


**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Львівський національний університет імені Івана Франка**  
**Факультет електроніки та комп'ютерних технологій**  
**Кафедра радіоелектронних і комп'ютерних систем**

**Затверджено**

На засіданні  
кафедри радіоелектронних і комп'ютерних  
систем  
факультету електроніки та комп'ютерних  
технологій  
Львівського національного університету  
імені Івана Франка  
(протокол № 1/24 від 28.08. 2023 р.)

Завідувач кафедри:

  
\_\_\_\_\_ Ігор ОЛЕНИЧ

**Силабус з навчальної дисципліни**  
**“Програмування вбудованих систем”,**  
**що викладається в межах ОПП**  
**“ Інженерія програмного забезпечення ”**  
першого (бакалаврського) рівня вищої освіти для здобувачів з  
спеціальності 121 – Інженерія програмного забезпечення

Львів 2023 р.

<b>Назва дисципліни</b>	Програмування вбудованих систем
<b>Адреса викладання дисципліни</b>	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005; вул. Ген. Тарнавського 107, м. Львів, 79011
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет електроніки та комп'ютерних технологій Кафедра радіоелектронних і комп'ютерних систем
<b>Галузь знань, шифр та назва спеціальності</b>	12 – інформаційні технології 121 – Інженерія програмного забезпечення
<b>Викладачі дисципліни</b>	Флюнт О. Є., кандидат фізико-математичних наук, доцент, доцент кафедри радіоелектронних і комп'ютерних систем
<b>Контактна інформація викладачів</b>	Orest.Fliunt@lnu.edu.ua , <a href="https://electronics.lnu.edu.ua/employee/fliunt-orest-yevhenovych">https://electronics.lnu.edu.ua/employee/fliunt-orest-yevhenovych</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю) у комп'ютерному класі № 10 корпусу факультету електроніки та комп'ютерних технологій, вул. Драгоманова 50, м. Львів
<b>Сторінка курсу</b>	<a href="https://e-learning.lnu.edu.ua/course/view.php?id=4968">https://e-learning.lnu.edu.ua/course/view.php?id=4968</a>
<b>Коротка анотація дисципліни</b>	Дисципліна “Програмування вбудованих систем” є нормативною дисципліною з спеціальності 121 – Інженерія програмного забезпечення для освітньої програми “Інженерія програмного забезпечення”, яка викладається в 7-му семестрі в обсязі 4-ох кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
<b>Інформація про дисципліну</b>	Курс розроблено для ознайомлення з основами побудови та програмування сучасних вбудованих цифрових систем; вивчення різних підходів до вирішення задач, які виникають при розробці вбудованих систем; вивчення основ написання програмного коду для мікроконтролерів мовою асемблера та з використанням операційної системи реального часу FreeRTOS.
<b>Мета та цілі дисципліни</b>	<i>Мета:</i> надати студентам основні поняття про вбудовані обчислювальні системи, їхнє призначення та завдання, які вони вирішують; основні принципи побудови вбудованих обчислювальних систем та підходів до їхнього програмування. <i>Цілі:</i> ознайомити студентів з апаратними елементами для побудови вбудованих обчислювальних систем; ознайомити з основними програмними завданнями для побудови систем, наприклад, ввід і вивід даних за допомогою елементарних пристроїв, цифрових кнопочних клавіатур малої розрядності, семисегментних індикаторів, рідкокристалічних дисплеїв та цифрових індикаторів; навчити вирішувати базові задачі такі, як програмування часових затримок за допомогою програмного коду або вбудованих таймерів, використання апаратних переривань; програмувати паралельні процеси за допомогою операційної системи реального часу FreeRTOS з використанням черг Queues, м'ютексів Mutexes, семафорів Semafors.
<b>Література для вивчення дисципліни</b>	1) Хоменко М. А., Велігорський О. А. Програмування вбудованих систем: методичні вказівки до виконання курсового проекту для студентів

	<p>спеціальності 172 "Телекомунікації та радіотехніка" усіх форм навчання. 2019 URI: <a href="http://ir.stu.cn.ua/123456789/19709">http://ir.stu.cn.ua/123456789/19709</a></p> <p>2) Barry R. Mastering the FreeRTOS™ Real Time Kernel [Електронний ресурс]/ Richard Barry// © Real Time Engineers Ltd. – 2016. – Режим доступу до ресурсу: <a href="https://www.freertos.org/fr-content-src/uploads/2018/07/161204_Mastering_the_FreeRTOS_Real_Time_Kernel-A_Hands-On_Tutorial_Guide.pdf">https://www.freertos.org/fr-content-src/uploads/2018/07/161204_Mastering_the_FreeRTOS_Real_Time_Kernel-A_Hands-On_Tutorial_Guide.pdf</a></p> <p>3) Dogan I. Arm-Based Microcontroller Multitasking Projects Using the FreeRTOS Multitasking Kernel/ Ibrahim Dogan., 2019. – 524 с.</p> <p>4) ISOCPP. Resumable Functions (revision 4) [Електронний ресурс]/ ISOCPP – Режим доступу до ресурсу: <a href="https://isocpp.org/files/papers/N4402.pdf">https://isocpp.org/files/papers/N4402.pdf</a></p> <p>5) Філіппенко І. В. Огляд графічних бібліотек для вбудованих платформ / І. В. Філіппенко, В. Р. Корнієнко, Г. К. Кулак// RI. – 2019. – С. 47–53. <a href="https://openarchive.nure.ua/bitstreams/1b9833a6-91a8-4d43-8efe-009a282cb5a4/download">https://openarchive.nure.ua/bitstreams/1b9833a6-91a8-4d43-8efe-009a282cb5a4/download</a></p> <p>6) Development of coroutines usage model for cooperative multitasking implementation on the systems with limited resources [Електронний ресурс] / Amin Salih Mohammed, I. Filippenko, V. B. Saravana та ін.]// Springer. – 2019. Режим доступу до ресурсу: <a href="https://link.springer.com/article/10.1007%2Fs10479-021-04417-1">https://link.springer.com/article/10.1007%2Fs10479-021-04417-1</a></p> <p>7) P.Long, G. Fei, P. Luc, T. Martin.: Behaviour and performance comparison between FreeRTOS and <math>\mu</math>C/OS-III. – International Journal of Embedded Systems. – 2016. – DOI: 10.1504/IJES.2016.077774</p> <p>8) Паращук С. Д. Практикум із процедурно-орієнтованого програмування (мова C): Навчальний посібник. – Кіровоград: ФОП Александра М. В., 2016. – 220 с.</p> <p>9) Мова програмування C, друге видання (The C Programming Language) Браян В. Керніган, Деніс М. Річі (Brian Kernighan and Dennis Ritchie). – 232 с.</p> <p>10) Програмування мікроконтролерів AVR: [навчальний посібник]/ С. М. Цирульник, О. Д. Азаров, Л. В. Крупельницький, Т. І. Трояновська. – Вінниця : ВНТУ. – 2018. – 111 с.</p>
<b>Обсяг курсу</b>	Загальний обсяг: 120 години. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 56 год.
<b>Очікувані результати навчання</b>	<p>У результаті вивчення даного курсу студент буде:</p> <p><b>знати:</b> призначення вбудованих обчислювальних систем; основи архітектури вбудованих систем; основні завдання і алгоритми, які реалізують у вбудованих системах; основні протоколи обміну даними між вузлами вбудованих систем; основні засоби введення і виведення інформації у вбудованих системах; основи програмування в операційній системі реального часу FreeRTOS;</p> <p><b>вміти:</b> аналізувати та будувати структуру простих вбудованих систем; організувати обмін даними між вузлами вбудованої системи; створювати і програмувати прості пристрої вводу і виводу інформації; програмувати часові затримки програмним кодом або таймерами мовою асемблера мікроконтролерів PIC16; програмувати паралельні процеси в</p>

	<p>мікроконтролері модуля STM32 за допомогою операційної системи реального часу FreeRTOS; створювати процеси (Tasks) з заданими пріоритетами, використовувати черги, м'ютекси, семафори.</p> <p>Після вивчення даного курсу «Програмування вбудованих систем» здобувачі набудуть таких Загальних та Фахових компетентностей та Програмних результатів навчання:</p> <p>K01. Здатність до абстрактного мислення, аналізу та синтезу.  K02. Здатність застосовувати знання у практичних ситуаціях.  K03. Здатність спілкуватися державною мовою як усно, так і письмово.  K04. Здатність спілкуватися іноземною мовою мовою як усно, так і письмово.  K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.  K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.  K20. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.  K26. Здатність до алгоритмічного та логічного мислення  ПР23: Вміти документувати та презентувати результати розробки програмного забезпечення.</p>
<b>Ключові слова</b>	Вбудовані обчислювальні системи, мікроконтролер, архітектура CISK і RISK, ввід і вивід даних, часові затримки, операційна система реального часу FreeRTOS, черги Queues, м'ютекси, семафори.
<b>Формат курсу</b>	Очний
<b>Теми</b>	<p>Тема 1. Вбудовані системи, їхнє призначення та застосування.</p> <p>Тема 2. Архітектура CISC і RISC. Застосування архітектури RISC у програмування вбудованих систем.</p> <p>Тема 3. Програмування часових затримок за допомогою програмного коду у мікроконтролерах.</p> <p>Тема 4. Програмування апаратних таймерів у мікроконтролерах Мікрочіп.</p> <p>Тема 5. Застосування апаратних переривань у програмуванні мікроконтролерів для вбудованих систем.</p> <p>Тема 6. Програмування засобів вводу і виводу інформації мовою асемблера мікроконтролерів Microchip.</p> <p>Тема 7. Протоколи передавання і приймання інформації у вбудованих системах. Синхронно-асинхронний приймач-передавач USART.</p> <p>Тема 8. Програмування простих засобів виводу інформації мовою асемблера.</p> <p>Тема 9. Операційна система реального часу FreeRTOS. Симуляція роботи FreeRTOS в операційних системах Linux і Windows.</p> <p>Тема 10. Типи змінних FreeRTOS. Керування завданнями Tasks.</p> <p>Тема 11. FreeRTOS черги Queue. Обмін даними між процесами за допомогою черг.</p> <p>Тема 12. Семафори FreeRTOS. Застосування семафорів для використання спільних ресурсів.</p> <p>Тема 13. Використання м'ютексів у FreeRTOS.</p> <p>Тема 14. Керування розподілом пам'яті в операційній системі FreeRTOS.</p> <p>Тема 15. Програмування таймерів у системі FreeRTOS.</p>
<b>Підсумковий</b>	Екзамен у кінці семестру

<b>контроль, форма</b>	
<b>Пререквізити</b>	<p>Для вивчення даного курсу студентам потрібні базові знання з курсів:</p> <ul style="list-style-type: none"> <li>- вища математика;</li> <li>- дискретна математика;</li> <li>- теорія алгоритмів;</li> <li>- алгоритмізація і програмування;</li> <li>- об'єктно-орієнтоване програмування.</li> </ul>
<b>Навчальні методи та техніки, які будуть використовуватися під час викладання курсу</b>	<p>Інформаційні методи (лекція, бесіда, ілюстрація, демонстрація); дедуктивні методи на основі узагальнень; евристичні методи (проблемна лекція); інтерактивні методи (дискусія).</p>
<b>Необхідне обладнання</b>	<p>Для проведення лекційних занять:</p> <ul style="list-style-type: none"> <li>● монітор TFT 20";</li> <li>● системний блок (процесор Intel i5-6500, 8GB оперативної пам'яті, HDD 256GB);</li> <li>● мультимедійне обладнання (проектор, проекційний екран, дошка настінна, звуковий підсилювач та аудіосистема);</li> <li>● комутатор мережевий для доступу до мережі Internet.</li> </ul> <p>Для проведення лабораторних занять:</p> <ul style="list-style-type: none"> <li>● комп'ютерна лабораторія з 12-14 робочими місцями;</li> <li>● монітори TFT 20";</li> <li>● системні блоки (процесор Intel i3-6500, 8GB оперативної пам'яті, HDD 256GB);</li> <li>● комутатор мережевий для доступу до мережі Internet.</li> </ul> <p>Необхідне програмне забезпечення:</p> <ul style="list-style-type: none"> <li>● операційна система ОС Ubuntu 22 LTS;</li> <li>● операційна система реального часу FreeRTOS;</li> <li>● середовище розробки: MrLabX IDE (версії 6.15).</li> </ul>
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням:</p> <ul style="list-style-type: none"> <li>• Змістовий модуль 1: до 25 балів за виконані лабораторні роботи.</li> <li>• Змістовий модуль 2: до 25 балів за виконані лабораторні роботи.</li> <li>• Екзамен: до 50 балів.</li> </ul> <p>Підсумкова максимальна кількість балів 100. Звітність за курс – екзамен в кінці семестру.</p> <p><b>Академічна доброчесність:</b> Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні завдань є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p><b>Відвідування занять</b> є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів</p>

визначених для виконання всіх видів робіт, передбачених курсом.

**Література.** Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.

**Політика виставлення балів.** Враховуються бали, набрані при поточному контролі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.

Жодні форми порушення академічної доброчесності не толеруються.

**Оцінювання лабораторних робіт** (13 лабораторних робіт, максимальна кількість балів: 50) відбувається шляхом оцінки роботи студента під час захисту звіту по виконаній лабораторній роботі (0-5 балів за одну роботу). У підсумку, всі набрані бали множаться на коефіцієнт (0.77) для переведення у 50-ти бальну шкалу.

Бали оцінювання лабораторних робіт нараховуються за наступним співвідношенням:

5 – студент в повному обсязі володіє навчальним матеріалом, має повне розуміння розглянутої теми, надає правильні відповіді на запитання по темі, код програми функціонує відповідно до завдання;

4 – студент достатньо розуміє розглянутий матеріал та принципи написаного ним коду програми, присутні неточності та незначні помилки у відповідях на запитання по темі, код програми функціонує відповідно до завдання (або з несуттєвими недоліками);

3 – студент не досить добре розуміє розглянутий матеріал та написаний ним код програми, вагається та надає неточні/не конкретні відповіді на запитання по темі, код програми функціонує неточно, або з помірними недоліками;

2 – студент погано розуміє розглянутий матеріал та написаний ним код програми, студент в більшості надає помилкові відповіді на питання по темі, код програми функціонує з суттєвими недоліками;

1 – студент погано розуміє розглянутий матеріал та написаний ним код програми, код програми не функціонує належним чином;

0 – студент зовсім не засвоїв розглянутий матеріал, написаний ним код програми не відповідає темі/не функціонує взагалі.

**Оцінювання екзамену** (3 питання, 17 балів за кожне, але не більше 50 балів за іспит) — за результатами написаних студентом письмових відповідей на 3 питання з білету.

Бали оцінювання питань нараховуються за наступним співвідношенням:

17-13 – розглянуте питання відтворюється в повному обсязі. Засвідчено глибоке володіння матеріалом. Наведені приклади коду повністю робочі та відповідають темі питання. Можуть бути присутні несуттєві помилки та невідповідності;

13-10 – відтворюється значна частина питання. Виявлено знання і розуміння основних положень навчальної дисципліни, проте присутні неточності та/або невідповідності основному питанню. Наведені приклади

	<p>коду частково робочі, проте в загальному відповідають темі питання;  10-7 – відстежується загальне розуміння розглянутого питання. Виявлені множинні неточності та невідповідності, пояснення наведеного коду відсутні;  7-3 – студент погано розуміє розглянуте питання. Виявлені суттєві неточності та невідповідності. Наведені приклади коду з суттєвими недоліками, або не відповідають питанню;  3-0 – студент взагалі не розуміє розглянуте питання. Питання не розкрито, кількість викладеного матеріалу не відповідає загальним нормам. Наведений код не робочий, або відсутній як такий.</p> <p><b>Критерії оцінювання результатів неформальної освіти:</b>  Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо), курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів навчання з навчальної дисципліни.</p>
<b>Питання до заліку чи екзамену</b>	Перелік питань для підготовки до екзамену розміщений на сайті курсу в системі Moodle
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенні курсу.

**Схема курсу “Програмування вбудованих систем”  
для студентів спеціальності 121 – Інженерія програмного забезпечення**

**Перелік лекцій**

<b>Тиж .</b>	<b>Тема, план, короткі тези</b>	<b>Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)</b>	<b>Література.** * Ресурси в інтернеті</b>	<b>Завдання, год</b>	<b>Термін виконання</b>
1	Тема 1. Вбудовані системи, їхнє призначення та застосування.	Лекція	[1], [2], [3], [9], Сайт курсу	2	кінець поточного тижня
2	Тема 2. Застосування архітектури RISK у програмування вбудованих систем.	Лекція	[1], [2], [3], [10], Сайт курсу	2	кінець поточного тижня
3	Тема 3. Програмування часових затримок за допомогою програмного коду у мікроконтролерах.	Лекція	[1], [2], [10], Сайт курсу	2	кінець поточного тижня
4	Тема 4. Програмування апаратних таймерів у мікроконтролерах Мікрочіп.	Лекція	[1], [2], [10], Сайт курсу	2	кінець поточного тижня
5	Тема 5. Застосування апаратних переривань у програмуванні мікроконтролерів для вбудованих систем.	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
6	Тема 6. Програмування простих засобів вводу інформації мовою асемблера. Механічні кнопки, перехресне опитування простої клавіатури, світлочутливі індикатори.	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
7	Тема 7. Протоколи передавання і приймання інформації у вбудованих системах. Синхронно-асинхронний приймач-передавач USART.	Лекція	[1], [2], [3], [5], Сайт курсу	2	кінець поточного тижня
8	Тема 8. Програмування простих засобів виводу інформації мовою асемблера. Світлодіоди, семисегментні індикатори, рідкокристалічні матриці та індикатори.	Лекція	[1], [2], [6], [10], Сайт курсу	2	кінець поточного тижня
9	Тема 9. Операційна система реального часу FreeRTOS. Симуляція роботи FreeRTOS в операційних системах Linux і Windows.	Лекція	[1], [2], [3], [10], Сайт курсу	2	Кінець поточного тижня
10	Тема 10. Типи змінних FreeRTOS. Керування завданнями Tasks.	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
11	Тема 11. FreeRTOS черги Queue. Обмін даними між процесами за допомогою черг.	Лекція	[1], [2], [10], Сайт курсу	2	кінець поточного тижня
12	Тема 12. Семафори FreeRTOS. Застосування семафорів для використання спільних ресурсів.	Лекція	[1], [2], [4], [8], Сайт курсу	2	кінець поточного тижня
13	Тема 13. Використання м'ютексів у FreeRTOS.	Лекція	[1], [2], [4], [8], Сайт курсу	2	кінець поточного тижня
14	Тема 14. Керування розподілом пам'яті в операційній системі FreeRTOS.	Лекція	[1], [2], [4], [8], Сайт курсу	2	кінець поточного тижня
15	Тема 15. Програмування таймерів у	Лекція	[1], [2], [4], [10]	2	кінець



	системі FreeRTOS.		Сайт курсу		поточного тижня
--	-------------------	--	------------	--	-----------------

### Теми лабораторних робіт

Тиж .	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Література.** * Ресурси в інтернеті	Завдання, год	Термін виконання
1	Вступне заняття. Інструктаж з безпеки життєдіяльності.	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
2	ЛР1. Встановлення та налаштування середовища розробки програмного коду та симуляції роботи мікроконтролерів MrLabX IDE	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
3	ЛР2. Ввід і вивід інформації за допомогою цифрових портів мікроконтролера PIC16	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
4	ЛР3. Програмування часових затримок за допомогою програмного коду мікроконтролера PIC16	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
5	ЛР4. Програмування часових затримок за допомогою вбудованого таймера мікроконтролера PIC16	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
6	ЛР5. Вивід цифрової інформації на семисегментні світлодіодні індикатори за допомогою мікроконтролера PIC16	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
7	ЛР6. Опитування стану клавіатури 3*4 за допомогою мікроконтролера PIC16	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
8	Підсумкове заняття ЗМ 1	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
9	ЛР7. Встановлення і симуляція роботи FreeRTOS в операційній системі Linux	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
10	ЛР8. Створення нового проекту FreeRTOS з використанням шаблону	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
11	ЛР9. Робота з завданнями Task в операційній системі FreeRTOS	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
12	ЛР10. Організації обміну даними між завданнями за допомогою черг в операційній системі FreeRTOS	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
13	ЛР11. Організація використання спільних ресурсів за допомогою м'ютексів в операційній системі FreeRTOS	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
14	ЛР12. Використання ресурсів за допомогою семафорів в операційній системі FreeRTOS	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
15	ЛР13. Розподіл використання пам'яті в операційній системі FreeRTOS	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня

16	Підсумкове заняття	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
----	--------------------	--------------------	------------	---	------------------------