



**СИСТЕМНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
КУРСОВА РОБОТА
К (Силабус)**

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалавр)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Освітня програма	<i>Комп'ютерні системи та мережі</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна(денна)/заочна</i>
Рік підготовки, семестр	<i>4 курс, 7 семестр</i>
Обсяг дисципліни	<i>1 кредит (30 годин)</i>
Семестровий контроль/ контрольні заходи	<i>курсова робота / захист курсової роботи</i>
Розклад занять	
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Доктор техн. наук, проф. Сімоненко В.П., svp@comsys.kpi.ua</i>
Розміщення курсу	<i>Системне програмне забезпечення: Методичні вказівки до виконання курсової роботи для студентів напряму підготовки "Комп'ютерна інженерія" / бібліотека - Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського".</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Метою вивчення дисципліни є формування у студентів здатностей:

Метою кредитного модуля є формування у студентів здатностей:

- виконувати аналітичний огляд за темою КР
- аналізувати особливості архітектури багато процесорних систем
- вивчити алгоритми статичного планування для багато процесорних систем
- розробляти та налагоджувати програму статичного планування
- проводити експериментальні дослідження ефективності програми, що розробляється

Основні завдання при вивченні дисципліни

Згідно з вимогами освітньо-професійної програми студенти після засвоєння дисципліни "Системне програмне забезпечення" мають продемонструвати такі результати навчання:

Знати:

- основні принципи організації, планування та керування обчислювальними процесами в обчислювальних системах, комплексах та мережах
- принципи проектування системного програмного забезпечення. Методи будовання та

організацію систем переривання

- участь системних програм в організації обчислювального процесу обчислювальної системи при проходженні завдань
- знати структуру і організацію системного програмного забезпечення, що керує даними, пам'яттю, пристроями введення/виведення

Вміти:

- формувати завдання на роботу ЕОМ
- оперувати наборами даних на різних рівнях ієрархічної системи керування даними, програмувати обмін інформації у ЕОМ з використанням стандартних засобів обміну та власними драйверами обміну
- проектувати елементи системного програмного забезпечення, що виконує спеціальні функції обробки системної інформації при виконанні задач планування та розподілу завдань в обчислювальному середовищі
- підключати додаткові функції в операційну систему

Досвід:

- роботи у різних операційних системах
- інсталяції операційної системи, модифікації операційної системи

Написання курсової роботи сприяє формуванню творчого мислення студента, проведенню аналізу та вибору потрібних джерел й літератури по вибраній темі, вмінню формулювати висновки.

Кредитний модуль забезпечує наступні компетентності і програмні результати освітньо-практичної програми першого рівня вищої освіти: ФК17, ПРН16.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Вивчення дисципліни “Системне програмне забезпечення” дозволяє сформувати у студентів компетенції, необхідні для розв’язання практичних задач професійної діяльності, пов’язаної з аналізом та використанням сучасних інформаційних технологій із створення системного програмного забезпечення для сучасних операційних систем.

Дисципліна “Системне програмне забезпечення” ґрунтується на вивчення таких кредитних модулів “Програмування”, “Системне програмування”, “Архітектура КС”.

Для успішного вивчення курсу СПЗ необхідні знання про мови програмування різного рівню, навички роботи на КС та вміння працювати в різних середовищах програмування.

3 Зміст навчальної дисципліни

КР включає три розділи. Розділ перший пов’язаний з виконанням аналітичного огляду апаратних або програмних засобів побудови сучасних ПКС.

Два інших розділи пов’язані з розробкою програм для ПКС для заданої архітектури. Для задачі необхідне виконати аналіз сучасних алгоритмів статичного планування.

В розділі 2 виконується розробка програми розподілу задач згідно варіанту. Варіант КР визначає архітектуру ПКС.

Необхідно:

- проаналізувати особливості запропонованої архітектури багатопроцесорної системи
- розробити алгоритм статичного планування
- розробити та налагодити програму
- провести експериментальні дослідження ефективності програми

Важливою складовою розділів 2 і 3 є тестування програм зі змінними вихідними даними. Необхідно дослідити ефективність виконання програми і порівняти результати з зоною пошуку оптимального варіанту, яка обчислюється заздалегідь. Для цього визначається час виконання програми, розраховуються значення коефіцієнтів прискорення та ефективності (завантаження), які

відображаються за допомогою графіку Ганта.

Кожен розділ завершується висновками. Також КР завершується висновком та описом основних результатів роботи. В додатках наводяться лістинг програми, схеми алгоритмів.

Графік виконання курсової роботи

Табл. 3.1

Тиждень семестру	Назва етапу роботи	Навчальний час СРС
1	<i>Отримання теми та завдання</i>	1
2-5	<i>Підбор та вивчення літератури</i>	4
6-7	<i>Виконання розділу 1</i>	2
7-8	<i>Виконання розділу 2</i>	2
8-9	<i>Виконання розділу 3</i>	4
11-15	<i>Проведення експериментальних досліджень</i>	6
16	<i>Подання курсової роботи на перевірку</i>	
17	<i>Захист курсової роботи</i>	1
		20

Захист курсової роботи проводиться у формі співбесіди зі з'ясуванням усіх питань, що виникли у керівника під час перевірки роботи.

За результатами захисту, у відповідності до критеріїв оцінювання, що наведені у підрозділі 7, визнається оцінка.

На оцінку за курсову роботу впливають:

- якість розробленого програмного забезпечення
- якість розробленої програмної документації
- компетентність та загальна ерудиція студента при відповідях на запитання під час захисту
- ступінь виконання графіку підготовки курсової роботи

Якщо студент подав на захист не самостійно виконану роботу, курсова робота до захисту не допускається, що супроводжується записом “не допущений” у екзаменаційній відомості. Такий самий запис робиться у випадку, якщо курсова робота не завершена на час захисту. В цих випадках запис “не допущений” еквівалентний отриманню оцінки “незадовільно”.

4 Навчальні матеріали та ресурси

Базова література

1. Modern Operating Systems, 4th edition. Tanenbaum.
2. Understanding the Linux Kernel. Daniel P. Bovet, Marco Cesati.
3. Solaris Internals. Richard McDougall, Jim Mauro.
4. HP-UX 11i Internals. Chris Cooper, Chris Moore.
5. Mac OS X Internals: A Systems Approach. Amit Singh.

Додаткова література

- [1] Adam T.L., Chandy K.M. and Dickson J.R., A comparison of List Scheduling for Parallel processing systems, Comm. ACM, Dec. 1974, pp685-690.
- [2] Baker T.P., Stack Based Scheduling of Real-Time Processes, Journal on Real-Time Systems, Vol 3, N1, March 1991, pp67-99.
- [3] Brendan Tangney, Donal O'Mahony, "Local Area Networks and Their Application", "Prentice Hall", New York, London, Toronto, Sydney, Tokyo, 1988
- [4] Berge C., Two Theorems in Graph Theory, Proc. National Acad. of Science USA, 43 (1957), 842-844.

- [5] Blazevicz J., Drozdowski M., G. Schmidt, and D. De Werra, Scheduling independent multiprocessor tasks on a uniform k-processor system, *Parallel Computer* 20(1994),pp15-28.
- [6] Blazevicz J., Drozdowski M., and J. Weglarz, Scheduling multiprocessor tasks to minimize schedule length, *IEEE Trans. Computer* C35(1986),pp389-393.
- [7] Bultan T. and Aykanat C. , “A new mapping heuristic based on mean field annealing”, *Journal of Parallel and Distributed Computing*, Vol. 16, n4, Dec 1992.
- [8] Brendan Tangney, Donal O’Mahony, "Local Area Networks and Their Application", "Prentice Hall", New York, London, Toronto, Sydney, Tokyo,1988
- [9] Butler R., Lusk E., Monitors, messages and clusters: The p4 parallel programming system, Technical Report Preprint MCS-P362-0493, Argone National Laboratory, Argone, 1993.
- [10] Carriero N., Gelernter D., LINDA in context., *Communication of the ACM* 32(4): 444-458, 1989.
- [11] Cheriyan J., Hagerup T. and Mehlhorn K., Can a Maximum Flow be Computed in $O(nm)$ time?, to be presented at 17th ICALP,1990.
- [12] Casavant Thomas L. and Kuhl John G., A taxonomy of Scheduling in General-Purpose Distributed Computing Systems, *IEEE Transactions on Software Engineering*, Vol. 14,N2,1988, pp.141-154.
- [13] Chang Y.C., Shin K.G., Optimal Load Sharing in Distributed Real-Time Systems, *Journal of Parallel and Distributed Computing*,Vol 19,1993,pp38-50.
- [14] Coffman E. and Graham R., Optimal Scheduling for two-processor systems, *Acta Information*, Vol. 1,1972.
- [15] Coli M. and Palazzari P. , Load Balancing with Internode Precedence Relations: a New Method for Static Allocation of DAGs into Parallel System. 1066-6192/96.
- [16] Digital:, Digital's Networrs: an architecture with a future, Digital Equipment Corporation, 1984 (order no. EB 26013-42).
- [17] Derigs U., Meier W., Implementing Goldberg's max-Flow-algorithm-A computational investigation. *Z.oper. Res. A.*-1989.-33,N 6.-p.383-403.
- [18] Darte A., Two heuristics for Task Scheduling, Lab LIP-IMAG Ecole Norm. Super de Lyon,1991.
- [19] Edmonds J., Karp R. M., Theoretical improvements in algorithmic efficiency for network flow problems, *J. ACM.*-Apr. 1972.-Vol. 19, N 2.-P. 248-264. (Combinatorial Structures and Their Applications. Gordon and Breach, New York, 1970, pp.93-96 (abstract presented at Calgary International Conference on Combinatorial Structures and Their Applications, June 1969)).
- [20] Eager D.L., Lazowska E.D., Zahorjan J., Adaptive Load Sharing in Homogeneous Distributed Systems, *IEEE Trans. on Software Engineering*, Vol. 12, (1986),pp662-675.
- [21] Efe K., “Heuristic models for task assignment scheduling in distributed systems”, *IEEE Computer*, June (1982)

- [22] Elsadek A. A. and Wells B.E., “Heuristic model for task allocation in a heterogeneous distributed systems”, *proc. of PDPTA '96*, Vol.2, (1996), pp659-671.
- [23] El-Rewini H., Lewis T.G., “Scheduling Parallel tasks onto Arbitrary Target Machines”, *Journal of Par. and Distr. Com.*, Vol.9,(1990),pp138-153.
- [24] Evans D.J., Butt W.U.N., Dynamic Load Ballancing Using Task-Transfer Probabilities, *Parallel Computing*, Vol 19,1993,pp897-916.
- [25] Feautrier Paul, Fine-grain Scheduling under Resource Constraints, *Inter. Journal of Parallel Programming*, Vol. 21, N 3, 1993, pp1-15.
- [26] Feautrier Paul, Some efficient solutions to the affine scheduling problem,II, multidementional time, *Int. Journal of Parallel Programming*, Vol. 21, N 6, December 1992, pp389-420.
- [27] Freund Richard F., Carter B.R., Watson Daniel, E. Keith, and F. Mirable, ”Generational Scheduling for Heterogeneous Computing Systems”, *proc. of PDPTA '96*, Vol.2, (1996), pp769-778.
- [28] Flover J., Kolawa A., Bharadwaj S., The Express way to distributed processing, *Supercomputing Review*, pp. 54-55, 1991.
- [29] Foster I., *Designing and Building Parallel Programs*, /book/ book/ ntm/1995.
- [30] Gasperoni Franco and Schwiegelshohn Uwe, Scheduling loops on parallel processors: a simple algorithm with close to optimum performance, *Parallel processing: CONPAR 92-VAPP V*, Springer,1992, pp 625-636.
- [31] Gelenbe E.,Kushwaha R., Dynamic Load Balancing in Distributed Systems, *Mascost'94 IEEE Computer Society Press*, pp.245-249, 1994.
- [32] Gerasoulis A., Yang T., On the granularity and clustering of directed acyclic task graphs, *IEEE Transactions on Parallel and Ditrubed Systems*, Vol 4, N. 6, June. 1993.
- [33] Gerasoulis A and. Yang T., Scheduling Programs Task Graph on MIMD Architectures, Report DCS, Rutgers Uni.,1991.
- [34] Gabow H. N., Scaling algorithms for network problems. *J. omput. Syst. Sci.* 31(1985), 148-168.
- [35] Heywood T., Ranka S., A practical Hierarchical model of parallel computation, *Journal of Parallel and Distributed Computing*, 16, pp. 212-232, 1992.
- [36] Herken R., *The universal Turing Machine: a half-century survey*, Oxford Press, 1988.
- [37] Hwang K., Briggs F. A., *Computer architecture and parallel processing*, McGraw-Hill Booc C., 1989.

Навчальний контент

5 Методика опанування навчальної дисципліни (освітнього компонента)

Політика навчальної дисципліни (освітнього компонента)

Як викладач, так і студент зобов'язані дотримуватись Кодексу честі інституту.

Основні положення політики:

- розділи курсової роботи повинні бути виконані згідно встановленого календарного графіку робіт
- студенти мають право оскаржити результати поточного контролю ходу виконання курсової роботи, аргументовано пояснивши з яким критерієм не погоджуються відповідно до зауважень
- у випадку виявлення факту академічної недоброчесності робота не зараховується

Календарний контроль

Календарний контроль студентів проводиться двічі на семестр, як моніторинг поточного стану виконання вимог силабусу. Умови позитивного календарного контролю:

- за результатами навчальної роботи на першому календарному контролі (8-й тиждень) студент отримує “атестований”, якщо він виконав обсяг робіт за перші 7 тижнів, згідно таблиці 3.1
- за результатами навчальної роботи на другому календарному контролі (14-й тиждень) студент отримує “атестований”, якщо він виконав обсяг робіт за 13 тижнів, згідно таблиці 3.1

Політика та контроль

6 Політика навчальної дисципліни (освітнього компонента)

Політика щодо академічної доброчесності:

Усі письмові роботи перевіряються на наявність плагіату і допускаються до захисту із коректними текстовими запозиченнями не більше 20%.

Поточний контроль

Поточний контроль виконання курсової роботи виконується шляхом перевірки розділів курсової роботи згідно графіку виконання курсової роботи, який надається у таблиці 3.1.

7 Види контролю та рейтингова система оцінювання результатів навчання (PCO) по дисципліні “Системне програмне забезпечення” – курсова робота

Виконання курсової роботи розглядається як окремий модуль

Протягом семестру студенти виконують 1 курсову роботу.

Максимальна кількість балів за курсову роботу: 100 балів.

Бали нараховуються за:

- Своєчасність подання роботи до захисту 0-20 балів,
- Оформлення пояснювальної записки до курсової роботи 0-20 балів,
- Виконання змістовного завдання на роботу 0-30 балів
- Захист (відповіді на запитання викладачів) 0-30 балів

Критерії оцінки якості захисту

Захист курсової роботи відбувається перед членами комісії. На захисті студент виступає з доповіддю за матеріалами КР та відповідає на питання.

Критерії оцінювання виступу з доповіддю за матеріалами КР та відповідей на питання:

- ступінь володіння теоретичним матеріалом до 10 балів;
- ступінь володіння кодом програми в цілому до 10 балів;
- вміння ввести зміни у програмний код до 10 балів

Захист курсових робіт відбуваються впродовж двох останніх тижнів семестру (без зниження балів), або під час сесії (із зниженням балів за захист на 50%).

Для отримання студентом відповідних оцінок (ECTS та традиційних) його рейтингова оцінка переводиться згідно з таблицею

Рейтинг	Оцінка ECTS	Традиційна оцінка
95...100	A - відмінно	Відмінно
85...94	B - дуже добре	Добре
75...84	C - добре	
65...74	D - задовільно	Задовільно
60...64	E - достатньо	
28...59	FX - незадовільно	Незадовільно
RD<28	F	Не допущений

Робочу програму курсової роботи (силабус):

Складено *Доктор техн. наук, проф. Сімоненко В.П.,*

Ухвалено кафедрою Обчислювальної техніки (протокол № 10 від 25.05.2022)

Погоджено Методичною комісією ФІОТ (протокол № 10 від 09.06.2022)

...