

**Методичні вказівки
до виконання курсової роботи
з «Програмування»
для студентів напряму підготовки
6.050201 – «Системна інженерія»**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Вінницький національний технічний університет

**Методичні вказівки
до виконання курсової роботи
з «Програмування»
для студентів напрямку 6.050201 – «Системна
інженерія»**

Вінниця
ВНТУ
2015

Рекомендовано до друку Методичною радою Вінницького національного технічного університету Міністерства освіти і науки України (протокол № 11 від 20.06.2013 р.)

Рецензенти:

О. І. Хома, доктор філософських наук, професор

І. Д. Похило, кандидат політичних наук, доцент

Методичні вказівки та плани практичних занять з «Політології» для студентів всіх напрямів підготовки та спеціальностей / Уклад. С. С. Плaxотнюк, В. О. Корнієнко. – Вінниця : ВНТУ, 2014. – 44 с.

У даних методичних вказівках та планах практичних занять наводяться основні рекомендації до вивчення, підготовки та проведення вищезгаданих практичних занять з дисципліни «Політологія» та організації самостійної роботи студентів.

ВСТУП

Курсова робота (КР) з дисципліни «Програмування» – це навчальна самостійна робота, яка охоплює весь матеріал, викладений під час вивчення дисципліни, і містить елементи (задачі) навчального, аналітично-розрахункового та науково-дослідницького характеру.

В курсовій роботі студент повинен розкрити зміст теми, показати знання літературних джерел і нормативних актів. Студент повинен показати знання мов програмування, розуміння основних концепцій сучасних технологій програмування, вміння самостійно розробити схему роботи програми в цілому та алгоритми складових поставленої задачі, підібрати засоби його програмної реалізації та подати розробку у вигляді, зручному для його використання сторонніми користувачами.

Зміст курсової роботи визначається завданням, яке видається на консультації викладачем кожному студенту. Завдання видається на першому тижні останнього навчального семестра. Курсове проектування включає декілька послідовних етапів, які, в загальному випадку, пов'язані зі змістовною постановкою задачі, розробкою індивідуального технічного завдання, вибором форми подання задачі, розробкою математичної моделі, вибором оптимального алгоритму реалізації задачі, проведенням досліджень режимів роботи програми та формулюванням обґрунтованих висновків щодо отриманих в роботі результатів. Кожен етап роботи обов'язково має знайти своє відображення в пояснювальній записці, що містить вхідні, вихідні та пояснювальні матеріали, які пов'язані з виконанням курсової роботи.

Завдання для курсових робіт визначаються викладачем із загального списку завдань на курсову роботу. Заохочуються пропозиції студентів щодо самостійного, за узгодженням з викладачем, вибору теми КР поза межами запропонованого в методичних вказівках переліку. Самостійний вибір предметної області, в якій доцільно використовувати сучасні методи програмування та оригінальні алгоритми, дозволяє зробити висновок щодо рівня творчої активності студента, його вміння самостійно здійснити попередній аналіз предметної області і розробити технічне завдання.

Програма, яка є результатом виконання курсової роботи, повинна бути повноцінним додатком операційної системи Windows, Android, iOS тощо. Розроблена програма обов'язково повинна мати такі складові:

1. Застосування основних принципів об'єктно-орієнтованого програмування.
2. Дотримання технології модульного програмування.
3. Реалізація дружнього інтерфейсу
4. Використання файлів для зберігання та зчитування інформації.
5. Перевірка цілісності даних.
6. Подання інформації.
7. Розробка супроводжувальної документації.

ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Загальні правила оформлення

При оформленні пояснювальної записки необхідно дотримуватись вимог до КР за ДСТУ 3008-95. Пояснювальна записка КР, з врахуванням вимог до нормативно-технічних документів, має подаватись на аркушах паперу формату А4 за вимогами ЄСКД. Текст пояснювальної записки (ПЗ) виконується у відповідності з вимогами ГОСТ 2.105-95 одним із застосовуваних друкувальних та графічних пристроїв виведення ЕОМ використавши шрифт Times New Roman (Сур), 14 пт, через один інтервал (ГОСТ 2.004-88). Пояснювальна записка відноситься до текстових документів, яка подається технічною мовою. Графічна інформація має подаватись у вигляді ілюстрацій (схеми, рисунки, графіки, діаграми тощо). Цифрова – у вигляді таблиць.

Відповідно до ДСТУ 3008-95 документацію оформляють на стандартних аркушах паперу з однієї сторони. Відступи від країв аркуша: зверху, знизу і зліва – 20 мм; справа – 10 мм. Абзац – 5 знаків. Шрифт та міжрядковий інтервал у додатках можуть бути довільними, але такими, щоб можна було прочитати і зрозуміти. Нумерація сторінок в правому верхньому кутку, починаючи зі змісту. Нумерація додатків продовжує основну нумерацію. Заголовки структурних частин, розділів виконують великими літерами посередині рядка, всі інші з абзацу малими літерами починаючи з великої. Слово «Додатки» малими літерами з першої великої посередині рядка.

Вимоги до оформлення розділів та підрозділів

Структурними елементами основної частини ПЗ є розділи, підрозділи, пункти, підпункти, переліки.

Розділ – головний ступінь поділу тексту, позначена номером і має заголовок.

Підрозділ – частина розділу, позначена номером і має заголовок.

Пункт – частина розділу чи підрозділу, позначена номером і може мати заголовок. Підпункт – частина пункту, позначена номером і може мати заголовок.

Заголовки структурних елементів необхідно нумерувати тільки арабськими числами.

Допускається розміщувати текст між заголовками розділу та підрозділу, між заголовками підрозділу та пункту.

Кожен розділ рекомендується починати з нової сторінки.

Заголовок розділу записують посередині (ДСТУ 3008-95) великими буквами з більш високою насиченістю.

Заголовки розділів, підрозділів, пунктів і підпунктів (за наявності заголовка) записують з абзацу малими буквами починаючи з великої.

Розділи нумерують порядковими номерами в межах всього документа (1, 2, і т.д.). Після номера крапку не ставлять, а пропускають один знак.

Підрозділи нумерують в межах кожного розділу, пункти в межах підрозділу і т.д. за формою (3.1, 3.2, 3.2.1, 3.2.2, 3.2.2.1 і т.д.).

Цифри, які вказують номер, не повинні виступати за абзац.

Посилання в тексті на розділи виконується за формою: «...наведено в розділі 3».

В тексті документа може наводитись перелік, який рекомендується нумерувати малими літерами української абетки з дужкою або тире перед текстом. Для подальшої деталізації переліку використовують арабські цифри з дужкою.

Кожну частину переліку записують з абзацу, починаючи з малої букви та закінчуючи крапкою з комою, в кінці останньої ставлять крапку.

Правила написання тексту

Під час написання тексту слід дотримуватися таких правил:

а) текст необхідно викладати обґрунтовано в лаконічному технічному стилі;

б) числа з розмірністю слід записувати цифрами, а без розмірності словами (*відстань – 2 мм, відміряти три рази*);

в) позначення одиниць слід писати в рядок з числовим значенням без перенесення в наступний рядок. Між останньою цифрою числа та позначенням одиниці слід робити пропуск (*100 Мб*);

г) якщо наводиться ряд числових значень однієї і тієї ж фізичної величини, то одиницю фізичної величини вказують тільки після останнього числового значення (*7,5; 1,75; 2 мм*);

д) позначення величин з граничними відхиленнями слід записувати так: *100 ± 5 мм*;

е) буквені позначення одиниць, які входять в добуток, розділяють крапкою на середній лінії (\square); знак ділення замінюють косою рисою (/);

ж) порядкові числівники слід записувати цифрами з відмінковими закінченнями (*9-й день, 4-а лінія*); у випадку кількох порядкових числівників відмінкове закінчення записують після останнього (*3, 4, 5-й графіки*); кількісні числівники записують без відмінкових закінчень (*на 20 аркушах*); не пишуть закінчення в датах (*21 жовтня*) та при римських числах (*XXI століття*);

з) скорочення слів в тексті не допускаються, крім загальноприйнятих в українській мові та установлених в ГОСТ 2.316-2008, а також скорочень, які прийняті для надписів на виробі (в тексті вони повинні бути виділені великими літерами: ON, OFF), а якщо надпис складається з цифр або знаків, то в лапках. Лапками також виділяють найменування команд, режимів, сигналів (*«Запуск»*);

к) дозволяється виконувати записи математичних виразів в одну стрічку:

л) не дозволяється:

- допускати професійних або місцевих слів і виразів (техніцизмів);
- після назви місяця писати слово «місяць» (не «в травні місяці», а «в травні»);
- використовувати вирази: «цього року», «минулого року», слід писати конкретну дату «в червні 2001 року»;
- використовувати позначення одиниць фізичних величин без цифр, необхідно писати повністю: “кілька кілограмів” (за винятком оформлення таблиць і формул);
- з’єднувати текст з умовним позначенням фізичних величин за допомогою математичних знаків (не «швидкість = 5 км/год», а «швидкість дорівнює 5 км/год», не «температура дорівнює -5° С», а «температура дорівнює мінус 5 °С»);
- використовувати математичні знаки $<$, $>$, №, %, sin, cos, tg, log та ін. без цифрових або буквених позначень. В тексті слід писати словами «нуль», «номер», «логарифм» і т.д.;
- використовувати індекси стандартів (ДСТУ, ГОСТ, СНІП, СТІ) без реєстраційного номера.

Оформлення формул

Кожну формулу записують з нового рядка, симетрично до тексту. Між формулою та текстом пропускають один рядок.

Умовні буквені позначення (символи) в формулі повинні відповідати установленим ГОСТ 1494-77. Їх пояснення наводять в тексті або зразу ж під формулою. Для цього після формули ставлять кому та записують пояснення до кожного символа з нового рядка в тій послідовності, в якій вони наведені у формулі, розділяючи крапкою з комою. Перший рядок повинен починатися з абзаца з слова «де» і без будь-якого знака після нього.

Всі формули нумерують в межах розділу арабськими числами. Номер вказують в круглих дужках з правої сторони, в кінці рядка, на рівні закінчення формули. Номер формули складається з номера розділу та порядкового номера формули в розділі, розділених крапкою. Дозволяється виконувати нумерацію в межах всього документа.

Числово підстановку та розрахунок виконують з нового рядка не нумеруючи. Одиницю вимірювання беруть в круглі дужки. Розмірність одного й того ж параметра в межах документа повинна бути однаковою.

Якщо формула велика, то її можна переносити в наступні рядки. Перенесення виконують тільки математичними знаками, повторюючи знак на початку наступного рядка.

Формула є частиною речення, тому до неї застосовують такі ж правила граматики, як і до інших членів речення. Якщо формула знаходиться в кінці

речення, то після неї ставлять крапку. Формули, які йдуть одна за одною і не розділені текстом, відокремлюють комою.

Посилання на формули в тексті дають в круглих дужках за формою: «...у формулі (5.2)»; «... у формулах (5.7, ..., 5.10)».

Оформлення ілюстрацій

Для пояснення викладеного тексту рекомендується його ілюструвати графіками, кресленнями, фрагментами схем та ін.

Розміщують ілюстрації в тексті або в додатках.

В тексті ілюстрацію розміщують симетрично до тексту після першого посилання на неї або на наступній сторінці, якщо на даній вона не уміщується без повороту.

На всі ілюстрації в тексті ПЗ мають бути посилання. Посилання виконують за формою: «...показано на рисунку 3.1.» або в дужках за текстом (рисунком 3.1), на частину ілюстрації: «...показані на рисунку 3.2,б». Посилання на раніше наведені ілюстрації дають зі скороченим словом «дивись» відповідно в дужках (див. рисунок 1.3).

Наведена форма запису (рисунок ...) відповідає вимогам ГОСТ 2.105-95, ДСТУ 3008-95 допускає скорочення, тобто замість «Рисунок ...» – «Рис ...».

Між ілюстрацією та текстом пропускають один рядок (3 інтервали).

Всі ілюстрації в ПЗ називають рисунками та позначають під ілюстрацією симетрично до неї за такою формою: «Рисунок 3.5 – Найменування рисунка». Крапку в кінці не ставлять, знак переносу не використовують. Якщо найменування рисунка довге, то його продовжують у наступному рядку починаючи від найменування.

Нумерують ілюстрації в межах розділів, вказуючи номер розділу та порядковий номер ілюстрації в розділі розділяючи їх крапкою. Дозволяється нумерувати в межах всього документа.

Пояснюючі дані розміщують під ілюстрацією над її позначенням.

У випадку, коли ілюстрація складається з частин або більше, їх позначають малими буквами українського алфавіту з дужкою (а), б,) під відповідною частиною. В такому випадку після найменування ілюстрації ставлять двокрапку та дають найменування кожної частини за формою:

а) – найменування першої частини; б) – найменування другої частини або за ходом найменування ілюстрації, беручи букви в дужки:

Рисунок 3.2 – Структурна схема (а) і часові діаграми (б) роботи фазометра

Якщо частини ілюстрації не вміщуються на одній сторінці, то їх переносять на наступні сторінки. В цьому випадку, під початком ілюстрації вказують повне її позначення, а під її продовженнями позначають «Рисунок (продовження)». Пояснюючі дані розміщують під кожною частиною ілюстрації.

Якщо в тексті є посилання на складові частини зображеного засобу, то на відповідній ілюстрації вказують їх порядкові номери в межах ілюстрації.

Якщо ілюстрація є фрагментом повної розробленої схеми, то для всіх компонентів вказують ті позиційні позначення, які вказані на схемі.

2.1.5 Оформлення таблиць

Таблицю розміщують симетрично до тексту після першого посилання на даній сторінці або на наступній, якщо на даній вона не уміщується і таким чином, щоб зручно було її розглядати без повороту або з поворотом на кут 90° за годинниковою стрілкою.

На всі таблиці мають бути посилання за формою: «*наведено в таблиці 3.1*»; «... *в таблицях 3.1 – 3.5*» або в дужках по тексту (*таблиця 3.6*). Посилання на раніше наведену таблицю дають зі скороченим словом «*дивись*» (*див. таблицю 2.4*) за ходом чи в кінці речення.

Таблицю розділяють на графи (колонки) та рядки. В верхній частині розміщують головку таблиці, в якій вказують найменування граф. Діагональне ділення головки таблиці не допускається. Ліву графу (боковик) часто використовують для найменування рядків. Допускається нерозділяти рядки горизонтальними лініями. Мінімальний розмір між основами рядків 8 мм. Розміри таблиці визначаються об'ємом матеріалу.

Графу «№ п/п» в таблицю не включають. За необхідності нумерації, номери вказують в боковику таблиці перед найменуванням рядка.

Найменування граф може складатися з заголовків і підзаголовків, які записують в однині, симетрично до тексту графи малими буквами, починаючи з великої. Якщо підзаголовок складає одне речення з заголовком, то в цьому випадку його починають з малої букви. В кінці заголовків та підзаголовків граф таблиці крапку не ставлять. Дозволяється заголовки та підзаголовки граф таблиці виконувати через один інтервал.

Якщо всі параметри величин, які наведені в таблиці, мають одну й ту саму одиницю фізичної величини, то над таблицею розміщують її скорочене позначення (*мм*). Якщо ж параметри мають різні одиниці фізичних величин, то позначення одиниць записують в заголовках граф після коми (*Довжина, мм*).

Текст заголовків і підзаголовків граф може бути замінений буквеними позначеннями, якщо тільки вони пояснені в попередньому тексті чи на ілюстраціях (*D – діаметр, H – висота і т.д.*). Однакові буквені позначення групують послідовно в порядку росту їх індексів, наприклад: (*L1, L2, ...*).

Найменування рядків записують в боковику таблиці у вигляді заголовків в називному відмінку однини, малими буквами, починаючи з великої і з однієї позиції. В кінці заголовків крапку не ставлять. Позначення одиниць фізичних величин вказують в заголовках після коми.

Для опису визначеного інтервалу значень в найменуваннях граф і рядків таблиці можна використовувати слова: «*більше*», «*менше*», «*не більше*», «*не менше*», «*в межах*». Ці слова розміщують після одиниці фізичної величини:

(Напруга, V , не більше),

а також використовують слова «від», «більше», «до»: (Від 10 до 15; більше 15; до 20)

Дані, що наводяться в таблиці, можуть бути у вигляді тексту або числовими.

Слова записують в графах з однієї позиції. Якщо рядки таблиці не розділені лініями, то текст, який повторюється та складається з одного слова дозволяється замінювати лапками (,,). Якщо текст складається з двох і більше слів, то за першого повторення його замінюють словами «те ж», а далі лапками. У випадку розділення таблиці горизонтальними лініями – ніякої заміни не виконують.

Числа записують посередині графи так, щоб їх однакові розряди по всій графі були точно один під одним, за виключенням випадку, коли вказують інтервал. Інтервал вказують від меншого числа до більшого з тире між ними:

$12 - 35$

$122 - 450$.

Дробові числа наводять у вигляді десяткових дробів, з однаковою кількістю знаків після коми в одній графі. Розміри в дюймах можна записувати у вигляді: $1/2"$, $1/4"$, $1/8"$.

Ставити лапки замість цифр чи математичних символів, які повторюються, не можна. Якщо цифрові чи інші дані в таблиці не наводяться, то ставиться прочерк.

Таблиці нумерують в межах розділів і позначають зліва над таблицею за формою: «Таблиця 4.2 – Найменування таблиці». Крапку в кінці не ставлять. Якщо найменування таблиці довге, то продовжують у наступному рядку починаючи від слова «Таблиця». Номер таблиці складається з номера розділу та порядкового номера таблиці в розділі, розділених крапкою. Дозволяється нумерувати в межах всього документа.

Таблиця може бути великою як в горизонтальному, так і в вертикальному напрямках або іншими словами може мати велику кількість граф і рядків. В таких випадках таблицю розділяють на частини та переносять на інші сторінки або розміщують одну частину під іншою чи поряд.

Якщо частини таблиці розміщують поряд, то в кожній частині повторюють головку таблиці, а у випадку розміщення однієї частини під іншою – повторюють боковик.

Якщо в кінці сторінки таблиця переривається і її продовження буде на наступній сторінці, в першій частині таблиці нижню горизонтальну лінію, що обмежує таблицю, не проводять.

У всіх випадках найменування (за його наявності) таблиці розміщують тільки над першою частиною, а над іншими частинами зліва пишуть «Продовження таблиці 4.2» без крапки в кінці.

Інші вимоги до виконання таблиць – відповідно до чинних стандартів на технічну документацію.

Оформлення розділів і підрозділів. Структурними елементами основної частини ПЗ є розділи, підрозділи, пункти, підпункти, переліки.

Розділ – головна ступінь поділу тексту, позначена номером і має заголовок. *Підрозділ* – частина розділу, позначена номером і має заголовок. *Пункт* – частина розділу чи підрозділу, позначена номером і може мати заголовок. *Підпункт* – частина пункту, позначена номером і може мати заголовок. Заголовки структурних елементів необхідно нумерувати тільки арабськими числами.

Кожен розділ рекомендується починати з нової сторінки. Заголовок розділу записують посередині (ДСТУ 3008-95) великими літерами з більш високою насиченістю.

Заголовки розділів, підрозділів, пунктів та підпунктів (при наявності заголовка) записують з абзацу малими літерами, починаючи з великої. Перед заголовком і після нього пропускають один рядок.

Розділи нумерують порядковими номерами в межах всього документа (1, 2, і т.д.). Підрозділи нумерують в межах кожного розділу, пункти – в межах підрозділу і т.д. за формою (3.1, 3.2, 3.2.1, 3.2.2, 3.2.2.1 і т.д.). Цифри, які вказують номер, не повинні виступати за абзац. Після номера крапку не ставлять, а пропускають один знак.

Допускається розміщувати текст між заголовками розділу і підрозділу, між заголовками підрозділу і пункту. Посилання в тексті на розділи виконується за формою: “...наведено в розділі 3”.

Оформлення таблиць. Таблицю розміщують симетрично до тексту після першого посилання на даній сторінці або на наступній, якщо на даній вона не уміщується і таким чином, щоб зручно було її розглядати безповороту або з поворотом на кут 90°. Таблиці у тексті пояснювальної записки набираються основним шрифтом, в деяких випадках розмір шрифту може бути зменшений до 10-12. Підписи таблиць розташовуються над таблицею з вказанням її номера і назви, вирівнявши по лівому краю таблиці. Наприклад,

Таблиця 1 – Основні типи даних

Н	Тип даних	Опис
1	Float	Дійсні числа з плаваючою точкою
2	Integer	Цілі числа
..

На всі таблиці мають бути посилання за формою “ ... в табл. 1 або в дужках по тексту (табл. 1). Посилання на раніше наведену таблицю да- ють

зі скороченим словом ”дивись” (див. табл. 1) за ходом чи в кінці речення.

При перенесенні частин таблиці на інші сторінки, повторюють або продовжують найменування граф. Допускається виконувати нумерацію граф на початку таблиці і при перенесенні частин таблиці на наступні сторінки повт орювати тільки нумерацію граф. У всіх випадках найменування (при його наявності) таблиці розміщують тільки над першою частиною, а над іншими частинами зліва пишуть “Продовження таблиці 1” без крапки в кінці, наприклад,

Продовження таблиці 1

	2	3
	String	Рядок символів
	Char	Символьні літерали

Оформлення рисунків. Розміщують рисунки в тексті або в додатках. В тексті ілюстрацію розміщують симетрично до тексту після першого посилання на неї або на наступній сторінці, якщо на даній вона не уміщується без повороту. На всі рисунки мають бути посилання за формою: “ ... на рис. 3–5”, або в дужках по тексту (рис. 3.6). Посилання на раніше наведений рисунок дають зі скороченим словом ”дивись” (див. рис. 4) за ходом чи в кінці речення.

Кожен рисунок повинен мати номер і підпис, розташовані під рисунком по центру. Крапку в кінці не ставлять, знак переносу не використовують. Якщо найменування рисунка довге, то його продовжують у наступному рядку, починаючи від найменування. Наприклад,



Рисунок 12 – Вигляд екрана при обробці пункту меню “Файл”

Між ілюстрацією і текстом пропускають один рядок.

Нумерують ілюстрації в межах розділів, вказуючи номер розділу і порядковий номер ілюстрації в розділі, розділяючи крапкою. Дозволяється нумерувати рисунки в межах всього документа.

Оформлення формул. Кожну формулу записують з нового рядка, симетрично до тексту, курсивом. Між формулою і текстом пропускають один рядок. Умовні літерні позначення в формулі наводять в тексті або зразу ж під формулою. Для цього після формули ставлять кому і записують пояснення до кожного символу з нового рядка в тій послідовності, в якій вони

наведені у формулі, розділяючи крапкою з комою. Перший рядок повинен починатися з абзацу зі слова “де” і без будь-якого знака після нього.

Всі формули нумерують в межах розділу арабськими числами. Номер вказують в круглих дужках з правої сторони, в кінці рядка, на рівні закінчення формули. Номер формули складається з номера розділу і порядкового номера формули в розділі, розділених крапкою. Дозволяється виконувати нумерацію в межах всього документа.

Формула є частиною речення, тому до неї застосовують такі ж правила граматики, як і до інших членів речення. Якщо формула знаходиться в кінці речення, то після неї ставлять крапку. Формули, які йдуть одна за одною і не розділені текстом, відокремлюють комою.

Структура пояснювальної записки

Пояснювальна записка повинна відповідати індивідуальному завданню, а її оформлення – чинним державним стандартам, які слід враховувати на момент виконання розробки з врахуванням всіх офіційних змін, введених в дію.

Пояснювальна записка повинна мати таку структуру:

Зміст пояснювальної записки (ГОСТ 19.402-78):

- Титульний лист.
- Індивідуальне завдання.
- Анотація.
- Зміст.
- Вступ.
- Аналіз методів рішення задачі.
- Опис логічної структури програми.
- Технічні засоби, що використовуються у розробленій програмі.
- Виклик та завантаження програми.
- Вхідні дані
- Вихідні дані
- Інструкція для користувача
- ВИСНОВКИ
- СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ
- Додатки

В додатках повинні бути:

- лістинг програми;
- схеми алгоритмів (ГОСТ19.701-90 та ГОСТ 19.005-85);
- тестовий приклад.

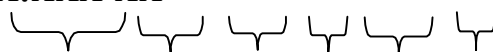
Титульний аркуш

Титульний аркуш є першою сторінкою КР, яка не нумерується. Згідно з діючим стандартом ДСТУ 3008-95 титульний аркуш виконується за встановленим зразком. Зразок титульного аркушу пропонується у додатку Б. Для курсової роботи титульний аркуш виконується без рамки.

На титульному аркуші для курсових робіт подаються: тема курсової роботи; запис „Пояснювальна записка ...” із зазначенням спеціальності, умовне позначення згідно з прийнятою системою (див. далі); перераховується науковий ступінь та звання керівника. Підписи керівника та студента із зазначенням термінів обов’язкові.

Для курсових робіт доцільною є предметна система умовних позначень, яка має таку структуру:

XX-XX.XXX.XXX.XX.XXX XX



1 2 3 4 5 6

де

- 1 (XX-XX) – числовий шифр кафедри, прийнятий у ВНТУ;
- 2 (XXX) – умовне скорочення для дисципліни (ПРГ – Програмування, ЗПЗ – Захист програмного забезпечення і т.д.);
- 3 (XXX) – перша цифра 0, якщо це проект або 1, якщо робота, друга і третя цифри означають рік, наприклад, 08 – 2008 рік);
- 4 (XX) – варіант завдання на курсову роботу (наприклад, 01, 02, ...);
- 5 (XXX) – перший символ – номер групи (1 або 2), наступні два символи задають номер студента за списком у журналі академічної групи;
- 6 (XX) – код документа (ПЗ – пояснювальна записка).

Приклад: 08-02.ПРГ.115.11.105 ПЗ (08-02 – шифр кафедри АІВТ. Програмування. Курсова робота виконана в 2015 році, 11 варіант. Перша група, 5 номер за списком. Пояснювальна записка.)

Робота, яка подається у вигляді копії, до захисту не приймається.

Індивідуальне завдання

Конкретний зміст кожної КР, етапи виконання визначає керівник на підставі індивідуального завдання, затвердженого завідувачем кафедри і затвердженого на засіданні кафедри.

Попередньо керівник видає індивідуальне завдання до курсової роботи. Індивідуальне завдання в перелік змісту не вноситься і має бути другою сторінкою після титульного аркуша. Зразок індивідуального завдання до курсової роботи наведено в додатка В. Обов'язковим в індивідуальному завданні є наведення вхідних і вихідних даних.

Керівник роботи пропонує зміст пояснювальної записки, як правило, в розроблених методичних вказівках. У навчальних цілях зміст може висвітлюватись в індивідуальному завданні. В залежності від специфіки індивідуального завдання керівник курсової роботи може пропонувати тему, яка підлягає конкретному обґрунтуванню та розробці індивідуального завдання, яке дещо відрізняється від типового.

Індивідуальне завдання до курсової роботи має містити термін видачі, підписи керівника та студента.

Завдання на курсову роботу повинно бути підготовлено студентом не пізніше другого тижня з початку навчального семестру, підписано викладачем, що видав завдання і студентом, що прийняв його до виконання.

Анотація

Анотація призначена для ознайомлення з текстовим документом курсової роботи. Анотація повинна коротко характеризувати мету роботи, засоби, використані для досягнення поставленої задачі, коротку інформацію про досягнуті результати. Розмір анотації повинен становити приблизно $\frac{1}{3}$ частину сторінки (не перевищувати $\frac{1}{2}$ сторінки).

Анотацію розміщують безпосередньо за аркушем з індивідуальним завданням, починаючи з нової сторінки (третьої), нумерація якої не зазначається і в зміст не входить.

Зміст

Зміст розташовують безпосередньо після анотації, починаючи з нової сторінки. До змісту включають: вступ; послідовно перелічені назви всіх розділів, підрозділів, пунктів і підпунктів (якщо вони мають заголовки) суті роботи; висновки; перелік використаних джерел; назви додатків і номери сторінок, які містять початок матеріалу. Зміст не включає титульний лист, індивідуальне завдання на курсову роботу та анотацію. Нумерація у змісті починається зі ВСТУПУ (відповідно до нумерації у пояснювальній записці). Сам зміст за нумерацією пояснювальної записки є четвертою сторінкою. Нумерація сторінок повинна бути наскрізною, включаючи додатки.

Назви заголовків змісту повинні однозначно відповідати назвам заголовків пояснювальної записки за текстом. Формування змісту у текстовому документі бажано здійснювати автоматично, використовуючи засоби обраного текстового редактора.

Приклад оформлення змісту:

ВСТУП

1 ЗАГАЛЬНІ ВІДОМОСТІ

2 Функціональне призначення.

3 Аналіз методів рішення задачі.

4 Опис логічної структури програми.

5 Технічні засоби, що використовуються у розробленій програмі.

6 Виклік та завантаження програми.

7 Вхідні дані та Вихідні дані

8 Інструкція для користувача

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Додатки

Додаток А. Назва першого додатка

Додаток Б. Назва другого додатка

Вступ

Вступ пишуть з нової пронумерованої сторінки з заголовком, «**ВСТУП**» посередині (ДСТУ 3008-95 – для КР) **великими** літерами з більш високою насиченістю (жирністю) шрифту.

Текст вступу повинен бути коротким і висвітлювати питання актуальності, значення, сучасний рівень і призначення курсового проекту (роботи). У вступі і далі за текстом не дозволяється використовувати скорочені слова, терміни, крім загальноприйнятих.

Вступ висвітлює:

- стан розвитку проблеми в даній галузі, до якої має відношення розробка;
- галузь використання та призначення;
- мету та загальну постановку задачі;
- актуальність, яка повинна подаватись в останньому абзаці вступу, з метою стислого викладання суті розробки цього напрямку. Кількість сторінок вступу не повинна перевищувати, 1-2 сторінок.

Аналіз методів рішення задачі.

Складовими даного розділу когуд бути наступні елементи:

Виділення математичного об'єкта. Розглянути можливість зберігання та оброблювання даних про об'єкт, в якому вигляді ці дані потрібно представити. Для цього потрібно перейти від фізичного об'єкта до об'єкта математичного. Аналіз існуючих аналогів. Якими засобами можливо зберігання та оброблювання даних

Аналіз технічних і програмних засобів. ПК - технічний засіб. Пояснити, чому обрано середовище C ++ Builder, а не Excel або ін. (.. Можливості ООП і візуального).

Розробка математичної моделі. Под математичною моделлю розуміють систему математичних співвідношень - формул, рівнянь, нерівностей і т.д., відбивають суттєві властивості об'єкта. Метод математичного моделювання зводить дослідження поведінки об'єкта або його властивостей до математичних завдань. Варто виписати формули, наприклад з використанням знаків підсумовування

При цьому можна користуватися і такою схемою дій:

1. виділити припущення, на яких ґрунтуватиметься математична модель. Наприклад, припустити, що інформація про товар буде зосереджена в структурах даних (або містяться у файлі, або в таблиці, або ...);

2. визначити, що вважати вихідними даними і результатами;

3. записати математичні співвідношення, що зв'язують результати з вихідними даними. Які методи обробки (з описом підходу до вирішення, наприклад, «... Рішення зводиться до задачі накопичення суми елементом тов ... формула»).

Виведені залежності і формули в загальному випадку можуть бути представлені рівняннями, системами рівнянь - лінійних, інтегральних, матричних, диффе-ренціальних та ін .. На цьому етапі для знаходження рішення також будується нефор-мінімальний алгоритм (похідна від довжини і рішення рівняння: похідна дорівнює 0,).

При побудові математичних моделей далеко не завжди вдається знайти формули, явно виражають шукані величини через дані. У таких випадках використовуються мате-автоматично методи, що дозволяють дати відповіді тій чи іншій мірі точності.

Розробка вимог до додатка. Наприклад, сказати про те, що користувач повинен мати можливість для завдання вихідних даних вручну, можливість для збереження даних у файлі, можливість формувати файл даних з метою повторного використання; отримувати дані з файлу даних; вибирати вид відображуваної інформації (наприклад, загальна кількість товару, сумарна вартість усього товару, номенклатура товару і т.д. відповідно до вимог); відображати потрібну інформацію у відповідності з обраним видом. А також забезпечення з'єднань цих даних відповідно до користувальницької логікою (всі виходячи з поставленого завдання і особистих мотивів).

Опис логічної структури програми.

В даному розділі повинна бути детально описана мета роботи, алгоритм досягнення результату, передбачена розробка загальної схеми функціонування програми.

В цьому розділі формуються і обґрунтовуються основні вимоги до технічних і програмних показників розробки, які надалі у стислому вигляді будуть подані у вигляді інструкцій, вимоги до характеристик програми, яка розробляється для розв'язання поставленої задачі. Тут наводиться математична модель (якщо необхідно), проводяться дослідження використуваних методів, аналіз роботи методів і алгоритмів на простих прикладах. Результатом такого аналізу повинні бути розроблені блок-схеми алгоритмів, схеми взаємозв'язків між підпрограмами.

Підготовка інтерфейсу програми, що розробляється, також повинна знайти відображення у даному розділі, які види вікон треба використати і чому, які пункти меню доцільно включити і для чого, які елементи керування слід передбачити і для чого, – все це повинно знайти відображення в даному розділі. В результаті повинна бути розроблена схема роботи програми.

Кожна програма повинна мати певні вхідні і вихідні дані, як вони будуть готуватись, звідки вводяться у програму, де зберігатись, що буде вихідними даними і у якому вигляді – це також повинно бути описано у цьому розділі або подано у вигляді схеми даних.

Крім того, в даному розділі слід розробити послідовність основних кроків, необхідних для створення конкретного програмного продукту. Якщо використовуються якісь особливі методи або логічні розв'язування поставленої задачі, їх теж потрібно описати. Таким чином, даний розділ повинен бути підготовкою для наступного етапу – етапу програмування.

Використання схем

Окремими підпунктами основного розділу або завершенням певних підрозділів бажано розробити і описати різні схеми:

– схеми даних;

- схеми програм;
- схеми роботи системи;
- схеми взаємодії програм;
- схеми ресурсів системи.

Розробник програми сам повинен вирішувати, які саме схеми доцільно розробляти у своїй роботі.

Схеми можуть використовуватися на різних рівнях деталізації, причому кількість рівнів залежить від розмірів і складності задачі оброблення даних.

Схеми алгоритмів, програм, даних і систем складаються з символів, що мають задане значення, короткого тексту пояснення і з'єднувальних ліній. Усі символи поділяються на такі підгрупи:

- символи даних (додаток Г, табл. Г. 1);
- символи процесів (додаток Г, табл. Г. 2). Приклад наведено на рис.1.;
- спеціальні символи (додаток Г, табл. Г. 3). Приклад наведено на рис.2;
- символи ліній (додаток Г, табл. Г. 4). Приклад наведено на рис.3.

Символи поділяються на:

- основні, для випадків, коли точний вигляд процесу або носія даних невідомий або відсутня необхідність в описі фактичного носія даних;
- специфічні, використовувані тоді, коли відомий точний вигляд процесу або носія даних або коли необхідно описати фактичний носій даних.

Схеми даних відображають шлях даних при розв'язанні задач і визначають етапи обробки, а також різні використовувані носії даних. Схема даних складається з:

- символів даних, які можуть також вказувати вид носія даних;
- символів процесу, який виконується над даними (можуть також вказувати функції, виконувані обчислювальною машиною);
- символів ліній, які вказують потоки даних між процесами і (або) носіями даних;
- спеціальних символів для полегшення написання і читання схеми.

Символи даних чергуються з символами процесу. Схема даних починається і закінчується символами даних (за винятком спец. символів).

Схеми програм відображають послідовність операцій в програмі.

Схема програми складається з:

- символів процесу, що вказують фактичні операції обробки даних (включаючи символи, що визначають шлях, якого слід дотримуватися з урахуванням логічних умов);
- лінійних символів, що вказують потік управління;
- спеціальних символів, використовуваних для полегшення написання і читання схеми.

Схеми роботи системи відображають управління операціями і потік даних в системі. Схема роботи системи складається з:

- символів даних, що вказують на наявність даних (символи даних можуть також вказувати на вид носія даних);
- символів процесу, що вказують операції, які слід виконати над даними, а також визначають логічний шлях, якого слід дотримуватися;
- лінійних символів, що вказують на потоки даних між процесами і (або) носіями даних, а також потік управління між процесами;
- спеціальних символів, використовуваних для полегшення написання і читання блок-схеми.

Схема взаємодії програм відображає шлях активацій програм і взаємодій з відповідними даними. Кожна програма в схемі взаємодії програм показується тільки один раз (у схемі роботи системи програма може зображатися більше, ніж в одному потоці управління). Схема взаємодії програм складається з:

- символів даних, що вказують на наявність даних;
- символів процесу, що вказують на операції, які виконують над даними;
- лінійних символів, що відображають потік між процесами і даними, а також ініціації процесів;
- спеціальних символів, використовуваних для полегшення написання і читання схеми.

Схема ресурсів системи відображає конфігурацію блоків даних, блоків обробки цих даних, яка потрібна для розв'язання задачі або набору задач. Схема ресурсів системи складається з:

- символів даних, що відображають вхідні, вихідні і запам'ятовують пристрої обчислювальної машини;
- символів процесу, що відображають процесори (центральні процесори, канали і т.д.);
- лінійних символів, що відображають передачу даних між пристроями введення-виведення і процесорами, а також передачу управління між процесорами;
- спеціальних символів, використовуваних для полегшення написання і читання схеми.

Програмна реалізація задачі

Необхідно дати обґрунтування того, чому ті або інші засоби програмування доцільно використати для реалізації певних цілей завдання. Проводиться покроковий опис програмної реалізації алгоритму поставленої задачі, додержуючись принципів структурного програмування. Описуються основні структури мови програмування, які використані в даній роботі. Наводяться фрагменти (не більше в 5-10 операторів, а не програма цілком!). Описуються всі основні змінні, а також допоміжні масиви, якщо вони використовуються при розв'язанні задачі. Всі допоміжні модулі, класи, процедури і функції, основні і заголовочні файли, які використовуються при розробці програмного

забезпечення, також повинні бути описані коротко, якщо вони є стандартними, і детально, якщо вони є продуктом роботи студента.

Якщо розроблено власні підпрограми або функції, вони повинні бути описані і дано посилання на ті додатки, де знаходиться лістинг програми. Якщо використовуються стандартні функції, навести їх доцільність та мету використання і внутрішнє наповнення цих функцій (також з посиланням на відповідні додатки).

Оскільки програма розробляється для її використання в операційному середовищі Windows, то слід звернути особливу увагу на повідомлення від операційної системи та їх обробку і на використання API-функцій.

Крім того, оскільки розроблена програма має бути повноцінним додатком Windows, то необхідно докладно описати програмні засоби для реалізації користувацького інтерфейсу: які діалогові вікна використані, які елементи керування за що відповідають, які засоби використані для реалізації підказок, надання допомоги, виведення повідомлень.

Даний розділ також має бути дуже змістовним, конкретним, зрозумілим, оскільки саме він демонструє знання та навички у галузі програмування. Рекомендований обсяг даного розділу – 8-10 сторінок пояснювальної записки.

Технічні засоби, що використовуються у розробленій програмі.

- *призначення і умови застосування програми* (тут слід вказати призначення і функції, виконувані програмою, умови, необхідні для виконання програми – об'єм оперативної пам'яті, вимоги до складу і параметрів пристроїв, вимоги до програмного забезпечення і т.п.);

Виклик та завантаження програми.

- *характеристика програми* (опис основних характеристик і особливостей програми – тимчасові характеристики, режими роботи, засоби контролю правильності виконання і самовідновлення програми і т.п.);

- *звернення до програми* (опис процедур виклику програми (способи передачі управління, параметрів і ін.);

Вхідні дані та вихідні дані

вхідні і вихідні дані (опис організації використовуваної вхідної і вихідної інформації і, при необхідності, її кодування);

Інструкція оператора

Інструкція оператора повинне містити такі підрозділи (відповідно до ГОСТ 19.505-79):

- *призначення програми* (відомості про призначення програми і інформація, достатня для розуміння функцій програми і її експлуатації);

- *умови виконання програми* (умови, необхідні для роботи програми: мінімальний і/або максимальний склад апаратних і програмних засобів і т. п.);

- *виконання програми* (послідовність дій оператора, що забезпечують завантаження, запуск, виконання і завершення програми, опис функцій, формату і можливих варіантів команд, за допомогою яких оператор здійснює завантаження і управління ходом виконання програми);

повідомлення оператору (тексти повідомлень в ході виконання програми, опис їх змісту і відповідні дії оператора: у разі збою, можливості повторного запуску програми і т. п.).

Висновки

Висновки оформляють з нової пронумерованої сторінки посередині (ДСТУ 3008-95) великими літерами більш високої насиченості.

У висновках наводяться основні результати роботи над курсовою роботою. На основі проведених досліджень результатів роботи надаються обґрунтовані висновки щодо переваг та недоліків застосування тієї чи іншої мови програмування, того чи іншого засобу програмування, недоліки та переваги даного програмного продукту, труднощі при розробленні програми та причини, що їх обумовили і можливі шляхи їх подолання, можливі рекомендації прикладного застосування та шляхи (перспективи) удосконалення розробленого програмного забезпечення.

Оформлення переліку використаних джерел

Список містить перелік літературних джерел, на які повинні бути обов'язкові посилання в тексті пояснювальної записки. Література (книги, статті, патенти, журнали) в загальний список записується в порядку посилання на неї в тексті. В даному переліку дається оформлений відповідно до вимог державних стандартів, список тих джерел (книги, підручники, журнали, електронні адреси), які було використано в процесі виконання роботи, і на яку є посилання в тексті пояснювальної записки. Кожне джерело повинно бути вказано разом з видавництвом, роком видання, кількістю сторінок. Літературу записують мовою оригіналу. В списку кожен літературний запис записують з абзацу, нумерують арабськими цифрами, починаючи з одиниці. Правильне оформлення певного джерела інформації можна переглянути у переліку літературних джерел у будь-якому навчальному посібнику. Якщо у списку використаних джерел є посилання на Інтернет-сторінки, слід наводити разом з назвою Інтернет-сторінки.

Приклад оформлення переліку використаних джерел різного характеру.

Посилання на книги

1. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы и исходные тексты на языке С. 2-е изд. - СПб.: Вильямс, 2000. - 789 С.
2. Бабаш А. В., Шанкин Г. П. Криптография. Под ред. Шерстюка В. П., Применко Э. А. / Шанкин Г. П.. - М.: СОЛОН-Р, 2002. - 512 С.
3. Алферов А. П., Зубов А. Ю., Кузьмин А. С., Черемушкин А. В. Основы криптографии. Учебное пособие. - М.: Гелиос АРВ, 2001. - 480 С.

Посилання на журнали

4. Ершов А. А. Стабильные методы оценки параметров // Автоматика и телемеханика. – 1978. – №8. – С. 86-91.

Посилання на ГОСТ і ДСТУ

5. ГОСТ 7.9-77. Реферат и аннотация. – М.: Издательство стандартов, 1981.

– 6 с.

Посилання на патенти

6. Пат. 3818311, США, МКИ НОЗК 17/60. Схема защиты полупроводникового переключателя. – Оpubл. 04.05.84.

Посилання на web-сторінки

7. Інформаційна безпека // <http://www.domarev.ru/>.
8. Научная библиотека // www.abris.crimea.ua/index.php.

Оформлення додатків

Додатки повинні містити матеріал, який не увійшов в основні розділи пояснювальної записки: лістинги програм, підпрограм та функцій, результати тестування програми у вигляді образів екранів, таблиць, графіків, схеми роботи програм, блок-схеми алгоритмів, схеми ресурсів і даних, схеми взаємодії програм тощо.

Кожен додаток необхідно починати з нової сторінки, вказуючи зверху посередині рядка слово “Додаток” і через пропуск – його позначення. Додатки позначають послідовно великими українськими літерами, за винятком літер Г, Є, З, І, Ї, Й, О, Ч, Ь, наприклад, Додаток А, Додаток Б.

Кожен додаток повинен мати тематичний (змістовний) заголовок, який записують посередині рядка малими літерами з першої великої.

Сторінки додатків нумеруються, продовжуючи загальну нумерацію у пояснювальній записці.

Всі додатки включають у зміст, вказуючи номер додатка, заголовок і номер сторінки, з яких вони починаються.

Приклад оформлення додатків можна переглянути у додатках до даних методичних вказівок.

5 ГРАФІК ВИКОНАННЯ КУРСОВОЇ РОБОТИ І ПОРЯДОК ЇЇ ЗАХИСТУ

Рекомендується такий графік виконання курсової роботи, який враховує самостійну роботу студентів (16 тижнів).

Зміст розділу	Термін виконання
Отримання завдання на курсову роботу, розробка і оформлення індивідуального завдання	1-2 тижні
Розробка структури програмного забезпечення: дослідження алго- ритму задачі, виконання вручну контрольних прикладів, розробка інтерфейсу, обґрунтування необхідності додаткових засобів, розробка структури вхідних і вихідних даних, підбір	3-4 тижні
Розробка програмного забезпечення і налагоджування його: програмування та тестування основних процедур та функцій, програмна реалізація інтерфейсу, програмна реалізація роботи з файлами, з елементами керування,	5-11 тижні
Тестування розробленого програмного продукту та виправлення виявлених недоліків. Підготовка тестових прикладів.	11 тиждень
Оформлення пояснювальної записки до курсової роботи, розробка рекомендацій для роботи з розробленою програмою	12-13 тижні
Здача курсової роботи на попередню перевірку: демонстрація роботи програми та чернетки пояснювальної записки (бажано тверда копія, але можливий електронний варіант)	13 тиждень
Корегування і доповнення (при необхідності) програми згідно із зауваженнями керівника курсової роботи, врахування і виправлення	13-14 тижні
Захист курсової роботи	13-14 тижні

Готовність до захисту курсової роботи визначає керівник за результатами попередньої перевірки якості пояснювальної записки та робоздатності програми (згідно із графіком попереднього захисту). Записка повинна бути здана керівнику на перевірку не менше, як за тиждень до визначеного терміну захисту роботи. Якщо робота виконана в повному обсязі і не має принципових помилок, керівник допускає студента до захисту. В іншому випадку робота повертається студенту на доопрацювання протягом вказаного терміну. Після позитивного висновку про готовність курсової роботи студент повинен захистити її перед комісією у складі двох викладачів, які призначені кафедрою.

КРИТЕРІЇ ОЦІНКИ КУРСОВОГО ПРОЕКТУ

Оцінка відмінно (5) виставляється, якщо проект виконано в повному обсязі згідно завдання. Мають місце:

- 1) обґрунтований опис алгоритму, алгоритм у вигляді блок-схеми, спроектований оптимально, приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана оптимально, текст програми має необхідний опис чи коментарі;
- 3) є повний опис роботи програми та роботи з програмою для користувача;
- 4) пояснювальна записка оформлена згідно з вимогами стандартів та акуратно;
- 5) при захисті курсового проекту студент вільно орієнтується в матеріалі, який відноситься до теми курсового проекту та своїй записці, відповідає на запитання.

Можлива присутність деяких недоліків у вигляді описок та дрібних помилок при відповідях на запитання.

Оцінка добре (4) – виставляється, якщо курсовий проект виконаний в повному обсязі згідно завдання. Мають місце:

- 1) обґрунтований опис алгоритму, сам алгоритм у вигляді блок-схеми, спроектований переважно оптимально, приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана переважно оптимально, текст програми має необхідний опис чи коментарі;
- 3) є повний опис роботи програми та роботи з програмою для користувача;
- 4) пояснювальна записка оформлена згідно з вимогами стандартів та в основному акуратно;
- 5) при захисті курсового проекту студент в основному орієнтується в матеріалі, який відноситься до теми курсового проекту та своїй записці, відповідає на більшу частину запитань.

Можлива присутність недоліків у вигляді описок, помилок, які не впливають на працездатність програми та помилок при відповідях на запитання.

Оцінка задовільно (3) – виставляється, якщо робота виконана не в повному обсязі, але згідно завдання. Мають місце:

- 1) необґрунтований опис алгоритму, алгоритм у вигляді блок-схеми, спроектований не оптимально, не приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана не оптимально, текст програми не має необхідний опис чи коментарі;

- 3) є неповний опис роботи програми та роботи з програмою для користувача;
- 4) пояснювальна записка оформлена в основному згідно вимог стандартів але з помилками та неакуратно;
- 5) при захисті курсового проекту студент погано орієнтується в матеріалі який відноситься до теми курсового проекту та в своїй записці, відповідає на меншу частину запитань.

Можлива присутність помилок, які впливають на працездатність програми, або відсутність частини алгоритму, студент допускає грубі помилки при відповідях на запитання.

Оцінка незадовільно (2) – виставляється, якщо робота виконана в неповному обсязі згідно технічного завдання, не згідно технічного завдання або зовсім не виконано. Мають місце:

- 1) відсутній опис алгоритму, сам алгоритм у вигляді блок-схеми або спроектований повністю не оптимально, не приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана не оптимально або взагалі неправильно, текст програми не має необхідний опис чи коментарі;
- 3) не має опису роботи програми та роботи з програмою для користувача;
- 4) при захисті курсового проекту студент не орієнтується в матеріалі, який відноситься до теми курсового проекту та в своїй записці, не відповідає на запитання.

ПЕРЕЛІК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Либерти Джес С. Освой самостоятельно С++ за 21 день. – Москва- Санкт-Петербург-Киев: «Вильямс», 2001. - 815 с.
2. Павловская Т.А. Программирование на языке высокого уровня С/С++. – С.-П.: «Питер», 2002. - 460 с.
3. Герберт Шилдт, Полный справочник по С. – Москва-Санкт-Петербург- Киев: «Вильямс», 2003. – 800 с.
4. Грегори К. Использование Visual С++ 6. – Москва-Санкт-Петербург- Киев: «Вильямс», 2000. – 850 с.
5. Глушаков С. В., Коваль А. В., Черепнин С. А. Программирование на Visual С++ 6.0. – Харьков: Фолио, 2002. – 726 с.
6. Глушаков С. В., Коваль А. В., Смирнов С.В.. Язык программирования С++. – Харьков: «Фолио», 2002. - 500 с.
7. Гради Буч. Объектно-ориентированное программирование с примерами применения.: Пер. с англ.. – М.: Конкорд, 1992. - 519 с.
8. Ганеев Р. М. Проектирование пользовательского интерфейса средствами Win32API. – М.: Горячая линия - Телеком, 2001. - 334 с.
9. Румянцев П. В. Азбука программирования Win32API. – М.: Горячая линия - Телеком, 2001. - 310 с.
10. Румянцев П. В. Работа с файлами в Win32API. – М.: Горячая линия - Телеком, 2002. - 216 с.
11. Кухар В. М., Тадян С. І. Математика. Множини. Логіка. Цілі числа. Практикум. – Київ: Вища школа, 1989. – 333 с.

Додаток А

Варіанти завдань на курсову роботу

Варіант № 1.

Тема: Аналіз методів введення обмежених обсягів текстової інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити довільний текст в область, обмежену екраном монітора. Програма повинна передбачати можливість зберігання і завантаження тексту з файла.

Варіант № 2.

Тема: Аналіз методів введення обмежених обсягів текстової інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити довільний текст в файл. Програма повинна передбачати можливість зберігання і завантаження тексту з файла.

Варіант № 3.

Тема: Аналіз методів введення обмеження обсягів графічної інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити графічне зображення в область, обмежену екраном монітору. Програма повинна передбачати можливість зберігання і завантаження тексту з файла. Формат файла визначається в процесі уточнення технічного завдання.

Варіант № 4.

Тема: Аналіз методів введення довільних обсягів графічної інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити графічне зображення довільного розміру. Програма повинна передбачати можливість зберігання і завантаження тексту з файла. Формат файла визначається в процесі уточнення технічного завдання.

Варіант № 5.

Тема: Аналіз методів введення текстової і графічної інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити текстову і графічну інформацію в ЕОМ. Програма повинна передбачати можливість зберігання і завантаження

тексту з файла. Формат файла визначається в процесі уточнення технічного завдання.

Варіант № 6.

Тема: Аналіз методів шифрування текстової інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє довільний текст, що зберігається у файлі, шифрувати і зберігати у файл під старим ім'ям. Формат файлу визначається у процесі уточнення технічного завдання. Алгоритм шифрування – додавання до кожного байту файлу по черзі байтів розпакованої дати і часу створення файлу. Програма складається з основної програми і трьох підпрограм: шифрування, дешифрування і відновлення після збою.

Варіант № 7.

Тема: Аналіз методів шифрування текстової інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє довільний текст, що зберігається у файлі, шифрувати і зберігати у файл під старим ім'ям. Формат файлу визначається у процесі уточнення технічного завдання. Алгоритм шифрування – попарне додавання по модулю 2 по трьох байтів файлу починаючи з третього байту. Програма складається з основної програми і трьох підпрограм: шифрування, дешифрування і відновлення після збою.

Варіант № 8.

Тема: Аналіз методів перетворення графічних форматів файлів.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє перетворювати графічну інформацію, що зберігається у файлі, з одного формату в інший. Формати для обробки визначаються в процесі уточнення технічного завдання (BMP, GIF, JPG, і т. д.).

Варіант № 9.

Тема: Аналіз методів перетворення текстових форматів файлів.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє перетворювати текстову інформацію, що зберігається у файлі, з одного формату в інший. Формати для обробки визначаються в процесі уточнення технічного завдання (TXT, DOC, HTML і т. д.).

Варіант № 10.

Тема: Аналіз алгоритмів ігор із нечіткою стратегією.

Пояснення: аналіз повинен проводитися на основі розробленої ігрової програми. Правила ігри формулюються на етапі уточнення технічного завдання.

Варіант № 11.

Тема: Аналіз методів програмування електронних таблиць обмеженого розміру.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити і виконувати елементарні арифметичні операції з числовими даними. Розмір таблиці обмежений екраном монітора. Режим монітору алфавітно – цифровий. Кількість знаків кожної граfi – 15. Керуючі клавіші: F1 – введення даних; F2 – введення розрахункової формули; F3 – перерахунок; Стрілки – переміщення по графах; F4 – припинення роботи.

Варіант № 12.

Тема: Аналіз методів програмування великих електронних таблиць.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити і виконувати елементарні арифметичні операції з числовими даними. Розмір таблиці не повинних бути обмежений екраном монітора. Режим монітору алфавітно – цифровий. Кількість знаків кожної граfi – 15. Керуючі клавіші: F1 – введення даних; F2 – введення розрахункової формули; F3 – перерахунок; Стрілки – переміщення по графах; F4 – припинення роботи.

Варіант № 13.

Тема: Аналіз методів сортування даних.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє сортувати дані декількома методами. Методи сортування вибираються в процесі уточнення постановки задачі (бульбашкова, Шела і т. д.) Режим монітора графічний. Програма повина графічно подавати принцип кожного з методів.

Варіант № 14.

Тема: Аналіз методів автматизації обробки економічної інформації.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє розраховувати заробітну плату співробітникам підприємства. Порядок розрахунків та перелік вихідних форм з'ясовуються в результаті уточнення технічного завдання.

Варіант № 15.

Тема: Аналіз методів автматизації обробки інформації про законодавчі акти.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє обробляти інформацію про чинні законодавчі акти, що зберігаються в базі даних.

Варіант № 16.

Тема: Аналіз методів автоматизації обробки інформації про виробничу діяльність підприємства.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє обробляти інформацію про планування постачань сировини і комплектації підприємству.

Варіант № 17.

Тема: Аналіз методів автоматизації обробки облікової інформації.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє обробляти інформацію про наявність товарів на складі.

Варіант № 18.

Тема: Аналіз методів автоматизації обробки фінансової інформації.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє обробляти інформацію про банківські операції підприємства.

Варіант № 19.

Тема: Аналіз методів автоматизації обробки інформації, пов'язаної з діловодством.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє обробляти інформацію про кореспонденцію, що надходить і відправляється з підприємства. Вихідні дані: журнал урахування вхідної і вихідної кореспонденції. Програма повина забезпечувати пошук по заданим умовам з поступовим уточненням запиту. Результати пошуку відображаються у вигляді звіту з динамічним формуванням граф з списку полів.

Варіант № 20.

Тема: Аналіз методів автоматизації упорядкування розкладу занять.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє скласти розклад занять по таким вихідним даним: перелік дисциплін із указівкою прізвищ викладачів, видів занять і аудиторій, у яких вони можуть проводитися. Програма повина скласти розклад із мінімальною кількістю вільних пар.

Варіант № 21.

Тема: Аналіз методів захисту інформації від копіювання.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє скласти захищати файли від копіювання в інші місця жорсткого диска.

Варіант № 22.

Тема: Аналіз методів обробки результатів вимірів.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє обчислювати середнє квадратичне відхилення, середнє значення і гістограму на підставі 100 останніх значень даних, що вводяться з клавіатури або з файла. На екран повинен виводитися графік на підставі 50 останніх значень.

Варіант № 23.

Тема: Аналіз методів автоматизації перекладу тексту.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє перекладати довільний текст з української мови на англійську і навпаки. Програма повина мати можливість розширення бази даних.

Варіант № 24.

Тема: Аналіз методів обробки звукової інформації на ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє вводити, відтворювати, зберігати і завантажувати з файла звукову інформацію.

Варіант № 25.

Тема: Аналіз методів шифрування текстової інформації в ЕОМ.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє довільний текст, що зберігається у файлі, шифрувати і зберігати у файл під старим ім'ям. Формат файла визначається в процесі уточнення технічного завдання. 2. Алгоритм шифрування – віднімання від коду символу деякого числа: від першого – 1, від другого – 2, і т. д. Програма складається з основної програми і трьох підпрограм: шифрування, дешифрування і відновлення після збою.

Варіант № 26.

Тема: Аналіз методів рішення задач лінійного програмування.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє знайти рішення системи рівнянь.

$$\left\{ \begin{array}{l} x_1 - 2x_2 + x_3 = 10 \end{array} \right.$$

$$-2x_1 - x_2 - 2x_4 > 18$$

$$3x_1 + 2x_2 + x_4 > 36$$

при умовах $x_1, x_2, x_3, x_4 > 0$, що забезпечує найменше значення функції

$$F = 2x_1 - x_2 - x_4.$$

Варіант № 27.

Тема: Аналіз методів рішення двоїстих задач лінійного програмування.

Пояснення: аналіз повинен проводитися на прикладі розробленої програми, що дозволяє

Варіант № 28.

Тема: Аналіз методів рішення транспортної задачі.

Варіант № 29.

Тема: Аналіз методів рішення цілочисельної задачі лінійного програмування.

Варіант № 30.

Тема: Аналіз методів рішення задач параметричного програмування.

Примітка. Варіанти завдань можуть змінюватись та доповнюватись викладачем.

Додаток Б
Приклад оформлення титульного аркуша

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет комп'ютерних систем і автоматики

**АНАЛІЗ МЕТОДІВ ЗАХИСТУ ІНФОРМАЦІЇ ВІД
КОПІЮВАННЯ**

Пояснювальна записка
до курсової роботи з дисципліни "Програмування"
за напрямом підготовки
6.050201 – "Системна інженерія"
08-02.ПРГ.115.11.105 ПЗ

Керівник курсової роботи
Доцент каф. АІВТ _____ Довгалець С.М.

Розробив студент гр. 1СІ-14
_____ Іванов М. Д.
Курсову роботу захищено
з оцінкою _____

_____ 2015 р.
(Підпис керівника) (Дата)

(Підпис асистента)

Додаток В
Приклад оформлення індивідуального завдання

Вінницький національний технічний університет
Факультет комп'ютерних систем і автоматики

ЗАТВЕРДЖУЮ
Зав. кафедри АІВТ, д.т.н., проф.
_____ Р.Н. Кветний
"_____" _____ 20__р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу з дисципліни "Програмування"
студенту групи ІСІ-146 факультету КСА Іванову М. Д.
Тема: «Аналіз методів захисту інформації від копіювання»

Вимоги до курсової роботи.

1. Дослідити і засвоїти методи захисту даних.
2. Розробити загальну структурну схему функціонування програми в цілому і алгоритмів захиту.
3. Підібрати програмні засоби для реалізації розробленого алгоритму, підготувати складові користувацького інтерфейсу.
4. Програмно реалізувати поставлену задачу і підготувати детальний опис послідовності розробки програмного засобу.
5. Протестувати програму на предмет ефективності і правильності її роботи і розробити рекомендації для роботи з нею.

Вихідні дані: об'єкт захисту – текстові файли; інтерфейс – віконно - графічний; операційна система – сімейство Windows.

Дата видачі _____ 20__р.
Керівник _____ Довгалець С.М.
Завдання отримав _____ Іванов М.Д.





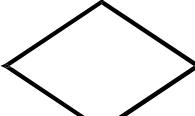
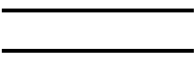
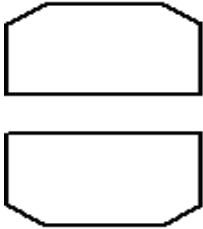
Додаток Г

Символи даних, процесів і ліній

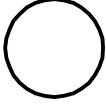

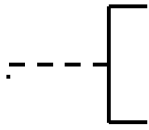

Таблиця Г.1 – Символи даних

<i>Основні символи даних</i>		
Дані		Символ відображає дані, носій даних невизначений
Дані, що запам'ятовуються		Символ відображає дані, що зберігаються, у вигляді, придатному для обробки, носій даних невизначений
<i>Специфічні символи даних</i>		
Оперативний запам'ятовувальний пристрій		Символ відображає дані, що зберігаються в оперативному запам'ятовувальному пристрої
Запам'ятовувальний пристрій з послідовним доступом		Символ відображає дані, що зберігаються в запам'ятовувальному пристрої з послідовним доступом (магнітна стрічка, касета з магнітною стрічкою, магнітофонна касета)
Запам'ятовувальний пристрій з прямим доступом		Символ відображає дані, що зберігаються в запам'ятовувальному пристрої з прямим доступом (магнітний диск, магнітний барабан, гнучкий магнітний диск)
Документ		Символ відображає дані, подані на носії в легкій для читання формі (документ для оптичного або магнітного зчитування, мікрофільм, рулон стрічки з підсумковими даними, бланки введення даних)
Ручне введення		Символ відображає дані, що вводяться вручну під час оброблення з пристроїв будь-якого типу (клавіатура, перемикачі, кнопки, світлове перо, смужки зі штриховим кодом)
Карта		Символ відображає дані, подані на носії у вигляді карти (перфокарти, магнітні карти, карти зі зчитуваними мітками, карти з відривним ярликом, карти зі сканованими мітками).
Паперова стрічка		Символ відображає дані, подані на носії у вигляді паперової стрічки
Дисплей		Символ відображає дані, подані у візуальній людиночитабельній формі на носії у вигляді пристрою відображення (екран для візуального спостереження, індикатори введення інформації)


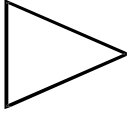


Таблиця Г. 2 – Символи процесу

<i>Основні символи, процесу</i>		
Процес		Символ відображає функцію обробки даних будь-якого вигляду (виконання певної операції або їх групи, що приводить до зміни значення, форми інформації).
<i>Специфічні символи процесу</i>		
Підпорядкований процес		Символ відображає підпорядкований процес, що складається з однієї або декількох операцій або кроків програми, які визначені у іншому місці
Ручна операція		Символ відображає будь-який процес, виконуваний людиною
Підготовка		Символ відображає модифікацію команди або групи команд з метою дії на деяку подальшу функцію (установлення перемикача, модифікація індексного реєстра або ініціалізація програми)
Умова або вибір		Символ відображає умову, вибір або функцію типу перемикача, що має один вхід і ряд альтернативних виходів, один і лише один з яких може бути активізований після обчислення умов, визначених усередині цього символу
Паралельні дії		Символ відображає синхронізацію двох або більше паралельних операцій
Межа циклу		Символ, що складається з двох частин, відображає початок і кінець циклу. Обидві частини символу мають один і той самий ідентифікатор. Умови для ініціалізації, прирости, завершення поміщаються усередині символу на початку або в кінці залежно від розташування операції, що перевіряє умову

Таблиця Г. 3 – Спеціальні символи

<p>З'єднувач</p> 	<p>Символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми і використовується для обривання лінії і продовження її у іншому місці. Відповідні символи-з'єднувачі повинні містити одне і те ж унікальне позначення</p>
<p>Термінатор</p> 	<p>Символ відображає вихід в зовнішнє середовище і вхід із зовнішнього середовища (початок або кінець схеми програми, зовнішнє використання і джерело або пункт призначення даних)</p>
<p>Коментар</p> 	<p>Символ використовують для додавання описових коментарів або записів пояснень з метою пояснення або приміток. Пунктирні лінії в символі коментаря пов'язані з відповідним символом або можуть окреслювати групу символів. Текст коментарів або приміток повинен бути поміщений біля обмежуючої фігури</p>
<p>Пропуск</p> 	<p>Символ (три крапки) використовують в схемах для відображення пропуску символу або групи символів, в яких не визначені ні тип, ні число символів. Символ використовують тільки в символах лінії або між ними. Він застосовується головним чином в схемах, що зображають загальні результати вибору з невідомим числом повторень</p>

Таблиця Г. 4 – Символи ліній

<p><i>Основний символ ліній</i></p>	
<p>Лінія</p> 	<p>Символ відображає потік даних або управління. У разі необхідності або для підвищення легкості читання можуть бути додані стрілки-показчики</p>
<p><i>Специфічні символи ліній</i></p>	
<p>Передача управління</p> 	<p>Символ відображає безпосередню передачу управління від одного процесу до іншого, іноді з можливістю прямого повернення до ініціувального процесу після того, як ініційований процес завершить свої функції. Тип передачі управління повинен бути названий усередині символу (наприклад, запит, виклик, подія)</p>
<p>Канал зв'язку</p> 	<p>Символ відображає передачу даних по каналу зв'язку</p>
<p>Пунктирна лінія</p> 	<p>Символ відображає альтернативний зв'язок між двома або більшою кількістю символів, а також використовується для обведення ділянки</p>

Інструктивно-методичне видання

**Методичні вказівки
до виконання курсової роботи
з «Програмування»
для студентів напрямку
6.050201 – «Системна інженерія»**

Редактор В. Дружиніна

Укладач: Довгалець С.М.

Оригінал-макет підготовлено С. Плахотнюк

Підписано до друку

Формат 29,7×42¼. Папір офсетний.

Гарнітура Times New Roman.

Друк різнографічний. Ум. др. арк.

Наклад ... прим. Зам. № 2014-

Вінницький національний технічний університет,
навчально-методичний відділ ВНТУ.

21021, м. Вінниця, Хмельницьке шосе, 95,

ВНТУ, к. 2201.

Тел. (0432) 59-87-36.

Свідоцтво суб'єкта видавничої справи
серія ДК № 3516 від 01.07.2009 р.

Віддруковано у Вінницькому національному технічному
університеті

в комп'ютерному інформаційно-видавничому центрі.

21021, м. Вінниця, Хмельницьке шосе, 95,

ВНТУ, ГНК, к. 114.

Тел. (0432) 59-85-32.

Свідоцтво суб'єкта видавничої справи
серія ДК № 3516 від 01.07.2009 р.