

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Волинський національний університет імені Лесі Українки
Факультет інформаційних технологій і математики
Кафедра комп'ютерних наук та кібербезпеки

СИЛАБУС
навчальної практики з
ПРОГРАМУВАННЯ

підготовки бакалавра

спеціальності 122 Комп'ютерні науки та інформаційні технології

освітньо-професійної програми Комп'ютерні науки та інформаційні технології

Силабус навчальної практики з програмування підготовки бакалавра, галузі знань 12 Інформаційні технології, спеціальності 122 Комп'ютерні науки, за освітньою програмою Комп'ютерні науки та інформаційні технології.

Розробник: Глинчук Л. Я., доцент кафедри комп'ютерних наук та кібербезпеки, к.ф.-м.н.

Погоджено

Гарант освітньо-професійної програми:



Гришанович Т. О.

Силабус освітнього компонента затверджено на засіданні кафедри комп'ютерних наук та кібербезпеки

протокол № 2 від 29 вересня 2022 р.

Завідувач кафедри:



Гришанович Т. О.

I. Опис освітнього компонента

Найменування показників	12 Інформаційні технології 122 Комп'ютерні науки Комп'ютерні науки та інформаційні технології бакалавр	Характеристика освітнього компонента
Денна форма навчання		Нормативна
		Рік навчання 2
		Семестр 3-ий
Кількість годин/кредитів 210 / 7		Консультації 14 год.
		Самостійна робота: 196 год
		Форма контролю: залік
Мова навчання: українська		

II. Інформація про керівника практики

ППП Глинчук Людмила Ярославівна
Науковий ступінь кандидат фізико-математичних наук
Вчене звання -
Посада доцент
Контактна інформація hlynchuk.ludmila@vnu.edu.ua
Дні занять <http://194.44.187.20/>

III. Опис практики

1. Анотація практики

Навчальна практика є складовою навчального плану підготовки бакалаврів спеціальності 122 Комп'ютерні науки денної форми навчання. Навчальна практика з програмування як невід'ємна частина навчального процесу, направлена на закріплення теоретичних знань, які одержані здобувачами в процесі вивчення освітніх компонент Програмування, Алгоритмів та структур даних, Комп'ютерної дискретної математики, Лінійної алгебри та аналітичної геометрії.

Практика із програмування орієнтована на розв'язування обчислювальних задач, які вимагають нестандартного підходу, розробку алгоритмів та їх подальшої реалізації на мовах програмування.

2. Пререквізити: Алгоритми та структури даних, програмування, комп'ютерна дискретна математика, лінійна алгебра та аналітична геометрія.

Постреквізити: Програмування, написання курсових робіт та кваліфікаційної роботи, проходження практик.

3. Мета і завдання освітнього компонента: поглиблення і закріплення здобутих теоретичних знань з розробки алгоритмів та програмування; розвиток логічного мислення; набуття професійних навиків з розробки програмного забезпечення, орієнтованого на розв'язання обчислювальних задач.

4. Результати навчання.

Загальні компетентності:

ЗК 1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК 2. Здатність застосовувати знання у практичних ситуаціях.

ЗК 3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК 6. Здатність вчитися й оволодівати сучасними знаннями.

ЗК 7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК 8. Здатність генерувати нові ідеї (креативність).

СК 1. Здатність до математичного формулювання та досліджування неперервних та дискретних математичних моделей, обґрунтування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування

СК 3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.

СК 8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

ПРН 5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислювальних функцій.

ПРН 9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

За результатами практики здобувачі освіти будуть компетентними у таких питаннях:

- розробка алгоритмів для розв'язування обчислювальних задач, які вимагають нестандартного підходу;
- опис алгоритмів із використанням блок-схем;
- вибір парадигми програмування для реалізації алгоритмів;
- вибір середовища розробки з метою реалізації алгоритмів на мові програмування;
- безпосередня реалізація розроблених алгоритмів на обраній мові програмування.

IV. ЕТАПИ ПРАКТИКИ

Етапи	Зміст, основні завдання, тривалість	Консультації	Самостійна робота
Підготовчий	1) Настановча конференція; 2) Консультація з керівником практики та знайомство з колом обов'язків та завданнями практики; 3) Розробка індивідуального графіку роботи студента-практиканта.	2	10
Основний	1) Розробка алгоритмів для запропонованих задач. 2) Знайомство із он-лайн редакторами блок-схем та діаграм.		25
			5

	3) Розробка блок-схем для розроблених алгоритмів. 4) Вибір середовища розробки. 5) Реалізація розроблених алгоритмів на мовах програмування. 6) Тестування розроблених програм. 7) Підготовка коментарів до програмного коду. 8) Відвідування консультації із керівником практики.	6	15 10 50 30 10 5
Підсумковий	1) Узагальнення та систематизація матеріалу щодо проходження навчальної практики. Підготовка звітної документації. 2) Оформлення звіту про практику за поданим зразком. 3) Представлення звіту про практику та його затвердження. 4) Захист практики.	2	15 15 5 5
Всього		10	200

V. ВИДИ (ФОРМИ) ІНДИВІДУАЛЬНИХ ЗАВДАНЬ

Кожен здобувач в якості індивідуального завдання отримує набір із трьох задач, для яких повинен бути розроблений алгоритм та здійснена його програмна реалізація.

Приклад індивідуального набору задач

1. За заданим непарним натуральним числом n вивести зображення розміром $n \times n$, утворене символами ‘ * ‘ та ‘ ‘, як показано у прикладі.

Вхідні дані: Одне непарне натуральне число n ($n > 1$).

Вихідні дані: Зображення розміром $n \times n$.

2. У вхідному файлі послідовно записані N степенів двійки, тобто числа від 2 до 2^N без пропусків ($1 \leq N \leq 1000$). Знайдіть значення N .

3. Леді бере ціле число n і видаляє одну цифру так, щоб число, що утворилось, було максимально можливим. Наприклад, якщо число, яке взяла Леді, 432, то вона видаляє цифру 2, яка стоїть на 3 місці, і отримує число 43.

Напишіть програму, яка знайде максимально можливе число, яке можна отримати після видалення однієї цифри.

Вхідні дані: У єдиному рядку задано одне число n ($10 \leq n \leq 10^{18}$).

Вихідні дані: Виведіть одне число — максимально можливе число, яке можна отримати після видалення однієї цифри.

Приклади

Standard input	Standard output
432	43
139	39
100	10

VI. ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ ПРАКТИКИ

Зміст роботи, що оцінюється	Кількість балів
1. Теоретична підготовка: <ul style="list-style-type: none">- вміння формулювати задачу в термінах предметної області;- вміння будувати модель задачі;- вміння перейти від індивідуальної задачі до масової;- вміння обирати методи для розв'язування задачі;- вміння розробляти алгоритми для задачі;- вміння обирати оптимальні структури даних для розробки алгоритму;- вміння використовувати нестандартні підходи для розв'язування задач;- вміння дотримуватись «чистоти коду».	25
2. Практична підготовка: <ul style="list-style-type: none">- вміння розробляти блок-схему алгоритму;- вміння реалізовувати алгоритми на мові програмування;- вміння реалізовувати структури даних на мові програмування;- вміння використовувати вбудовані засоби мов програмування для реалізації алгоритмів.	55
3. Особистісні характеристики: <ul style="list-style-type: none">– дисциплінованість під час проходження практики;– ініціативність;– самостійність;– професійна спрямованість;– вчасність виконання завдань.	10
4. Оцінювання звітної документації: <ul style="list-style-type: none">– підготовка звіту практики.	10
Сума	100

Політика оцінювання

Політика викладача щодо здобувача освіти

Усі учасники освітнього процесу повинні дотримуватись вимог чинного законодавства України, Статуту і Правил внутрішнього розпорядку ВНУ імені Лесі Українки, загально-прийнятих моральних принципів, правил поведінки та корпоративної культури; підтримувати атмосферу доброзичливості, відповідальності, порядності й толерантності.

Проходження практики передбачає відвідування студентами настановчої конференції перед початком практики, де вони отримують завдання. Поставити питання для керівника практики, уточнити завдання, з'ясувати незрозумілі моменти здобувачі освіти можуть на консультаціях, які проводяться керівником практики відповідно до графіку. Кожен здобувач повинен бути учасником дистанційного курсу “Навчальна практика з програмування”, розміщеного на платформі дистанційного навчання Moodle. (<https://moodle-cs.vnu.edu.ua/course/view.php?id=143>). Передбачено спілкування керівника практики зі здобувачами через форум дистанційного курсу. До захисту практики допускаються здобувачі, які виконали завдання, формили звіт практики та завантажили відповідну документацію до дистанційного курсу. Вимоги до звіту та приклад його оформлення здобувачі можуть переглянути на дистанційному курсі. Обов'язковим є завантаження

Політика щодо академічної доброчесності

Під час навчання учасники освітнього процесу зобов'язані дотримуватися академічної доброчесності: етичних принципів та визначених законом правил, якими мають керуватися учасники освітнього процесу під час навчання, викладання та провадження наукової діяльності.

Дотримання академічної доброчесності здобувачами передбачає: самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю (для осіб з особливим освітніми потребами ця вимога застосовується з урахуванням їх індивідуальних потреб і можливостей); посилення на джерела інформації у разі використання ідей, тверджень, відомостей; дотримання норм законодавства про авторське право; надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності.

Політика щодо дедлайнів та перескладання

Навчальна практика із програмування триває протягом семестру. Терміни практики оголошуються керівником практики на настановчій конференції, зазначаються у дистанційному курсі. Матеріали практики, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку. У випадку, коли здобувач освіти не виконав завдання практики у зазначені терміни, він має можливість виконати завдання під час ліквідації академічної заборгованості.

VI. Шкала оцінювання

Оцінка в балах за всі види навчальної діяльності	Оцінка
90–100	Зараховано
82–89	
75–81	
67–74	
60–66	

1–59	Незараховано (з можливістю повторного складання)
------	--

V. Підсумковий контроль

Підсумковою формою контролю освітнього компонента Навчальна практика з програмування є залік. Оцінювання навчальних досягнень здійснюється за 100 бальною шкалою. Оцінювання здійснюється на основі представленого звіту, демонстрації виконання розроблених програм, захисту матеріалів практики, що відбувається за присутності усіх студентів-практикантів та керівника практики.

Якщо за результатами виконання завдань накопичено не менше 60 балів і студент погоджується із цим результатом, то виставляється оцінка за семестр. В іншому випадку здобувач освіти має можливість скласти залік під час ліквідації академічної заборгованості.

Ліквідація академічної заборгованості із практики передбачає виконання того ж набору індивідуальних задач із подальшим захистом результатів практики.

VI. Рекомендована література та інтернет-ресурси.

Основна література

1. Глинчук Л. Я., Гришанович Т. О., Ступінь А. П. Реалізація стандарту симетричного шифрування DES мовою програмування C та порівняння часу його роботи з відомими утилітами. *Кібербезпека: освіта, наука, техніка*. 2021. Т. 2, № 14. С. 118–130.
2. Гришанович Т. О. Алгоритми та структури даних: навчальний посібник. Луцьк : ВНУ ім. Лесі Українки, 2021. 150 с. (Рекомендовано Вченою радою ВНУ імені Лесі Українки, протокол №13 від 28.12.2021)
3. Ільман В. М., Іванов О. П., Панік Л. О. Алгоритми, дані і структури : навч. посіб. Дніпро : Дніпропет. нац. ун-т залізн. трансп.ім. акад. В. Лазар., 2019. 134 с.
4. Крєневич А. П. Алгоритми і структури даних : Підручник. Київ : ВПЦ "Київ. Ун-т", 2021. 200 с.
5. Махровська Н.А., Погромська Г. С. Алгоритми і структури даних: навчально-методичний посібник. Миколаїв : МНУ ім. В.О. Сухомлинського, 2019. 279 с.
6. Основи об'єктно-орієнтованого програмування : навч. посібник / Гришанович Т. О., Глинчук Л. Я.; ВНУ імені Лесі Українки. Електронні текстові дані (1 файл: 998 КБ). Луцьк : ВНУ імені Лесі Українки, 2022. – 120 с. // Режим доступу <https://evnuir.vnu.edu.ua/bitstream/123456789/20320/1/oor.pdf>
7. Програмування: підручник [Електронний ресурс] / укладач Л. Я. Глинчук, Т.О. Гришанович; ВНУ ім. Лесі Українки. Електронні текстові дані (1 файл: 3 201 КБ). Луцьк: ВНУ ім. Лесі Українки, 2022. 160 с. // Режим доступу : https://evnuir.vnu.edu.ua/bitstream/123456789/20649/1/hlynchuk_hryshanovych.pdf

Додаткова література та Інтернет-ресурси

1. Кузьменко І. М., Дацюк О. А. Базові алгоритми та структури даних : навч. Посіб. Електронне мережеве навчальне видання. Київ : КПІ ім. Ігоря Сікорського, 2022. 137 с.
2. Курс | Algorithms101 | Prometheus. *Prometheus – Найкращі онлайн-курси України та світу*. URL: https://courses.prometheus.org.ua/courses/KPI/Algorithms101/2015_Spring/course/.

3. Ришковець Ю. В. Алгоритмізація та програмування. Частина 2. : навч. посіб. Львів : «Новий Світ-2000», 2020. 320 с.
4. Шаховська Н. Б., Голошук Р. О. Алгоритми і структури даних. Навчальний посібник. Львів : Магнолія, 2018. 216 с.
5. Data Structures and Algorithms. Coursera | Online Courses & Credentials From Top Educators. Join for Free | Coursera. *Coursera*. URL: <https://www.coursera.org/learn/algorithmic-toolbox/home/week/1>.
6. Sort Visualizer. *Sort Visualizer*. URL: <https://www.sortvisualizer.com/> (date of access: 28.08.2022).
7. Aho A. V., Hopcroft J. E., Ullman J. D. Data Structures and Algorithms. Murray Hill, Ithaca, Stanford, 2001. 620 p.