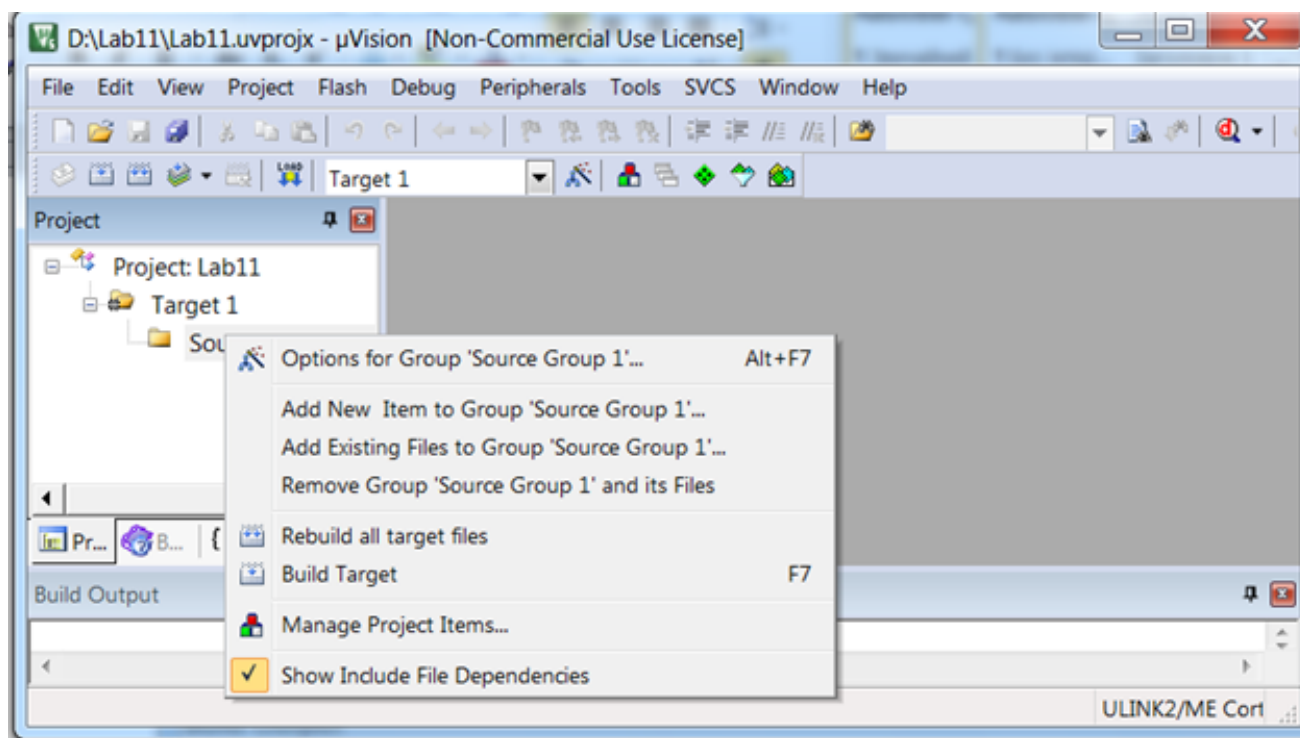


Рис. 2. Долучення нового файлу до проекту

Клацніть правою клавiшею миші на піктограмі групи файлів Source Group 1. У другій позиції контекстно-залежного меню є команда «Add New Item to Group 'Source Group 1'...» -«Додати новий об'єкт в групу 'Група початкових файлів 1' ». Новий об'єкт – це і є новий початковий файл - або мовою асемблера, або мовою СІ.

Вибір наступної позиції цього меню «Add Existing Files to Group 'Source Group 1'» дозволяє додати (під'єднати) вже існуючий файл (створений раніше або просто скопійований з іншого каталогу) до цієї групи.

Виберіть другий рядок контекстного меню, відкриється вікно «Add New Item to Group 'Source Group 1'» (рис. 3). В цьому вікні з типів файлів мовою програмування високого рівня С або С ++, асемблера, заголовочних тощо виберіть тип файлу Asm File (.s), а в полі Name вкажіть ім'я файлу.

Асемблерні файли за замовчуванням мають розширення «.s». Після введення імені файлу і натискання клавіші «Add» файл відразу буде під'єднаний до проекту (ще «порожній») і відкриється вікно редагування, в яке можна ввести текст програми.

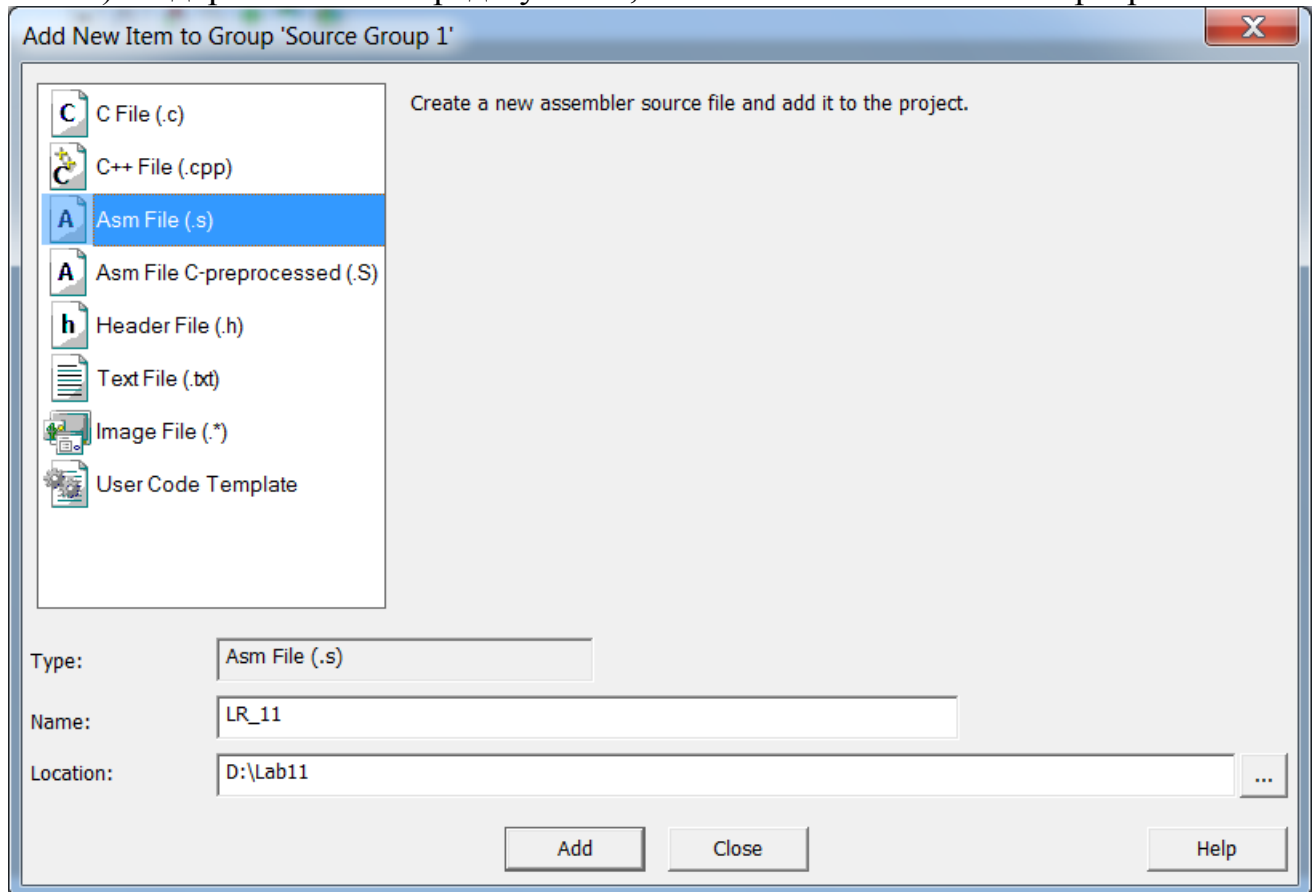


Рис. 3. Вказання типу та імені нового файлу

Тепер введіть просту програму LR_11.s, в якій оголошується кодова секція MyCode, точка входу в програму MyProg, яка повинна бути доступна з інших файлів проекту. Щоб уникнути виконання в процесі налагодження програми «невизначеного» коду програма зациклюється (керування передається на ту ж саму команду «B Stop»).

```
; Програма LR_11.s
; Оголосити кодову секцію MyCode
        AREA MyCode, CODE, ReadOnly
; Оголосити точку входу в програму додатка
        ENTRY
; Оголосити точку входу глобальної змінної
        EXPORT MyProg

MyProg
; Операції додавання / віднімання слів
; Виконати попарне додавання / віднімання
; 32-розрядних слів з масиву Array_W.
; Зберегти суми в пам'яті за адресою Sum_W
; Зберегти різниці в пам'яті за адресою Sub_W
; ініціалізація покажчиків
        LDR r0, = Array_W
        LDR r4, = Sum_W
        LDR r5, = Sub_W
; Перша сума і перша різниця
```

```

        LDR r1, [r0], #4    ; r0 = r0+4
        LDR r2, [r0], #4    ; r0 = r0+4
        ADDS r3, r1, r2
        STR r3, [r4], #4    ; r4 = r4+4
        SUBS r3, r1, r2
        STR r3, [r5], #4    ; r5 = r5+4

; Друга сума і друга різниця
        LDR r1, [r0], #4    ; r0 = r0+4
        LDR r2, [r0]
        ADDS r3, r1, r2
        STR r3, [r4]
        SUBS r3, r1, r2
        STR r3, [r5]
Stop      B Stop

        ALIGN

; Оголосити секцію даних в пам'яті
        AREA MyData, Data, ReadOnly
        EXPORT Array_W
        Array_W DCD 9, 4, 3, 5

; Оголосити секцію даних в оперативній пам'яті
        AREA MyData1, Data, ReadWrite
        EXPORT Sum_W
        EXPORT Sub_W

; Зарезервувати в ОЗУ місце для розміщення
; результатів додавання і віднімання слів
Sum_W     SPACE 4 * 2
Sub_W     SPACE 4 * 2
; Кінець асемблерного тексту
        END

```

Для відображення кирилических букв в середовищі μ Vision увійдіть в меню «Edit» («Редактор») і виберіть команду «Configuration» («Конфігурація»). Появиться вікно Configuration, в якому для поля «Encoding» («Кодування») вкажіть Eastern European. Ви зможете бачити коментарі кирилическими буквами скрізь, за винятком вікна «дизасемблера» і деяких інших вікон середовища μ Vision, де ці символи не відображаються.

Наведену програму скопіюйте і вставте у вікно редагування (рис. 4). Зверніть увагу на те, що у вікні редагування ім'я програми позначено зірочкою «*». Це означає, що вона ще не пройшла етап асемблювання (трансляції). Як тільки це відбудеться, «зірочка» автоматично зникне.

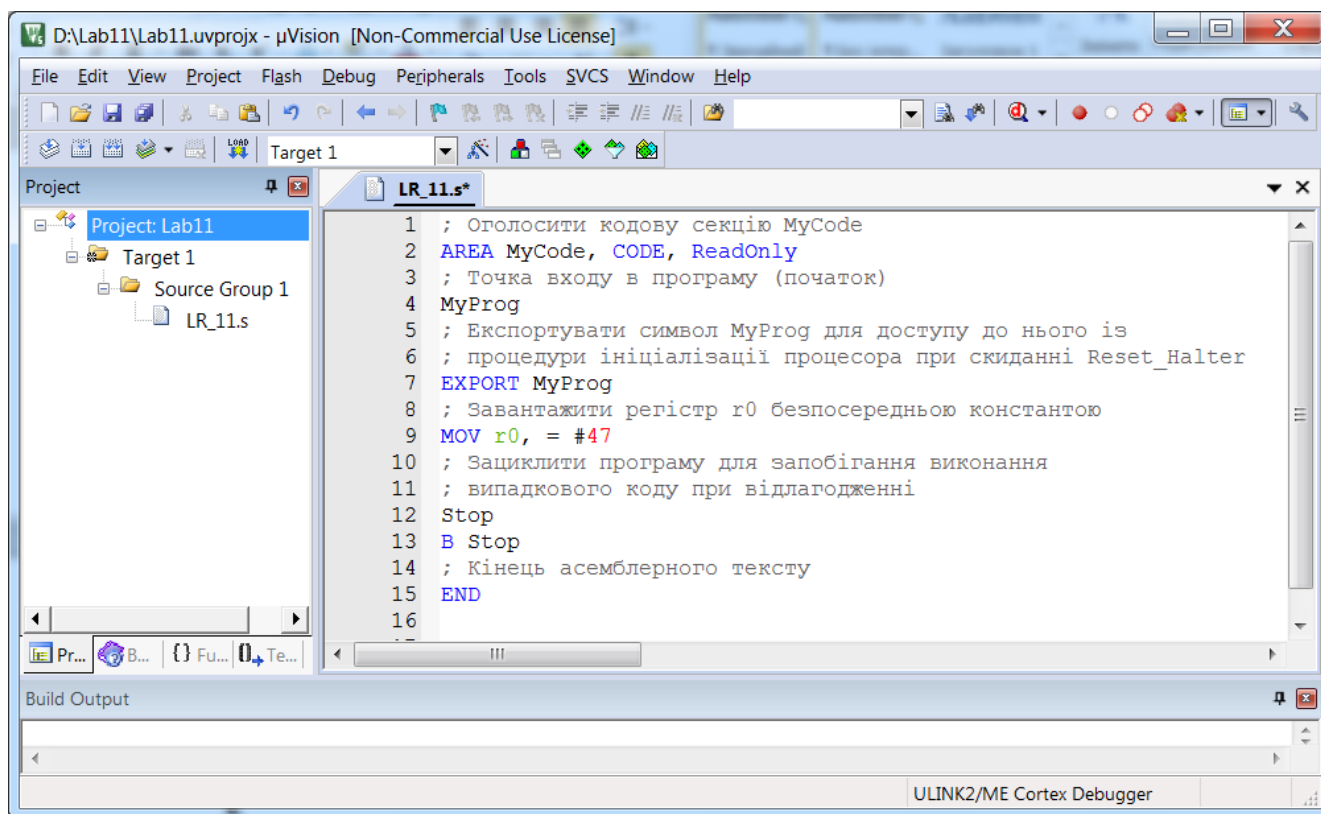


Рис. 4. Проста програма мовою асемблера в середовищі µVision

Для під'єднання стартового файлу викличіть ще раз контекстно-залежне меню до групи файлів Source Group 1 і виберіть команду «Add Existing Files to Group 'Source Group 1'». Появиться вікно «Add Files to Group 'Source Group 1'», в якому вкажіть тип файлу Asm Source file (*.s*; *src*; * .a*), відкрийте папку, призначену для поточного проекту і виберіть файл StartUp_1.s. (Цей файл треба попередньо створити за допомогою текстового редактора Блокнот і помістити папку проекту).

```
; Програмний модуль стартового файлу StartUp_1
; Визначити змінну "Розмір стека" (1 Кбайт)
Stack_Size EQU 0x00000400
; Оголосити секцію даних для розміщення стека системи
; без ініціалізації пам'яті, з атрибутом вирівнювання
; по 8 байтів
        AREA STACK, NOINIT, READWRITE, ALIGN = 3
; Зарезервувати область пам'яті під стек
; з числом байтів Stack_Size
Stack_Mem SPACE Stack_Size; (1 Кбайт)
; Мітка вершини стека (авто-декрементний стек)
__initial_sp

; Vector Table
; Оголосити секцію для розміщення таблиці векторів
; переривань/винятків
; Для компонування визначається як область пам'яті
; даних RESET. Буде автоматично розміщена компонуванням
; на початку пам'яті програм
        AREA RESET, DATA, READONLY
```

```

; Оголосити параметри таблиці векторів - глобальними іменами
EXPORT __Vectors
EXPORT __Vectors_End
EXPORT __Vectors_Size
; Ініціалізація векторів обробників
; переривань/винятків
__Vectors DCD __initial_sp ; Вершина стека - Top of Stack
          DCD Reset_Handler ; Точка виходу в обробник винятку
                                ; за скиданням процесора Reset_Handler
          DCD NMI_Handler ; Точка входу в обробник
                                ; немасковного переривання NMI
;
; ...
; Далі за аналогією можуть бути оголошені й інші
; вектори обробників переривань/винятків
__Vectors_End
__Vectors_Size EQU __Vectors_End - __Vectors

; Оголошення кодової секції для розміщення
; підпрограм обробників переривань/винятків
AREA |.text|, CODE, READONLY

; Обробник переривання за скиданням процесора Reset
Reset_Handler PROC
; Оголосити процедуру Reset_Handler загальнодоступною
; Вона може бути перепрограмована в подальшому
; Тому - використовується опція WEAK ("Слабка мітка")
EXPORT Reset_Handler [WEAK]
; Ініціалізація процесора (через регістр CONTROL)
; ...
; У даній версії стартового файлу не виконується
; Процесор буде працювати за замовчуванням
; в режимі потоку (Thread Mode)
; з привілейованим доступом до всіх ресурсів (Privileged)
; з поки вимкненим співпроцесором FPU

; Передати керування програмі користувача MyProg
; Оголошення точки входу в програму користувача
; (Зовнішня мітка)
IMPORT MyProg
; Псевдо-команда: завантажити адресу MyProg в регістр r0
LDR r0, =MyProg ; r0 <- адреса точки входу
; Непряма передача керування програмі користувача
BX r0 ; PC <- (r0)
ENDP ; Кінець процедури Reset_Handler

; "Порожній" обробник немасковного переривання NMI ("шаблон")
NMI_Handler PROC
; Оголосити процедуру NMI_Handler загальнодоступною
; В подальшому може бути перепрограмована,

```

```

; використана опція WEAK ( "Слабка мітка")
EXPORT NMI_Handler [WEAK]
; Зациклити програму обробника
B .
ENDP ; Кінець процедури NMI_Handler
; Вирівняти кодову секцію по 4-и байтовому слову
ALIGN
; Кінець асемблерного тексту стартового модуля
END

```

Вікно проектів зміниться: тепер в проекті вже два файли - з програмою LR_11.s та стартовий файл StartUp_1.s (рис. 5). Взагалі до проекту можуть під'єднуватися будь-які файли, розміщені в будь-яких каталогах, їх не обов'язково копіювати в каталог поточного проекту.

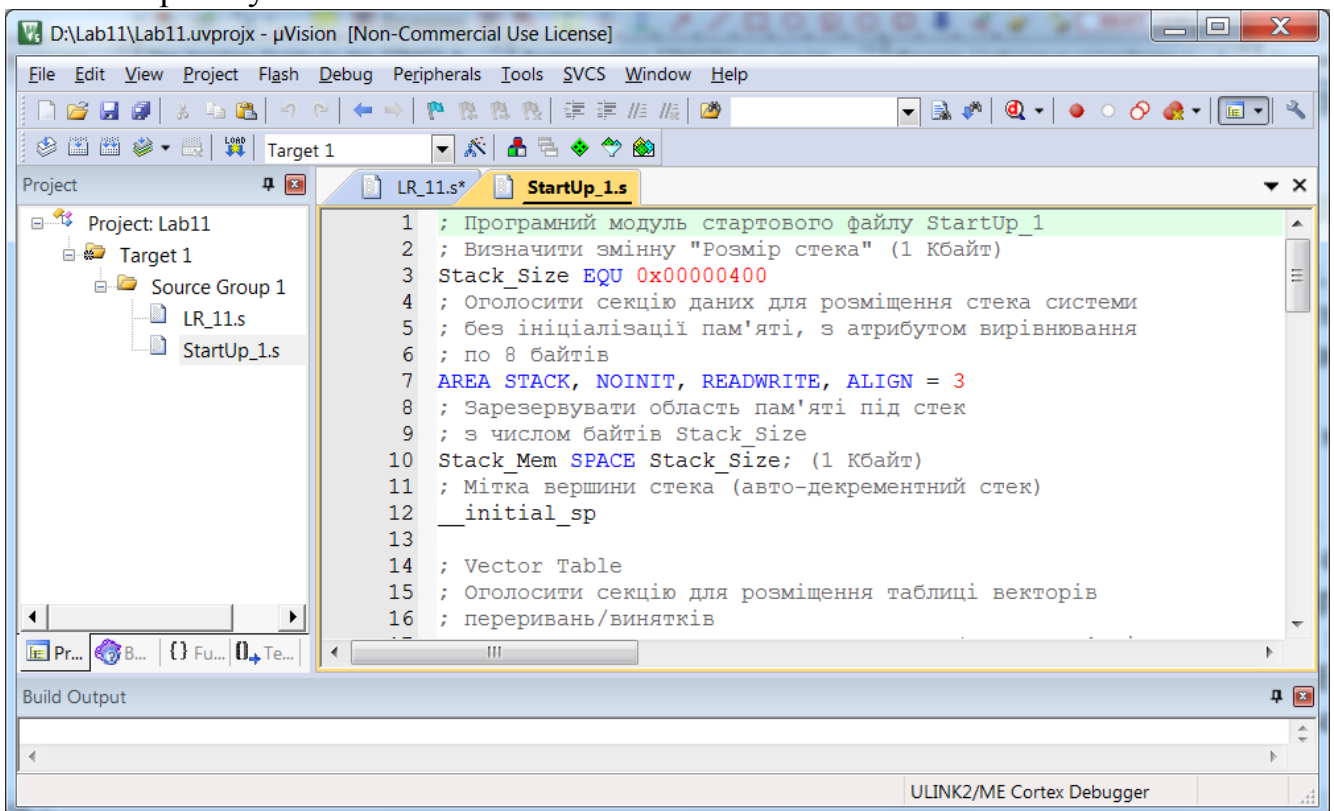


Рис. 5. Проект з під'єднаним стартовим файлом

Перш ніж запустити файли проекту на трансляцію/компіляцію і подальше відлагодження потрібно виконати налаштування середовища, яке дозволяє налаштувати всі інструменти середовища (транслятор, компоновник, відлагоджувач тощо) на потрібний користувачеві режим роботи. При програмуванні мовою асемблера налаштування помітно спрощується, і здебільшого досить буде опцій «за замовчуванням». У міру набуття навичок роботи з середовищем μVision можна самостійно змінювати налаштування, адаптуючи їх до особливостей своїх проектів.

Задати параметри для цільового пристрою можна з меню Проекту: «Project» - «Options for Target 'Target 1'... » або просто клацнувши мишею на піктограмі цієї команди в рядку інструментів.

На першому етапі доцільно встановити всі опції «за замовчуванням» «Defaults» і звірити з наведеними нижче короткими порадами та рекомендаціями.

Вкладка Device. Залиште тип пристрою, який був вибраний при створенні нового проекту.

Вкладка Target. «За замовчуванням». При налагодженні в симуляторі точної конфігурації пам'яті конкретного пристрою не потрібно. Можна прийняти діапазон вбудованої пам'яті програм IROM від 0x до 0x80000 (0.5 Гб), оскільки у відкритій версії середовища μ Vision код все одно обмежується обсягом 32 КБ. Обсягу вбудованої пам'яті даних IRAM 0x20000 (131 Кб) починаючи з адреси 0x20000000 також цілком достатньо. Обидві області кодової пам'яті і пам'яті даних відповідають уніфікованій карті пам'яті процесорів ARM. Решта опцій потрібні тільки при використанні реальної апаратури або при програмуванні мовою C/C++.

Вкладка Output. Досить дозволити створення вихідного виконуваного файлу з ім'ям поточного проекту «Create Executable», генерацію відлагоджувальної інформації «Debug Information» та інформації для переглядачів «Browse Information».

Вкладка Listing. Необхідно дозволити формування файлу лістингу «Assembler Listing» з розширенням «.lst», встановивши максимальну довжину рядка «Page Width» в кількості 132-х символів, щоб коментарі могли розміститися в рядку, а також формування файлу результатів компонування «Linker Listing» з розширенням «.map», який буде містити карту пам'яті «Memory Map», інформацію про символи «Symbols» і перехресні посилання «Cross Reference».

Вкладка User. «За замовчуванням». Використовуються тільки при програмуванні на C/C++. Можна додатково включити опцію видавання звукового сигналу при завершенні трансляції або побудови проекту «Beep When Complete». Якщо включити опцію старту відлагодження «Start Debugging», то відразу після побудови чи перебудови проекту автоматично запускається режим відлагодження.

Вкладка C/C++. «За замовчуванням». Використовується при програмуванні на C/C++.

Вкладка ASM. «За замовчуванням». Не встановлювати опцію Thumb, оскільки в процесорах Cortex-M3/M4 використовується єдиний набір команд Thumb-2, і вибір між генерацією 16- або 32-розрядних команд робиться транслятором автоматично. Опції роздільних областей пам'яті («Independent») і пам'яті тільки для виконання («Execute-only Code») не встановлювати, оскільки кодова пам'ять в процесорах Cortex-M3/M4 буде зберігати як коди, так і дані.

Вкладка Linker. «За замовчуванням». Початкові адреси областей пам'яті «Тільки для читання» R/O Base (0x00000000) і «Для читання і записування» R/W Base (0x20000000) відповідають розташуванню ПЗП і ОЗП в уніфікованій мапі пам'яті процесорів ARM. Корекція не вимагається.

Вкладка Debug. Виберіть налагодження програми в симуляторі «Use Simulator» і дозвольте встановлювати точки зупинки «Breakpoints», роботу з вікнами змінних «Watch Windows», виведення на дисплей вмісту пам'яті «Memory Display», роботу з інструментами «Toolbox» і перегляд стану системи «System Viewer». Дозвольте автоматичне завантаження програми при старті «Load Application at Startup» і виконання програми main () «Run to main ()».

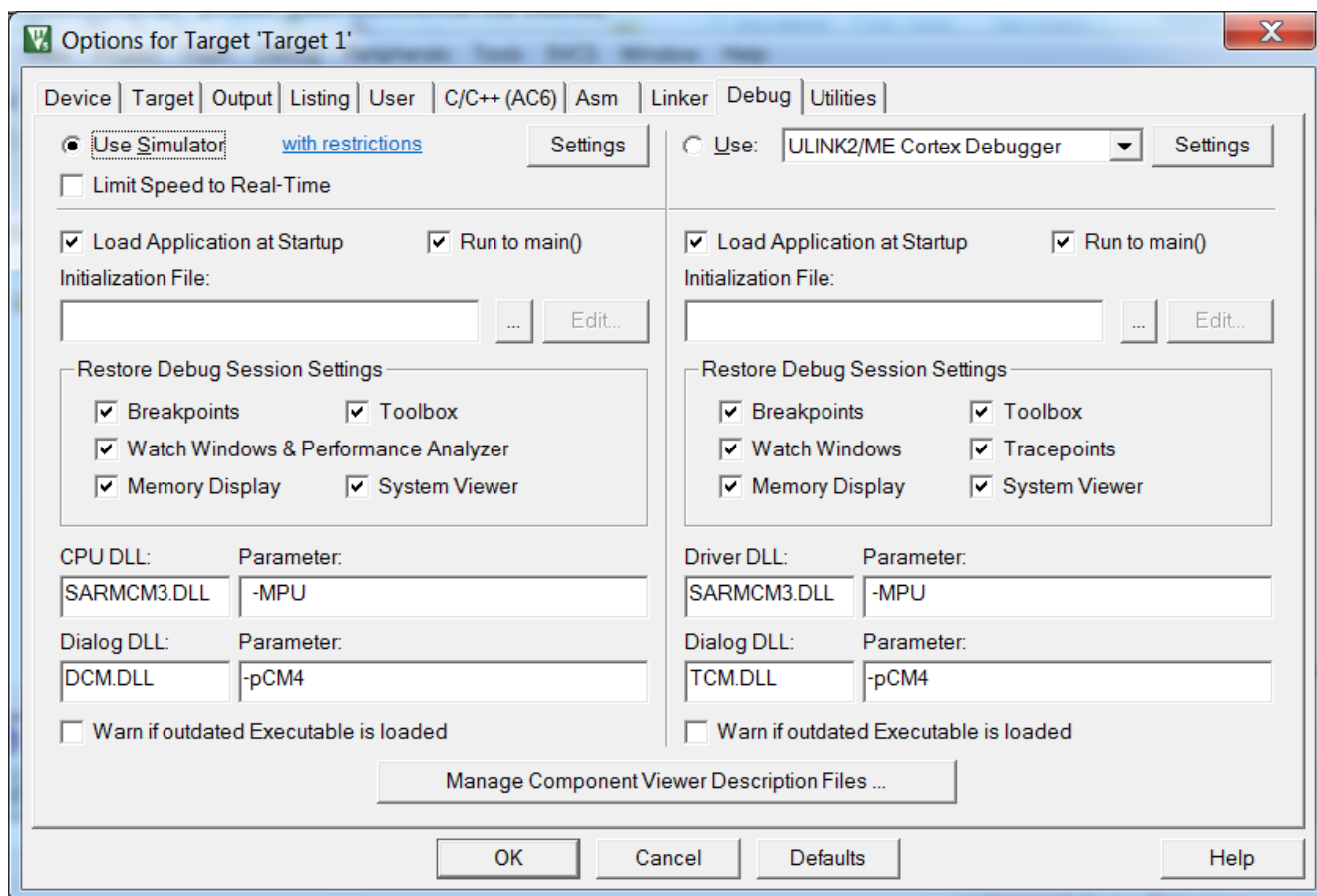







Рис. 6. Налаштування параметрів відлагодження на вкладці Debug

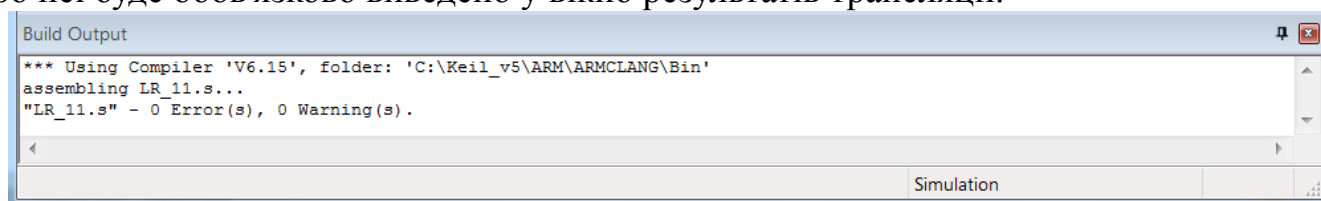
Вкладка Utilites. «За замовчуванням». На початковому етапі утиліти не використовуються.

У міру набуття навичок роботи з середовищем μ Vision можна додатково налаштовувати і опції групи файлів і навіть окремих файлів в групі. Якщо цього не робити, то опції файлів автоматично успадковуються від опцій груп, а опції груп - від опцій цільового пристрою.

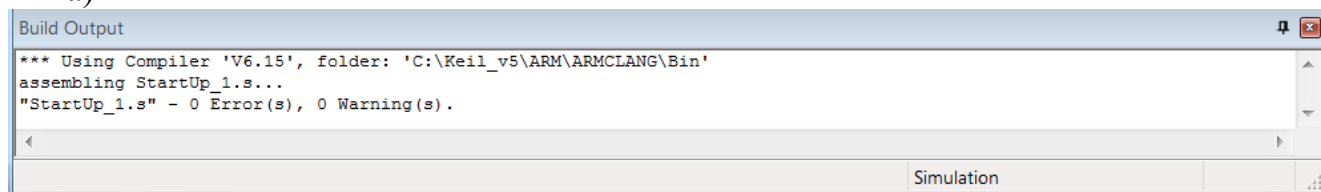
Для виконання трансляції початкових файлів проекту і збирання проекту можна скористатися меню «Проект» («Project») або піктограмами      на панелі інструментів середовища, які доступні простим «клацанням» миші. Перша піктограма запускає процес асемблювання/трансляції («Translate») поточного початкового файлу мовою асемблера або компілювання («Compile») початкового файлу мовою C/C++ (вибір робиться автоматично за розширенням файлу). Друга піктограма запускає процес «збирання» («Build» - Побудувати проект) вже відтрансльованих/відкомпільованих файлів проекту - створює вихідні файли проекту для завантаження і відлагодження. Третя - виконує послідовно обидві дії: трансляцію/компіляцію всіх початкових файлів і нове збирання проекту («Rebuild» - «Перебудувати проект»). На першому етапі (трансляція) створюються вихідні файли в перемістимому об'єктному коді, а на другому (збирання) - у виконуваному об'єктному коді, які можуть завантажуватися на виконання і відлагодження в симулятор або в цілу плату. Четверта піктограма призначена для пакетного оброблення початкових файлів (якщо проект великий і складний), а остання, п'ята, - для зупинення (переривання) запущеного процесу трансляції/компонування.

Перемикаючись між початковими файлами проекту і запускаючи кожен з них на трансляцію, можна переконавшись в тому, що, з точки зору синтаксису мови програмування, в програмах немає синтаксичних помилок. Приклади виведення

результатів трансляції програм у вікно «Build Output» (Вікно виведення результатів побудови проекту) показані нижче (рис. 7). Вони свідчать про те, що синтаксичних помилок в програмах немає. Якщо буде виявлено будь-яку помилку, то повідомлення про неї буде обов'язково виведено у вікно результатів трансляції.



а)



б)

Рис. 7. Результати трансляції програм: а) - LR_11.s; б) - StartUp_1.s

Тепер можна вивчити файли з результатами трансляції. Для цього прямо з середовища μ Vision треба викликати команду відкрити файл (можна клацнути на піктограмі клавішею миші), перейти до підкаталогу поточного проекту, який містить файли лістингів «Listings» та вказати типи файлів з розширенням (*.lst). Всі файли з таким розширенням будуть виведені у вікно. Далі необхідно вибрати потрібний файл (він буде автоматично підсвічений) і натиснути кнопку «Відкрити». У вікні середовища μ Vision з'явиться додаткова вкладка з вмістом файла лістингу. Він містить з правого боку копію програми мовою асемблера, а з лівого - результати її трансляції в машинний код: номер рядка, адресу розміщення даних (в HEX-коді) і код згенерованої машинної інструкції.

У нашій програмі всього дві команди: перша (MOV) є 32-розрядною, а друга (B) - 16-розрядною. Транслятор автоматично робить оптимальний вибір між командами ARM/THUMB уніфікованого єдиного набору команд Thumb-2, створюючи команду найменшої можливої довжини. Також можна переконатись, що директиви Асемблера в машинні коди не транслюються, оскільки є командами для транслятора, а не для процесора. Всі адреси в файлах лістингу є відносними - відраховуються від початкової (нульової) адреси поточної секції.

Отже, обидва вихідних файли проекту пройшли етап трансляції. Тепер можна виконати компонування проекту - побудувати проект. Для цього треба натиснути кнопку «Build» (Побудувати проект) на панелі інструментів. Необхідно звернути увагу на повідомлення у вікні виводу «Build Output», де має бути інформація про те, що при створенні вихідного об'єктного файлу помилок і попереджень не було.

У вікні виводиться також коротка інформація про використану в проекті пам'ять. Вона означає наступне: обсяг в байтах: всього коду проекту Code = 20; пам'яті тільки для читання RO-data = 12, неініціалізованої пам'яті даних для читання і записування RW-data = 0; ініціалізованої нулями пам'яті даних ZI-data = 1024.

В результаті компонування створюється не тільки вихідний об'єктний файл, який можна завантажити на виконання або відлагодження, але і файл карти завантаження «.map» проекту. Перш за все в файлі «.map» можна знайти інформацію про локальні та глобальні символи проекту, їх значення (Value), тип (Type), займаний обсяг пам'яті в байтах (Size) і про об'єкт (секції), в якому вони визначені.

У цьому ж файлі міститься карта пам'яті проекту. Перш за все треба зазначити, що в будь-якому проекті є точка входу в проект (Image Entry point), з якої починається відлагодження проекту. У нашому випадку - це процедура обслуговування переривання за скиданням процесора (розміщена за адресою Reset_Handler = 0xd), яка повинна виконати його ініціалізацію і передати керування додатку користувача.

У карті пам'яті можуть використовуватися такі символи:

- Load Region LR_1 - область завантаження;
- Execution Region ER_RO - область виконання «Тільки для читання»;
- Execution Region ER_RW - область виконання «Для читання і записування»;
- Execution Region ER_ZI - область виконання «Для читання і записування», ініціалізована нулями.

Перші дві області належать до кодової пам'яті (ПЗП), а інші - до пам'яті даних (ОЗП). Область завантаження може містити не тільки код програми, а й додаткові дані, до прикладу таблиці констант, якими повинна бути проініціалізована певна частина пам'яті даних (область ініціалізованих змінних). В нашому випадку весь код потрапляє в область виконання: це таблиця векторів переривань/винятків; код обробників переривань/винятків; власне програма користувача. Область пам'яті даних для читання і записування ER_RW поки що не використана, а область пам'яті для читання і записування, ініціалізована нулями ER_IZ, - використана для розміщення стека.

Проект створено: всі початкові програми пройшли стадію трансляції та компонування. Можна запустити процес відлагодження: з меню «Debug» («Відлагодження») або клацнувши мишею в рядку інструментів на піктограмі «Start/Stop Debug Session» (Старт/Стоп відлагоджувальної сесії).

Раніше було задано параметри цільового пристрою, щоб процес відлагодження виконувався в симуляторі. У цьому випадку після запуску відлагоджувальної сесії код проекту буде завантажено в програму-симулятор і керування кодом буде передано відлагоджувачу (автоматично з'явиться і рядок інструментів відлагоджувача).

У процесі завантаження коду буде автоматично сформовано таблицю векторів переривань/винятків (в нашому випадку її початкову частину), до якої процесор звертається на самому початку своєї роботи, зокрема для ініціалізації покажчика стека SP. Після цього буде викликано обробника переривання за скиданням процесора Reset_Handler. Його початкова адреса і є точкою входу в проект.

Так як обробник стану скидання процесора міститься в стартовому файлі, його буде активізовано (відкрито), і курсор поточної команди, що підлягає виконанню (синій і жовтий трикутнички), буде встановлено на початковій команді обробника.

Можна відкрити і вікно дизасемблера з меню View (Перегляд). У цьому вікні курсор (жовта стрілка) також буде показувати на команду, яка підлягає виконанню. Однак, на відміну від вікна програми в початкових кодах, це буде вже не псевдокоманда, а реальна, створена транслятором, команда процесора.

Таким чином, точка входу в процес відлагодження - це початок процедури оброблення переривання за скиданням процесора Reset_Handler (0xC). Обов'язково відкрийте вікно поточного вмісту регістрів центрального процесора з меню View (Перегляд) або клацнувши на піктограмі цієї команди в рядку інструментів відлагоджувача. Саме в цьому вікні ми будемо відстежувати хід виконання програми. Відзначимо, що поточний вміст регістрів ЦПП автоматично відображається в шістнадцятковій системі числення. Для розкриття/приховування вмісту певної групи регістрів досить клацнути мишею на відповідному прямокутничку (символ «+» - регістри приховані; «-» - відкриті).

В нашому прикладі відкрито вміст регістрів процесорного ядра і внутрішніх регістрів процесора, що відповідають за його режим роботи. Ви бачите, що покажчик стека SP вже ініціалізовано першим значенням вершини стека 0x20000400, лічильник команд PC - початковою адресою обробника Reset_Handler, а сам процесор перебуває в потоковому режимі роботи Thread з привілейованим доступом до всіх ресурсів Privileged, що і повинно бути за замовчуванням. Біт, встановлений в регістрі стану програми PSR - це той, який відповідає за поточний набір команд (ARM або Thumb), з яким в даний момент працює процесор. Так як всі процесори Cortex-M завжди працюють в одному і тому ж режимі Thumb-2 з автоматичною генерацією найбільш компактного коду, то за замовчуванням встановлений саме біт Thumb.

Тепер можна звернутися до команд керування відлагодженням, які є в меню Debug, або до панелі інструментів середовища μ Vision, які більш зручні, так як дозволяють запустити потрібну команду відлагоджувача простим клацанням мишки. Перша команда меню запускає/зупиняє відлагоджувальну сесію, друга - викликає скидання процесора, тобто перехід на обробник переривання Reset_Handler, третя - дозволяє запустити програму в режимі безперервного виконання (прогону - RUN), четверта - зупинити процес безперервного виконання, п'ята, який найбільш часто використовується на етапі вивчення команд процесора, - виконати одну поточну команду - зробити крок. Це так званий режим покрокового виконання програми.

Виконаємо нашу програму в покроковому режимі роботи. Зробимо один крок. Курсор поточної команди, що підлягає виконанню, у вікні з вихідним кодом програми переміститься, вказуючи на наступну команду, а вже виконана команда буде відзначена зліва зеленим прямокутником (ознакою виконання).

Так як ця псевдокоманда повинна завантажити в регістр r0 адресу точки входу в програму додатка, звернемо увагу у вікні регістрів на зміну вмісту регістра r0 (0x19), а факт зміни відзначений у вікні синім фоном. Точно так само змінився і стан лічильника команд PC, який тепер вказує на адресу чергової команди, яка підлягає виконанню. У вікні дизасемблера також буде видно адресу розташування цієї команди.

Зробимо ще один крок. Ми повинні перейти до програми користувача. Середовище μ Vision автоматично активізує вікно цієї програми (ім'я файлу виділяється зеленим фоном і підкресленням) і встановлює курсор зліва від команди, яка підлягає виконанню. Це перша команда нашої простої програми MOV r0, #47. У вікні дизасемблера також видно, що курсор команди, що підлягає виконанню, також перемістився і показує на ту ж команду.

Однак, необхідно звернути увагу на деяку невідповідність. Адреса цієї команди – не число, завантажене раніше в регістр r0 (0x19), а на одиницю менше. Справа в тому, що раніше в процесорах ARM молодшим бітом в команді переходу встановлювався потрібний набір команд процесора (0- ARM, 1 - Thumb), з яким процесор повинен працювати після завершення переходу. Так як в процесорах Cortex-M перемикач з одного набору команд в інший немає і процесор завжди працює в режимі Thumb-2, то для збереження сумісності з попередніми версіями ARM-процесорів адреса переходу завжди встановлюється на 1 більшою, ніж фактична адреса. В процесі виконання команди переходу (в даному випадку команди непрямого передавання керування BX r0 за адресою в регістрі r0) «зайвий» молодший біт автоматично обнуляється, і перехід виконується за правильною адресою, про що свідчить і новий вміст лічильника команд у вікні регістрів.

Зробимо черговий крок і побачимо результат виконання нашої програми - вміст регістра r0 зміниться і дорівнюватиме 0x2F = 47d, як і було потрібно. На черговому кроці виконується команда передавання керування на цю ж команду. Процес виконання

програми зациклюється. Це зроблено для того, щоб в режимі прогону не виникла ситуація виконання «сміттевого» (випадкового) коду. Вміст лічильника команд не змінюється, і процесор весь час виконує одну і ту ж команду переходу на саму себе. Зверніть увагу на те, що обидві команди нашої програми позначені зеленим прямокутником - значить виконані.

2. Порядок виконання роботи

1. В середовищі Keil μ Vision створіть проект Lab11. Відтрансльуйте програми StartUp_1.s та LR_11.s (скопійуйте ці програми з методичних вказівок).
2. Виконайте компонування проекту.
3. Запустіть процес відлагодження.
4. Виконайте проект в режимі відлагодження покроково, слідкуйте за змінами регістрів.
5. В середовищі Keil μ Vision створіть новий проект.
6. Для розміщення матриці опишіть та ініціалізуйте двовимірний масив з розмірами, відповідними до свого варіанту. Елементи матриці задавайте довільними різними дворозрядними цілими додатними або від'ємними числами. Значення елементів рядка чи стовпця, які необхідно перевіряти на виконання умови відповідно до індивідуального завдання, виберіть довільно, але вони мають бути і меншими, і рівними, і більшими за b та c.
7. Напишіть фрагмент програми для транспонування матриці ($b_{j,i} = a_{i,j}$), яку збережіть в іншому масиві.
8. В програмі реалізуйте вказані у своєму варіанті операції оброблення матриці в першому масиві.
9. Всі результати розміщуйте в пам'яті (копіюйте з регістрів в пам'ять).
10. Відтрансльуйте програму StartUp_1.s та розроблену програму.
11. Виконайте компонування проекту.
12. Запустіть процес відлагодження.
13. Виконайте проект в режимі відлагодження, слідкуйте за змінами регістрів.
14. Перевірте результат роботи програми. Наведіть розгорнутий розрахунок скалярного добутку та обчислення кількості та суми елементів, що задовільняють вказаній умові.
15. Збережіть проект.
16. У звіті наведіть текст програми та копії вікон з регістрами і всіма змінними.
17. Зробіть висновки про виконану роботу.

Табл. 1. Команди/директиви, які треба використати в програмі

Команда/директива	Опис
Var_1 DCD 0xff33, 37, 0xffaa55bb Array_W DCD 9, 4, 3, 5	Директива резервування пам'яті під три змінні типу слово з ініціалізацією
CMP {cond} Rn, Operand2	Команда CMP віднімає Operand2 з вмісту регістра Rn, змінює прапорці умов відповідно до результату порівняння, але без збереження результату віднімання.
B {cond} label	Перехід на мітку (за безпосередньою адресою)

Табл. 2. Суфікси умови виконання cond

Суфікс	Прапорці	Значення
EQ	$Z = 1$	«Еквівалентно» (Equal)
NE	$Z = 0$	«Не еквівалентно» (Not Equal)
HS або CS	$C = 1$	«Вище або таке саме» (Higher or same) або «Прапорець Carry встановлений» (Carry Set) Більше або дорівнює (\geq) при порівнянні чисел без знака (unsigned)
LO або CC	$C = 0$	«Нижче» (Lower) або «Прапорець Carry скинутий» (Carry Clear) Строго менше ($<$) при порівнянні чисел без знака (unsigned)
MI	$N = 1$	«Мінус» (Negative) - Від'ємний результат
PL	$N = 0$	«Плюс» (Positive or Zero) - Додатний результат або нуль
VS	$V = 1$	«Переповнення знакове» (Overflow)
VC	$V = 0$	«Немає знакового переповнення» (No Overflow)
HI	$C = 1$ і $Z = 0$	«Вище» (Higher) Строго більше ($>$) при порівнянні чисел без знака (unsigned)
LS	$C = 0$ або $Z = 1$	«Нижче або таке саме» (Lower or same) Менше або дорівнює (\leq) при порівнянні чисел без знака (unsigned)
GE	$N = V$	«Більше або еквівалентно» (Greater then or equal) Більше або дорівнює (\geq) при порівнянні чисел зі знаком (signed)
LT	$N \neq V$	«Менше ніж» (Less then) Строго менше ($<$) при порівнянні чисел зі знаком (signed)
GT	$Z = 0$ і $N = V$	«Строго більше» (Greater then) Строго більше ($>$) при порівнянні чисел зі знаком (signed)
LE	$Z = 1$ або $N \neq V$	«Менше ніж або еквівалентно» (Less then or equal) Менше або дорівнює (\leq) при порівнянні чисел зі знаком (signed)
AL	Б у д ь - я к і з н а ч е н н я прапорців	«Завжди». Суфікс за замовчуванням, який використовується завжди, коли код умовного виконання не специфікується.

3. Індивідуальні завдання

Завдання для ПЗ-21

Варіант	Розмір матриці ($n \times m$)	Операції оброблення матриці	b	c	Умова*
---------	------------------------------------	-----------------------------	---	---	--------

1	(6 × 8)	1. Обчисліть скалярний добуток 2-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-4 2	77	$b < a_i \leq c$
2	(7 × 6)	1. Обчисліть скалярний добуток 1-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-5 1	82	$b \leq a_i \leq c$
3	(6 × 9)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-6 7	94	$b < a_i < c$
4	(8 × 6)	1. Обчисліть скалярний добуток 3-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-2 9	48	$b < a_i \leq c$
5	(6 × 8)	1. Обчисліть скалярний добуток 1-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-3 5	55	$a_i \leq b$ або $a_i > c$
6	(6 × 7)	1. Обчисліть скалярний добуток 2-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-4 3	60	$a_i < b$ або $a_i \geq c$
7	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-2 9	83	$a_i \leq b$ або $a_i \geq c$
8	(6 × 7)	1. Обчисліть скалярний добуток 7-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-4 6	72	$a_i < b$ або $a_i > c$
9	(6 × 8)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові.	-5 8	95	$a_i \leq b$ або $a_i \geq c$
10	(6 × 7)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові.	-2 6	44	$a_i \leq b$ або $a_i > c$
11	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 2-го стовпця, які задовільняють вказаній умові.	-3 4	62	$a_i < b$ або $a_i > c$
12	(6 × 7)	1. Обчисліть скалярний добуток 5-го і 6-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-4 1	73	$a_i \leq b$ або $a_i \geq c$
13	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 2-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-2 8	86	$a_i < b$ або $a_i \geq c$
14	(6 × 7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-3 7	66	$a_i < b$ або $a_i > c$

15	(6×8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-3 9	67	$b \leq a_i \leq c$
16	(9×6)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-4 3	78	$b \leq a_i \leq c$
17	(5×10)	1. Обчисліть скалярний добуток 3-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 10-го стовпця, які задовільняють вказаній умові.	-5 2	83	$b \leq a_i \leq c$
18	(10×5)	1. Обчисліть скалярний добуток 2-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 10-го рядка, які задовільняють вказаній умові.	-6 6	95	$b \leq a_i < c$
19	(6×9)	1. Обчисліть скалярний добуток 4-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-2 8	47	$b \leq a_i \leq c$
20	(8×6)	1. Обчисліть скалярний добуток 3-го і 2-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-3 4	58	$a_i \leq b$ або $a_i \geq c$
21	(6×8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-4 6	63	$a_i < b$ або $a_i \geq c$
22	(6×7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-2 7	84	$a_i \leq b$ або $a_i \geq c$
23	(6×8)	1. Обчисліть скалярний добуток 5-го і 1-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-4 9	73	$a_i < b$ або $a_i \geq c$
24	(7×7)	1. Обчисліть скалярний добуток 1-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові.	-6 1	72	$b \leq a_i < c$
25	(7×6)	1. Обчисліть скалярний добуток 1-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-4 5	85	$a_i < b$ або $a_i > c$
26	(6×7)	1. Обчисліть скалярний добуток 7-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-5 6	82	$a_i < b$ або $a_i > c$
27	(8×6)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-4 9	77	$b \leq a_i \leq c$
28	(9×5)	1. Обчисліть скалярний добуток 2-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-4 6	85	$b < a_i < c$

29	(6 × 7)	1. Обчисліть скалярний добуток 2-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-3 7	69	$b \leq a_i < c$
----	---------	--	---------	----	------------------

*Елемент a_i – це поточний елемент заданого рядка чи стовпця.

Завдання для ПЗ-22

Варіант	Розмір матриці (n × m)	Операції оброблення матриці	b	c	Умова*
1	(6 × 8)	1. Обчисліть скалярний добуток 2-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-4 2	77	$b < a_i \leq c$
2	(7 × 6)	1. Обчисліть скалярний добуток 1-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-5 1	82	$b \leq a_i \leq c$
3	(6 × 9)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-6 7	94	$b < a_i < c$
4	(8 × 6)	1. Обчисліть скалярний добуток 3-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-2 9	48	$b < a_i \leq c$
5	(7 × 8)	1. Обчисліть скалярний добуток 1-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-3 5	55	$a_i \leq b$ або $a_i > c$
6	(8 × 7)	1. Обчисліть скалярний добуток 2-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-4 3	60	$a_i < b$ або $a_i \geq c$
7	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-2 9	83	$a_i \leq b$ або $a_i \geq c$
8	(7 × 9)	1. Обчисліть скалярний добуток 7-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-4 6	72	$a_i < b$ або $a_i > c$
9	(7 × 8)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця які задовільняють вказаній умові.	-5 8	95	$a_i \leq b$ або $a_i \geq c$
10	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові.	-2 6	44	$a_i \leq b$ або $a_i > c$
11	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 2-го стовпця, які задовільняють вказаній умові.	-3 4	62	$a_i < b$ або $a_i > c$

12	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 6-го стовпців. 2. Обчисліть кількість і суму елементів 8-го рядка, які задовільняють вказаній умові.	-4 1	73	$a_i \leq b$ або $a_i \geq c$
13	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 2-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-2 8	86	$a_i < b$ або $a_i \geq c$
14	(8 × 7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-3 7	66	$a_i < b$ або $a_i > c$
15	(7 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові.	-3 9	67	$b \leq a_i \leq c$
16	(9 × 6)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-4 3	78	$b \leq a_i \leq c$
17	(7 × 9)	1. Обчисліть скалярний добуток 3-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-5 2	83	$b \leq a_i \leq c$
18	(9 × 6)	1. Обчисліть скалярний добуток 2-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-6 6	95	$b \leq a_i < c$
19	(7 × 9)	1. Обчисліть скалярний добуток 4-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-2 8	47	$b \leq a_i \leq c$
20	(8 × 6)	1. Обчисліть скалярний добуток 3-го і 2-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-3 4	58	$a_i \leq b$ або $a_i \geq c$
21	(7 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-4 6	63	$a_i < b$ або $a_i \geq c$
22	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 1-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-4 9	73	$a_i < b$ або $a_i \geq c$
23	(6 × 7)	1. Обчисліть скалярний добуток 7-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-5 1	99	$a_i \leq b$ або $a_i \geq c$
24	(6 × 9)	1. Обчисліть скалярний добуток 2-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-2 4	43	$a_i \leq b$ або $a_i > c$
25	(6 × 7)	1. Обчисліть скалярний добуток 5-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-3 2	69	$a_i \leq b$ або $a_i > c$

26	(9×6)	1. Обчисліть скалярний добуток 9-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-4 5	77	$a_i \leq b$ або $a_i \geq c$
27	(6×7)	1. Обчисліть скалярний добуток 7-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові.	-5 1	99	$a_i \leq b$ або $a_i \geq c$
28	(6×9)	1. Обчисліть скалярний добуток 2-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-2 4	43	$a_i \leq b$ або $a_i > c$
29	(6×7)	1. Обчисліть скалярний добуток 5-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-3 2	69	$a_i \leq b$ або $a_i > c$
30	(7×8)	1. Обчисліть скалярний добуток 2-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-3 7	69	$b \leq a_i < c$

*Елемент a_i – це поточний елемент заданого рядка чи стовпця.

Завдання для ПЗ-23

Варіант	Розмір матриці ($n \times m$)	Операції оброблення матриці	b	c	Умова*
1	(6×8)	1. Обчисліть скалярний добуток 2-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 2-го стовпця, які задовільняють вказаній умові.	-4 2	77	$b < a_i \leq c$
2	(7×8)	1. Обчисліть скалярний добуток 1-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-5 1	82	$b \leq a_i \leq c$
3	(6×9)	1. Обчисліть скалярний добуток 3-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-6 7	94	$b < a_i < c$
4	(8×6)	1. Обчисліть скалярний добуток 3-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 8-го рядка, які задовільняють вказаній умові.	-2 9	48	$b < a_i \leq c$
5	(6×8)	1. Обчисліть скалярний добуток 1-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-3 5	55	$a_i \leq b$ або $a_i > c$
6	(8×7)	1. Обчисліть скалярний добуток 2-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-4 3	60	$a_i < b$ або $a_i \geq c$
7	(7×8)	1. Обчисліть скалярний добуток 5-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-2 9	83	$a_i \leq b$ або $a_i \geq c$

8	(8 × 7)	1. Обчисліть скалярний добуток 7-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-4 6	72	$a_i < b$ або $a_i > c$
9	(6 × 8)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-5 8	95	$a_i \leq b$ або $a_i \geq c$
10	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-2 6	44	$a_i \leq b$ або $a_i > c$
11	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-3 4	62	$a_i < b$ або $a_i > c$
12	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 8-го рядка, які задовільняють вказаній умові.	-4 1	73	$a_i \leq b$ або $a_i \geq c$
13	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 2-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові.	-2 8	86	$a_i < b$ або $a_i \geq c$
14	(8 × 7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-3 7	66	$a_i < b$ або $a_i > c$
15	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-3 9	67	$b \leq a_i \leq c$
16	(9 × 6)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-4 3	78	$b \leq a_i \leq c$
17	(6 × 9)	1. Обчисліть скалярний добуток 3-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-5 2	83	$b \leq a_i \leq c$
18	(9 × 7)	1. Обчисліть скалярний добуток 2-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 8-го рядка, які задовільняють вказаній умові.	-6 6	95	$b \leq a_i < c$
19	(6 × 9)	1. Обчисліть скалярний добуток 4-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-2 8	47	$b \leq a_i \leq c$
20	(8 × 6)	1. Обчисліть скалярний добуток 3-го і 2-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-3 4	58	$a_i \leq b$ або $a_i \geq c$
21	(7 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-4 6	63	$a_i < b$ або $a_i \geq c$

22	(8 × 7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-2 7	84	$a_i \leq b$ або $a_i \geq c$
23	(7 × 8)	1. Обчисліть скалярний добуток 5-го і 1-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-4 9	73	$a_i < b$ або $a_i \geq c$
24	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-3 9	62	$a_i \leq b$ або $a_i > c$
25	(7 × 8)	1. Обчисліть скалярний добуток 1-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-4 6	87	$a_i \leq b$ або $a_i > c$
26	(6 × 8)	1. Обчисліть скалярний добуток 2-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-3 2	78	$a_i \leq b$ або $a_i \geq c$
27	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 2-го стовпця, які задовільняють вказаній умові.	-3 3	65	$a_i < b$ або $a_i > c$
28	(9 × 6)	1. Обчисліть скалярний добуток 9-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-4 5	77	$a_i \leq b$ або $a_i \geq c$
29	(7 × 6)	1. Обчисліть скалярний добуток 7-го і 6-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-2 2	81	$a_i < b$ або $a_i \geq c$
30	(8 × 7)	1. Обчисліть скалярний добуток 3-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-3 7	69	$b \leq a_i < c$

*Елемент a_i – це поточний елемент заданого рядка чи стовпця.

Завдання для ПЗ-24

Варіант	Розмір матриці (n × m)	Операції оброблення матриці	b	c	Умова*
1	(6 × 8)	1. Обчисліть скалярний добуток 2-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-4 2	77	$b < a_i \leq c$
2	(7 × 6)	1. Обчисліть скалярний добуток 1-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-5 1	82	$b \leq a_i \leq c$
3	(6 × 9)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-6 7	94	$b < a_i < c$

4	(8 × 6)	1. Обчисліть скалярний добуток 3-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 8-го рядка, які задовільняють вказаній умові.	-2 9	48	$b < a_i \leq c$
5	(7 × 8)	1. Обчисліть скалярний добуток 1-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-3 5	55	$a_i \leq b$ або $a_i > c$
6	(6 × 7)	1. Обчисліть скалярний добуток 2-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-4 3	60	$a_i < b$ або $a_i \geq c$
7	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові.	-2 9	83	$a_i \leq b$ або $a_i \geq c$
8	(6 × 7)	1. Обчисліть скалярний добуток 7-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-4 6	72	$a_i < b$ або $a_i > c$
9	(6 × 8)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-5 8	95	$a_i \leq b$ або $a_i \geq c$
10	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 8-го рядка, які задовільняють вказаній умові.	-2 6	44	$a_i \leq b$ або $a_i > c$
11	(7 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові.	-3 4	62	$a_i < b$ або $a_i > c$
12	(6 × 7)	1. Обчисліть скалярний добуток 5-го і 6-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-4 1	73	$a_i \leq b$ або $a_i \geq c$
13	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 2-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-2 8	86	$a_i < b$ або $a_i \geq c$
14	(6 × 7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові.	-3 7	66	$a_i < b$ або $a_i > c$
15	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові.	-3 9	67	$b \leq a_i \leq c$
16	(9 × 6)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-4 3	78	$b \leq a_i \leq c$
17	(5 × 10)	1. Обчисліть скалярний добуток 3-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-5 2	83	$b \leq a_i \leq c$

18	(10 × 5)	1. Обчисліть скалярний добуток 2-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-6 6	95	$b \leq a_i < c$
19	(6 × 9)	1. Обчисліть скалярний добуток 4-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-2 8	47	$b \leq a_i \leq c$
20	(8 × 7)	1. Обчисліть скалярний добуток 3-го і 2-го стовпців. 2. Обчисліть кількість і суму елементів 8-го рядка, які задовільняють вказаній умові.	-3 4	58	$a_i \leq b$ або $a_i \geq c$
21	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-4 6	63	$a_i < b$ або $a_i \geq c$
22	(6 × 9)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-2 7	84	$a_i \leq b$ або $a_i \geq c$
23	(6 × 8)	1. Обчисліть скалярний добуток 3-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові.	-6 4	97	$b \leq a_i < c$
24	(7 × 7)	1. Обчисліть скалярний добуток 4-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-4 8	69	$a_i < b$ або $a_i > c$
25	(7 × 8)	1. Обчисліть скалярний добуток 2-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-5 3	91	$a_i < b$ або $a_i \geq c$
26	(8 × 6)	1. Обчисліть скалярний добуток 2-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-4 6	73	$b \leq a_i < c$

*Елемент a_i – це поточний елемент заданого рядка чи стовпця.

Завдання для ПЗ-25

Варіант	Розмір матриці (n × m)	Операції оброблення матриці	b	c	Умова*
1	(7 × 8)	1. Обчисліть скалярний добуток 2-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 2-го стовпця, які задовільняють вказаній умові	-4 2	77	$b < a_i \leq c$
2	(9 × 6)	1. Обчисліть скалярний добуток 1-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові	-5 1	82	$b \leq a_i \leq c$
3	(7 × 9)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові	-6 7	94	$b < a_i < c$

4	(8 × 6)	1. Обчисліть скалярний добуток 3-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові	-2 9	48	$b < a_i \leq c$
5	(6 × 9)	1. Обчисліть скалярний добуток 1-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові	-3 5	55	$a_i \leq b$ або $a_i > c$
6	(8 × 7)	1. Обчисліть скалярний добуток 2-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові	-4 3	60	$a_i < b$ або $a_i \geq c$
7	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові	-2 9	83	$a_i \leq b$ або $a_i \geq c$
8	(8 × 7)	1. Обчисліть скалярний добуток 7-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові	-4 6	72	$a_i < b$ або $a_i > c$
9	(6 × 8)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові	-5 8	95	$a_i \leq b$ або $a_i \geq c$
10	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові	-2 6	44	$a_i \leq b$ або $a_i > c$
11	(6 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові	-3 4	62	$a_i < b$ або $a_i > c$
12	(8 × 7)	1. Обчисліть скалярний добуток 5-го і 6-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові	-4 1	73	$a_i \leq b$ або $a_i \geq c$
13	(7 × 8)	1. Обчисліть скалярний добуток 5-го і 2-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові	-2 8	86	$a_i < b$ або $a_i \geq c$
14	(6 × 9)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові	-3 7	66	$a_i < b$ або $a_i > c$
15	(7 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця, які задовільняють вказаній умові	-3 9	67	$b \leq a_i \leq c$
16	(9 × 6)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові	-4 3	78	$b \leq a_i \leq c$
17	(5 × 10)	1. Обчисліть скалярний добуток 3-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 10-го стовпця, які задовільняють вказаній умові	-5 2	83	$b \leq a_i \leq c$

18	(10 × 5)	1. Обчисліть скалярний добуток 2-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові	-6 6	95	$b \leq a_i < c$
19	(5 × 9)	1. Обчисліть скалярний добуток 4-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові	-2 8	47	$b \leq a_i \leq c$
20	(8 × 6)	1. Обчисліть скалярний добуток 3-го і 5-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові	-5 3	87	$b \leq a_i < c$
21	(6 × 8)	1. Обчисліть скалярний добуток 2-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові	-5 5	91	$a_i \leq b$ або $a_i > c$
22	(8 × 7)	1. Обчисліть скалярний добуток 3-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові	-4 1	60	$a_i \leq b$ або $a_i > c$
23	(7 × 8)	1. Обчисліть скалярний добуток 4-го і 6-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові	-2 5	82	$a_i \leq b$ або $a_i > c$
24	(9 × 6)	1. Обчисліть скалярний добуток 2-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-4 9	72	$b < a_i \leq c$
25	(7 × 8)	1. Обчисліть скалярний добуток 2-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові	-3 3	64	$b < a_i \leq c$
26	(8 × 7)	1. Обчисліть скалярний добуток 2-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові	-3 7	69	$b \leq a_i < c$
27	(6 × 7)	1. Обчисліть скалярний добуток 3-го і 6-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-3 5	66	$b \leq a_i < c$

*Елемент a_i – це поточний елемент заданого рядка чи стовпця.

Завдання для ПЗ-26

Варіант	Розмір матриці (n × m)	Операції оброблення матриці	b	c	Умова
1	(5 × 8)	1. Обчисліть скалярний добуток 2-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-4 2	77	$b < a_i \leq c$

2	(7 × 6)	1. Обчисліть скалярний добуток 1-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-5 1	82	$b \leq a_i \leq c$
3	(5 × 9)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 4-го стовпця, які задовільняють вказаній умові.	-6 7	94	$b < a_i < c$
4	(8 × 5)	1. Обчисліть скалярний добуток 3-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-2 9	48	$b < a_i \leq c$
5	(5 × 8)	1. Обчисліть скалярний добуток 1-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 7-го стовпця, які задовільняють вказаній умові.	-3 5	55	$a_i \leq b$ або $a_i > c$
6	(6 × 7)	1. Обчисліть скалярний добуток 2-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-4 3	60	$a_i < b$ або $a_i \geq c$
7	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 3-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-2 9	83	$a_i \leq b$ або $a_i \geq c$
8	(6 × 7)	1. Обчисліть скалярний добуток 7-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-4 6	72	$a_i < b$ або $a_i > c$
9	(5 × 8)	1. Обчисліть скалярний добуток 1-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-5 8	95	$a_i \leq b$ або $a_i \geq c$
10	(6 × 7)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 6-го рядка, які задовільняють вказаній умові.	-2 6	44	$a_i \leq b$ або $a_i > c$
11	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 8-го стовпця, які задовільняють вказаній умові.	-3 4	62	$a_i < b$ або $a_i > c$
12	(6 × 7)	1. Обчисліть скалярний добуток 5-го і 6-го стовпців. 2. Обчисліть кількість і суму елементів 5-го рядка, які задовільняють вказаній умові.	-4 1	73	$a_i \leq b$ або $a_i \geq c$
13	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 2-го рядків. 2. Обчисліть кількість і суму елементів 6-го стовпця які задовільняють вказаній умові.	-2 8	86	$a_i < b$ або $a_i \geq c$
14	(6 × 7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-3 7	66	$a_i < b$ або $a_i > c$
15	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-3 9	67	$b \leq a_i \leq c$

16	(9 × 5)	1. Обчисліть скалярний добуток 5-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-4 3	78	$b \leq a_i \leq c$
17	(4 × 10)	1. Обчисліть скалярний добуток 3-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-5 2	83	$b \leq a_i \leq c$
18	(10 × 4)	1. Обчисліть скалярний добуток 2-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 9-го рядка, які задовільняють вказаній умові.	-6 6	95	$b \leq a_i < c$
19	(5 × 9)	1. Обчисліть скалярний добуток 4-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-2 8	47	$b \leq a_i \leq c$
20	(8 × 5)	1. Обчисліть скалярний добуток 3-го і 2-го стовпців. 2. Обчисліть кількість і суму елементів 7-го рядка, які задовільняють вказаній умові.	-3 4	58	$a_i \leq b$ або $a_i \geq c$
21	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-4 6	63	$a_i < b$ або $a_i \geq c$
22	(6 × 7)	1. Обчисліть скалярний добуток 1-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-2 7	84	$a_i \leq b$ або $a_i \geq c$
23	(5 × 8)	1. Обчисліть скалярний добуток 5-го і 1-го рядків. 2. Обчисліть кількість і суму елементів 5-го стовпця, які задовільняють вказаній умові.	-4 9	73	$a_i < b$ або $a_i \geq c$
24	(6 × 7)	1. Обчисліть скалярний добуток 7-го і 3-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові.	-5 1	99	$a_i \leq b$ або $a_i \geq c$
25	(5 × 9)	1. Обчисліть скалярний добуток 2-го і 5-го рядків. 2. Обчисліть кількість і суму елементів 9-го стовпця, які задовільняють вказаній умові.	-2 4	43	$a_i \leq b$ або $a_i > c$
26	(6 × 7)	1. Обчисліть скалярний добуток 5-го і 7-го стовпців. 2. Обчисліть кількість і суму елементів 2-го рядка, які задовільняють вказаній умові.	-3 2	69	$a_i \leq b$ або $a_i > c$
27	(9 × 5)	1. Обчисліть скалярний добуток 9-го і 4-го рядків. 2. Обчисліть кількість і суму елементів 3-го стовпця, які задовільняють вказаній умові.	-4 5	77	$a_i \leq b$ або $a_i \geq c$
28	(7 × 6)	1. Обчисліть скалярний добуток 7-го і 6-го стовпців. 2. Обчисліть кількість і суму елементів 4-го рядка, які задовільняють вказаній умові.	-2 2	81	$a_i < b$ або $a_i \geq c$
29	(6 × 7)	1. Обчисліть скалярний добуток 2-го і 4-го стовпців. 2. Обчисліть кількість і суму елементів 3-го рядка, які задовільняють вказаній умові.	-3 7	69	$b \leq a_i < c$

*Елемент a_i – це поточний елемент заданого рядка чи стовпця.

