

УДК 378.2(4+477)

**Владислав Круглик,
кандидат педагогічних наук, доцент
(Україна, Мелітополь, Мелітопольський державний педагогічний
університет ім. Б. Хмельницького)
Vladyslav Kruhlyk,
candidate of science, associate professor
(Ukraine, Melitopol, Melitopol State Pedagogical University named after
B. Khmelnytskyi)**

**Вячеслав Осадчий,
доктор педагогічних наук, професор
(Україна, Мелітополь, Мелітопольський державний педагогічний
університет ім. Б. Хмельницького)
Viacheslav Osadchyi
doctor of science, professor
(Ukraine, Melitopol, Melitopol State Pedagogical University named after
B. Khmelnytskyi)**

**Світлана Симоненко,
аспірант
(Україна, Мелітополь, Мелітопольський державний педагогічний
університет ім. Б. Хмельницького)
Svitlana Symonenko,
postgraduate
(Ukraine, Melitopol, Melitopol State Pedagogical University named after
B. Khmelnytskyi)**

**Аналіз змісту та організації підготовки фахівців з програмної
інженерії в університетах США**

У статті висвітлюються питання підготовки фахівців з програмної інженерії у провідних університетах Сполучених Штатів Америки. Представлено статистичні дані щодо популярності та затребуваності професії інженера-програміста у світі. Проаналізовано рекомендації для розробки навчальних планів підготовки бакалаврів та асоціатів з програмної інженерії, розроблені Міжнародними організаціями Association for Computing Machinery (ACM) та IEEE Computer Society (IEEE CS) у рамках Навчального плану з комп'ютеризації, їх структуру та відмітні особливості. Порівняно навчальні плани підготовки фахівців з програмної інженерії у провідних університетах США (Рочестерському технологічному інституті та Державному університеті Сан Хосе). Висвітлено спільні аспекти у навчальних планах з програмної інженерії у вищезазначених університетах. Акцентовано увагу на необхідності вивчення, адаптації та впровадження позитивного досвіду університетів США щодо підготовки фахівців з програмної інженерії в університетах України.

Ключові слова: *професійна підготовка, програмна інженерія, навчальний план, інженер-програміст, освітні знання з програмної інженерії.*

Analysis of content and organization of software engineer training at the universities of the USA

The paper deals with the problems of software engineer training at the advanced universities of the USA. Modern statistical data proving the popularity and high demand for a software engineer (a software developer) profession in the world are presented. Some ranking data related to countries of the world and numbers of best universities providing the best IT-education are demonstrated. It is focused that there are 63 universities from the USA in the Academic Ranking of World Universities in Computer Science – 2015 (Shanghai Ranking). It is noted that in the USA educational establishments offer different degrees: Associate's Degree in Software Engineering, Bachelor's Degree in Software Engineering, Master's Degree in Software Engineering and PhD in Software Engineering. Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering developed by

international organisations (Association for Computing Machinery (ACM) and IEEE Computer Society (IEEE CS), their content and peculiarities are analysed. It is shown that in university curricula much attention should be given not only to technical and professional knowledge, but to interpersonal relations, communication and problem-solving skills. Software engineering curricula of leading American universities (Rochester Institute of Technology and José State University) are compared. Similar aspects in software engineering curricula of these universities are highlighted. It was shown that professional training, liberal arts and sciences, natural sciences, wellness education and language knowledge are important for software engineers. It is emphasized that it is necessary to study, adapt and implement the positive experience of software engineer training of American universities to Ukrainian higher educational establishments.

Key words: *professional training, software engineering, curriculum, software engineer, software engineering education knowledge (SEEK).*

Постановка проблеми у загальному вигляді. Стрімке зростання рівня розвитку науки, техніки та технологій, соціально-економічний розвиток суспільства, виникнення нових областей знань призводять до підвищення вимог до якості професійної підготовки фахівців усіх галузей. Удосконалення професійної підготовки фахівців з програмної інженерії, критичний аналіз та модернізація освітніх програм, оновлення змісту професійної та загальної підготовки, узгодження кваліфікацій у відповідності до Європейських кваліфікацій є пріоритетними напрямками роботи вищих навчальних закладів, які здійснюють підготовку фахівців з програмної інженерії.

Так, у Листі Міністерства освіти України „Щодо покращення якості підготовки фахівців для ІТ-галузі” [1] зауважено необхідність перегляду змісту нормативних дисциплін, які викладаються під час підготовки фахівців у відповідності до сучасних досягнень розвитку інформаційних технологій, а також включити до навчальних планів дисципліни економічного та юридичного

спрямувань, а також впровадити поглиблене вивчення англійської мови за професійним спрямуванням.

Аналіз досліджень і публікацій. Аналіз наукової літератури з теми дослідження засвідчив, що в Україні аспекти проблеми підготовки інженерів у галузі інформаційно-комунікаційних технологій в Україні та у світі досліджували вітчизняні науковці: Г. Козлакова, Т. Морозова, З. Сейдаметова, С. Семеріков, А. Стрюк, М. Стрюк та інші. Дослідженням питань професійної підготовки фахівців за кордоном займалися вітчизняні науковці Н. Бідюк, В. Коваленко, Т. Кошманова, К. Корсак, Н. Пацевко та інші. Проте науковцями не було детально вивчено досвід провідних країн світу щодо організації професійної підготовки фахівців з програмної інженерії.

Формулювання цілей статті. Відповідно, метою статті є вивчення досвіду США щодо організації професійної підготовки фахівців з програмної інженерії, дослідження структури, змісту та особливостей навчальних планів провідних американських закладів освіти з метою адаптації та використання у педагогічній практиці університетів України.

Виклад основного матеріалу. Професія інженера-програміста є однією з популярних та затребуваних у сучасному світі. Про цей факт свідчать численні рейтинги та топ-листи. У рейтингу Forbes топ-професій для претендентів зі ступенем бакалавра, складеному агенцією CareerBuilder разом з компанією Economic Modeling Specialists Intl, [3] на другому місці за затребуваністю є розробник програмного забезпечення, на п'ятому – системний адміністратор, на шостому – веб-розробник.

Автор багатьох книг щодо працевлаштування Л. Кренц та головний редактор та генеральний менеджер журналу CareerJournal.com, Т. Лі у доповіді Jobs Rated Report 2015 щодо топ-професій [6] помістили професію розробника програмного забезпечення на восьме місце, аналітика комп'ютерних систем – на десяте, адміністратора мереж та комп'ютерних систем на тринадцяте місце. Місця у рейтингу розподілялися у відповідності до наступних критеріїв: прибуток, робоче середовище, стрес, перспективи працевлаштування. Метою

створення подібного документу була допомога учням старшої та середньої школи зробити розумний вибір щодо їх майбутньої професійної кар'єри.

Відображенням зростаючого попиту на фахівців з програмної інженерії є зростання чисельності закладів освіти, які здійснюють підготовку, та їх конкуренції. В Академічному рейтингу університетів світу у 2015 році (Шанхайському рейтингу) з інформатики [2] представлено 63 університети з США, 45 – з Китаю, 16 – з Великої Британії, 10 – з Австралії, 9 – з Іспанії, 7 – з Канади, 9 – з Ізраїлю, 5 – з Південної Кореї, 4 – з Італії, по 3 – з Саудівської Аравії та Бельгії, по 2 з Франції, Німеччини, Швейцарії, Бразилії, Австрії, Сінгапуру, Данії та Ірану, по 1 з Японії, Швеції та Португалії. Важливо відмітити, що перші 10 місць у рейтингу посідають університети США, з них на перших трьох позиціях знаходяться: 1) Стенфордський університет; 2) Массачусетський технологічний університет; 3) Каліфорнійський університет в Берклі.

Високий рейтинг американських університетів у світових рейтингах, безумовно, призводить до інтересу науковців та педагогів-практиків щодо вивчення організації самої підготовки, а також змісту та організації навчання.

Програми підготовки розробників програмного забезпечення (фахівців з програмної інженерії) у США пропонуються на рівнях асоціату (Associate's Degree in Software Engineering), бакалаврату (Bachelor's Degree in Software Engineering) і магістратури (Master's Degree in Software Engineering). Кількість університетів, які здійснюють підготовку фахівців в галузі інформаційних технологій, є близькою до 220.

Програмна інженерія (Software Engineering) визначається як «систематичне використання наукових та технологічних знань, методів та досвіду для проектування, впровадження, тестування та документування програмного забезпечення» [4]. Більш сучасним визначенням програмної інженерії можна вважати трактування програмної інженерії як «використання систематичного, дисциплінованого, кількісно вимірюваного підходу до

розвитку, функціонування та підтримки програмного забезпечення, тобто використання інженерії до програмного забезпечення» [5].

Найбільш сучасним документом, регламентуючим створення навчальних планів з програмної інженерії, є рекомендації для розробки навчальних планів підготовки бакалаврів та асоціатів з програмної інженерії, розроблені Міжнародними організаціями Association for Computing Machinery (ACM) та IEEE Computer Society (IEEE CS) у рамках Навчального плану з комп'ютеризації (Computing Curricula 2014) [7], у яких визначено структуру та зміст обсягу знань, виділено ядро базових знань. Рекомендації є логічним продовженням та адаптацією вже існуючих рекомендацій 2004 та 2009 років та служать основою освітніх стандартів підготовки фахівців з програмної інженерії у країнах світу.

Розробники документу обґрунтовують свій вибір структури та змісту навчального плану тим фактом, що програмна інженерія зазнала певної еволюції та кардинальних змін з початку свого існування. Автори підкреслюють, що підготовка фахівців з програмної інженерії має охоплювати інформатику, інженерію, математику та статистику, психологію та соціальні науки, менеджмент, та включати професійну практику та кодекс честі.

Доцільно розглянути, як у рекомендаціях визначено результати для студентів, які вивчають програмну інженерію. Студенти мають бути спроможними продемонструвати наступні якості:

- професійні знання (володіння знаннями програмної інженерії та навичками, а також професійними стандартами, необхідними для початку діяльності в якості програмного інженера);
- технічні знання (розуміння та використання відповідних теорій, моделей та технологій для визначення проблеми та її аналізу, проектування програмного забезпечення, його розробки, впровадження, перевірки та документування);
- робота у команді (робота індивідуально та у команді для розробки та впровадження якісних програмних продуктів);

- знайомство з кінцевим користувачем (розуміння важливості переговорів, звичок ефективної роботи, лідерства та комунікації з зацікавленими особами у типовому середовищі розробки програмного забезпечення);
- вміння йти на компроміс (приведення до відповідності конфліктних цілей проекту, знаходження компромісів в обмеженнях вартості, часу, знань, існуючих систем та організацій);
- продовження професійного розвитку (вивчення нових моделей, технологій та розуміння необхідності продовження професійного розвитку).

Як можна побачити, у рекомендаціях приділяється значна увага не лише професійним і технічним знанням студентів, а й міжособистісним відносинам, комунікації, прийняттю рішень.

В основу рекомендацій було покладено загальні принципи комп'ютерної науки, а також спеціальні принципи, які відображають специфіку програмної інженерії:

1. Програмна інженерія у спектрі комп'ютерної науки. Рекомендації стосуються саме програмної інженерії, але посилаються на інші комп'ютерні дисципліни, а також пропонують шляхи впровадження в інші дисципліни.
2. Довідкові дисципліни. Студент повинен вивчати не лише комп'ютерні дисципліни, а також інші, як математику, інженерію, управління проектами, як теоретично, так і практично.
3. Еволюція навчальних планів. У зв'язку з швидким прогресом програмної інженерії компоненти навчального плану можуть оновлюватися та удосконалюватися.
4. Організація навчальних планів. Моделі навчального плану komponують елементи знань у прості навчальні блоки, що робить їх простими для впровадження для педагогів, а також для видавців підручників.
5. Ядро програмної інженерії. Рекомендації визначають загальні теми дисципліни, навички і знання, якими мають володіти всі студенти.
6. Включення знань програмної інженерії. Опис ключових знань має бути стислим та відповідним, повинен бути визначений ключовий набір тем для всіх

ступенів. Вивчення має починатися з вступних курсів, продовжуватися протягом навчання та доповнюватися додатковими курсами, які можуть бути різними у залежності від закладу, програми або окремого студента.

Розглянемо структуру ядра знань, яке у рекомендаціях визначене як освітні знання з програмної інженерії (Software Engineering Education Knowledge, SEEK). Знання організовані ієрархічно у трьох рівнях:

- галузь знань (вищий рівень);
- блок/модуль (середній рівень);
- тема (найнижчий рівень).

SEEK надає базу для розробки, впровадження та надання освітніх блоків, які утворюють навчальний план. Організація та зміст галузей знань та освітніх блоків, їх порядок у навчальному плані можуть бути довільними. Навчальне навантаження, презентоване у SEEK, у США складає 25 відсотків від загального навчального плану, в інший час може вивчатися матеріал з спеціальної програми навчання студентів.

У таблиці 1 представлені галузі знань, які утворюють SEEK, та відповідні освітні блоки з рекомендованою кількістю годин.

Таблиця 1

Галузі знань з програмної інженерії та освітні блоки

Назва	Години	Назва	Години
Основи інформатики	152	Дизайн програмного забезпечення	48
Основи інформатики	120	Принципи дизайну	3
Конструкційні технології	20	Стратегії дизайну	6
Конструкційні інструменти	12	Архітектурний дизайн	12
		Дизайн взаємодії «людина-комп'ютер»	10
		Робочий проект	14
		Оцінювання дизайну	3
Основи математики та інженерії	80	Верифікація та перевірка правильності програмного забезпечення	37
Основи математики	50	Термінологія та основи верифікації та перевірки правильності програмного забезпечення	5
Основи інженерії для програмного забезпечення	22	Експертиза та статистичний аналіз	9
Інженерна економіка для	8	Тестування	18

програмного забезпечення			
		Аналіз проблем та повідомлення про них	5
Професійна практика	29	Процес програмного забезпечення	33
Групова динаміка та психологія	8	Принципи процесів	3
Комунікативні навички (специфічні для програмної інженерії)	15	Реалізація процесів	8
Професіоналізм	6	Планування та відслідкування проєктів	8
		Управління конфігурацією програмного забезпечення	6
		Еволюційні процеси та діяльність	8
Моделювання та аналіз програмного забезпечення	28	Якість програмного забезпечення	10
Основи моделювання	8	Принципи та культура якості програмного забезпечення	2
Типи моделей	12	Гарантування процесу	4
Основи аналізу	8	Гарантування продукту	4
Аналіз та специфікація технічних вимог	30	Безпека	20
Основи технічних вимог	6	Основи безпеки	4
Виявлення технічних вимог.	10	Комп'ютерна та мережева безпека	
Специфікація та документація технічних вимог	10	Розробка безпечного програмного забезпечення	8
Перевірка правильності технічних вимог	4		8

Автори документу наголошують, що при розробці навчальних планів та індивідуальних курсів з програмної інженерії на основі SEEK необхідно враховувати як методику викладання програмної інженерії, так і змістове наповнення, та надають наступні рекомендації:

1. Розробники навчальних планів та викладачі програмної інженерії повинні мати достатні відповідні знання та досвід, а також розуміння характеру програмної інженерії.

2. Розробники та викладачі мають думати з точки зору навчальних результатів.

3. Розробники мають дотримуватися балансу щодо вивчення матеріалів та гнучкістю щодо інновацій.

4. Багато понять, принципів і проблем програмної інженерії мають викладатися в якості тем, які періодично повторюються у навчальному плані, для допомоги студентам у розвитку їх програмного інженерного образу

мислення.

5. Вивчення тем програмної інженерії має відбуватися від простої до складної, тобто теми, які вимагають певної зрілості, повинні вивчатися наприкінці семестру.

6. Студенти мають розвивати розуміння області використання програмного забезпечення.

7. Програмна інженерія повинна викладатися способами, які визнають її як комп'ютерною, так і інженерною дисципліною.

8. Студентів потрібно навчати певним особистим навичкам, які знаходяться за межами предмету (критичне судження, визнання власних обмежень, ефективна комунікація тощо).

9. Студенти повинні розвивати оцінювання важливості продовження навчання та навичок для самостійного навчання.

10. Вирішення проблем програмної інженерії має викладатися як багатовимірне.

11. Треба акцентуватися на основні принципи програмної інженерії, а не на деталях останніх або специфічних інструментів.

12. Навчання повинно відбуватися таким чином, щоб студенти набували досвіду, використовуючи відповідні і сучасні інструменти, навіть коли подробиці інструментів не є фокусом навчання.

13. Матеріал, який викладається у курсі програмної інженерії, має ґрунтуватися на емпіричних дослідженнях і математичній або науковій теорії або широко прийнятій практиці.

14. Навчальний план повинен мати значну реальну основу (вивчення кейсів, проектна діяльність, практичний досвід, робочий досвід студентів).

15. Часто повинні розглядатися етичні, правові та економічні concerns, а також питання важливості бути професіоналом.

16. Процес розробки програмного забезпечення повинен бути центром організації навчального плану та центральним у розумінні студентами практики програмної інженерії.

17. Необхідно мотивувати студентів, використовуючи цікаві, конкретні та переконливі приклади.

18. Викладання програмної інженерії має рухатися за межі формату лекцій та використовувати численні підходи навчання та викладання.

19. Важливу ефективність та синергію можна досягнути, розробляючи навчальні плани з вивченням кількох типів знань одночасно.

20. Курси і навчальні плани мають регулярно переглядатися та оновлюватися.

Як вже було вказано, рекомендації можуть бути адаптовані для складання навчальних планів з дисципліни та окремих курсів у залежності від навчального закладу, навчальної програми або студентів.

Зміст освітніх програм підготовки асоціатів, бакалаврів та магістрів з програмної інженерії було досліджено на прикладі американських університетів: Рочестерського технологічного інституту (Rochester Institute of Technology) та Державного університету Сан Хосе (José State University), які входять до 50 найкращих університетів США за національним рейтингом (таблиця 2).

Таблиця 2

Порівняння навчальних планів з програмної інженерії у Рочестерському технологічному інституті та Державному університеті Сан Хосе (США)

Університет					
Rochester Institute of Technology		Rochester Institute of Technology		San José State University	
Ступінь					
Бакалавр наук (BS in SE)		Бакалавр наук / Магістр наук (BS/MS) (прискорена подвійна програма)		Бакалавр наук (BS in SE)	
1 рік навчання					
Дисципліна	кількість кредитів	Дисципліна	кількість кредитів	Дисципліна	кількість кредитів
Інформатика	8	Вступ до обчислювального вирішення проблем	4	Вступ до інженерії	3
Гуманітарні науки та науки, перспектива 7А, 7В: проектні	8	Обчислювальне вирішення проблем зі структурами	4	Математика, обчислення	3

числення I, II					
Гуманітарні науки та науки, основи 1: семінар	3	Гуманітарні науки та науки, перспектива 7А, 7В: проектні числення I, II	8	Комунікації	3
Гуманітарні науки та науки, перспектива 1	3	Гуманітарні науки та науки, основи 1: семінар	3	Англійська мова, твір I	3
Семінар першокурсників	1	Гуманітарні науки та науки, перспектива 1	3	Вступ до програмування	4
Дискретна математика для обчислювальної техніки	3	Семінар з програмної інженерії	1	Дискретна математика	3
Особиста програмна інженерія	3	Дискретна математика	3	Біологія, живий світ	3
Гуманітарні науки та науки, основи 2: письмо		Особиста програмна інженерія	3	Англійська мова, твір II	3
Університетський досвід	0	Гуманітарні науки та науки, основи 2: письмо	3		
Валеологічна освіта	0	Університетський досвід	0		
		Валеологічна освіта	0		
2 рік навчання					
Гуманітарні науки та науки, перспектива 5, 6: Фізика I, II	8	Гуманітарні науки та науки, перспектива 5, 6: Фізика I, II	8	Математика, обчислення II	4
Вступ до теорії інформатики	3	Вступ до теорії інформатики	3	Вступ до структур даних	4
Професійна комунікація для інженерів-програмістів	3	Професійна комунікація для інженерів-програмістів	3	Країнознавство (Америка)	6
Вступ до програмної інженерії	3	Вступ до програмної інженерії	3	Математика, обчислення III	3
Гуманітарні науки та науки, перспектива 2	6	Гуманітарні науки та науки, перспектива 2	3	Фізика, механіка	4
Прикладна статистика	3	Прикладна статистика	3	Країнознавство (Америка)	6
Математичні моделі програмної інженерії	3	Математичні моделі програмної інженерії	3		
Інженерія програмних підсистем	3	Інженерія програмних підсистем	3		
3 рік навчання					
Програмування і керування проектами	3	Програмування і керування проектами	3	Організація та архітектура комп'ютерів	3
Розробка безпечних програмних систем	3	Розробка безпечних програмних систем	3	Програмна інженерія I	3
Розробка проекту (вибірково)	3	Розробка проекту (вибірково)	3	Структури даних та алгоритми	3
Інженерні основи комп'ютерних систем	4	Інженерні основи комп'ютерних систем	4	Електричність та магнетизм	4

Гуманітарні науки та науки, перспектива 4	3	Гуманітарні науки та науки, перспектива 4	3	Програмна інженерія II	3
		Спільне навчання		Об'єктно-орієнтоване проектування	3
				Диференціальні рівняння та лінійна алгебра	3
				Інженерія, інженерна доповідь	3
4 рік навчання					
Технічні вимоги та архітектура програмних систем	3	Технічні вимоги та архітектура програмних систем	3	Основи вбудованих систем	3
Технічні вимоги та підтримка	3	Технічні вимоги та дизайн	3	Вступ то баз даних	3
Вибіркова дисципліна з програмування	3	Розробка технологічного процесу	3	Інформаційна безпека	3
Вибіркова дисципліна з математики / науки	3	Вибіркова дисципліна з математики / науки	3	Прикладна статистика	4
Вибіркова дисципліна з галузей використання	3	Вибіркова дисципліна з галузей використання	3	Керування процесом розробки програмного забезпечення	3
Гуманітарні науки та науки, занурення 1	3	Гуманітарні науки та науки, занурення 1	3	Комп'ютерні мережі	3
		Спільне навчання		Операційні системи	3
				Тестування якості програмного забезпечення	3
5 рік навчання					
Проект з програмної інженерії I, II	6	Проект з програмної інженерії I, II	6	Огляд програмного забезпечення підприємства	3
Вибіркова дисципліна з інженерії	6	Моделювання програмного забезпечення	3	Взаємодія людини і комп'ютера	3
Вибіркова дисципліна з галузей використання	6	Вибіркова дисципліна з інженерії	6	Проект дизайну I	3
Гуманітарні науки та науки, занурення 2, 3	6	Вибіркова дисципліна з галузей використання	6	Інженерія	3
Вільна вибіркова дисципліна	6	Гуманітарні науки та науки, занурення 2, 3	6	Вибіркова технічна дисципліна	3
	128	Вільна вибіркова дисципліна	6	Вибіркова технічна дисципліна	3
		Випускна вибіркова дисципліна	6	Проект дизайну I	3
				Інженерія	3
					120
6 рік навчання					
		Дисертація	6		
		Якість програмного забезпечення	3		

		Архітектури програмного забезпечення та лінії продуктів	3		
		Еволюція та модернізація програмного забезпечення	3		
		Випускна вибіркова дисципліна	3		
			158		

Як можна побачити, структура навчальних планів університетів США, які здійснюють підготовку фахівців з програмної інженерії, є подібною: навчальні плани поєднують загальнонаукову підготовку та професійну підготовку. Загальнонаукова підготовка включає гуманітарні дисципліни (liberal arts and sciences), математичні та природничі дисципліни (natural sciences), валеологію (wellness education) та вступні фахові дисципліни. Вивчення загальнонаукових дисциплін відбувається переважно у перші два роки навчання.

Професійна підготовка фахівців з програмної інженерії включає обов'язкові курси (required courses), курси за вибором студента (electives), а також вільні курси (free courses). Навчальні плани також включають спільне навчання, практику, дослідницьку роботу, валеологію.

Висновки. Детальне вивчення законодавчої та нормативної бази, науково-методичної літератури, інформаційних джерел та навчальних планів університетів США дозволило проаналізувати зміст та особливості організації професійної підготовки фахівців з програмної інженерії в університетах США. Наше дослідження виявило, що висока якість та ефективність професійної підготовки фахівців з програмної інженерії в університетах США, безумовно, потребують системного вивчення, адаптації та використання у підготовці інженерів-програмістів в Україні.

До перспективних напрямів досліджень у даній сфері вважаємо за доцільне віднести такі проблеми: подальше вивчення та системний аналіз професійної підготовки фахівців з програмної інженерії провідних країн світу з метою удосконалення вітчизняної підготовки інженерів-програмістів.

Список використаних джерел і літератури/References:

1. Лист МОНмолодьспорт „Щодо покращення якості підготовки фахівців для ІТ-галузі” від 16.02.2012 р. N 1/9-119 [Електронний ресурс]. – Режим доступу : http://osvita.ua/legislation/Vishya_osvita/27674/. – Назва з екрану.
2. Academic Ranking of World Universities in Computer Science – 2015 [Електронний ресурс]. – Режим доступу: <http://www.shanghairanking.com/SubjectCS2015.html>. – Назва з екрану.
3. Dill K. 10 Jobs In High Demand That Require A College Degree [Електронний ресурс]. – Режим доступу: <http://www.forbes.com/sites/kathryndill/2014/12/01/10-jobs-in-high-demand-that-require-a-bachelors-degree/#41264bec2adb>. – Назва з екрану.
4. ISO/IEC 2382-1:1993 Information technology – Vocabulary – Part 1: Fundamental terms [Електронний ресурс]. – Режим доступу: http://www.iso.org/iso/catalogue_ics. – Назва з екрану.
5. ISO/IEC/IEEE 24765:2010 Systems and software engineering – Vocabulary [Електронний ресурс]. – Режим доступу: http://www.iso.org/iso/catalogue_ics. – Назва з екрану.
6. Lee T. Jobs Rated Report 2015: Ranking The Top 200 Jobs [Електронний ресурс]. – Режим доступу: <http://www.careercast.com/jobs-rated/jobs-rated-report-2015-ranking-top-200-jobs>. - Назва з екрану.
7. Software Engineering 2014. Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. A Volume of the Computing Curricula Series. Joint Task Force on Computing Curricula IEEE Computer Society, Association for Computing Machinery.- IEEE Computer Society, Association for Computing Machinery, 2015. – 134 p.