

Структура модуля

ТЕМА: " МОДЕЛЮВАННЯ. ОСНОВИ АЛГОРИТМІЗАЦІЇ"

Успішне вивчення теми дозволить:

описувати:

- поняття моделі, об'єкту, предметної області;
- типи моделей, їх характеристики;
- поняття алгоритму;
- властивості алгоритмів;
- форми подання алгоритму;
- базові структури алгоритмів та їх особливості;
- порядок складання алгоритмів;
- поняття методу розв'язування задачі;
- технологію комп'ютерного моделювання;

називати

- призначення графічних схем базових структур алгоритмів;

пояснювати

- етапи розв'язування задачі за допомогою комп'ютера;
- поняття програми;

використовувати

- графічні схеми базових структур алгоритмів для опису алгоритмів;
- різні засоби подання алгоритмів;
- будувати інформаційну модель задачі;
- аналізувати алгоритм розв'язування задачі;
- будувати графічні схеми найпростіших алгоритмів;
- формально виконувати алгоритми

Ключові слова: постановка задачі, модель, математична модель, імітаційна модель, інформаційна модель, алгоритм, виконавець алгоритму, блок-схема, система команд виконавця, алгоритм слідування, алгоритм розгалуження, алгоритм повторення, метод, метод пошагової деталізації, мова, мова програмування, алгоритмічна мова, програма, синтаксична помилка, семантична помилка.

1. ІНСТРУКТИВНО-МЕТОДИЧНІ МАТЕРІАЛИ

Методичні рекомендації щодо вивчення змісту модуля

Методика викладання теми «Моделювання. Основи алгоритмізації»

Поняття моделі безпосередньо пов'язане з поняттям об'єкта. Не існує точного означення поняття об'єкта – це не означуване поняття. Вводячи це поняття, можна просто зазначити, що в житті людину оточують різні прояви живої та неживої природи, які можна називати об'єктами людської уваги. Взагалі об'єктами називають все, на що спрямована увага людини, з чим людина працює фізично чи розумово. Кожний об'єкт має певні характеристики і кожен з них можна вивчати з різних сторін. Для встановлення характеристик об'єкта необхідно з ним працювати безпосередньо, але часто через складність чи недоступність об'єкта це неможливо. В таких випадках дослідження проводять не безпосередньо з об'єктом, а з деяким заміником, який має такі ж або близькі характеристики в плані предмету дослідження. Такий заміник називають моделлю об'єкта дослідження.

Хоча технічні моделі не є предметом вивчення інформатики, однак доцільно зупинитися на їх обговоренні, інформатика вивчає інформаційні моделі. Разом з тим між поняттями фізичної (матеріальної) і інформаційної (знакової) моделей існують певні аналогії.

Моделі можуть бути матеріальними (фізичними), знаковими, уявними. Прикладами фізичних моделей можна вважати: для земної кулі - глобус; для реального автомобіля - іграшковий автомобіль; для реальної людини - лялька; для великої будівлі - маленька пластикова чи картонна і т.д.

Розширивши список матеріальних моделей, доцільно обговорити їх спільні властивості. Всі ці моделі до деякої міри характеризують з того чи Іншого боку об'єкт-оригінал. Часто модель відтворює лише форму реального об'єкта у зменшеному масштабі. Але можуть існувати моделі, які відтворюють деякі функції об'єкта. Наприклад, маленький автомобіль може їздити, модель корабля може плисти. Модель не обов'язково повинна мати всі характеристики такі ж, як у реального об'єкта, а лише ті, які необхідні для її майбутнього застосування з метою вивчення деяких характеристик реального об'єкта. Тому важливим поняттям в моделюванні є поняття мети. Мета моделювання - призначення майбутньої моделі. Мета моделювання визначає ті властивості об'єкта-оригіналу, які повинні бути відтворені в моделі.

Можна дати таке описове означення цього поняття: модель - штучно створений об'єкт у вигляді схеми, рисунка, логіко-математичних знакових формул, фізичних конструкцій та ін., який до деякої міри характеризує досліджуваний об'єкт, відображає і відтворює, можливо у спрощеному, зменшеному вигляді структуру, властивості, взаємозв'язки та відношення між елементами досліджуваного об'єкта, безпосереднє вивчення якого неможливе або важко визначити. Таким чином, за рахунок моделювання полегшується процес одержання інформації про об'єкт, який вивчається.

Слово модель походить від латинського *modulus* (міра, зразок, норма) і означає копію або образ. Доцільно відмітити, що для того, щоб відрізнити один об'єкт від іншого, кожному з них надають назви - імена. Об'єкт, крім того, має певні властивості-сукупність ознак, за якими його можна відрізнити від інших. Кожну з властивостей характеризують певні параметри, які можуть набувати різних значень.

При розв'язуванні життєвих задач першим кроком повинно бути визначення того, які фактори суттєві для задачі, а якими можна нехтувати. Факторів, які зумовлюють "поведінку" об'єкта, перебіг процесу чи прояв явища, що вивчаються, як правило, дуже багато, і часто людина навіть не знає про всі з них і не може перерахувати. Тому і виділяють невелику прийнятну їх кількість - лише ті, що є суттєвими в межах конкретного дослідження.

Крім того, слід звернути увагу на те, що моделлю можна вважати відображення об'єкта в тому або іншому середовищі тими чи іншими засобами. Гіпсова модель - зображення об'єкта в гіпсі. Математична модель - відображення об'єкта математичними засобами (формули, позначення чисел, функцій, знаки операцій тощо).

Для створення моделі об'єкта, яка з достатньою точністю характеризуватиме реальний об'єкт, необхідно навчитися збирати, правильно подавати і потім опрацьовувати інформацію про нього. Це означає, що для дослідження об'єкта не обов'язково створювати фізичну (матеріальну) модель.

Люди в своїй роботі і повсякденному житті часто використовують зовсім інші моделі. Наприклад, розклад руху потягів - це також модель, але зовсім іншого типу. В ній просто вказані характеристики об'єкта, в даному випадку деякої множини

потягів (можливо одного чи кількох), які направляються із одних пунктів призначення до інших.

Часто для вивчення характеристик об'єкта достатньо мати необхідну інформацію, подану у відповідній формі. В цьому випадку говорять про інформаційну модель об'єкта.

Інформаційна модель -це опис об'єкта чи процесу, в якому вказано деякі типові властивості і характеристики об'єкта, важливі для конкретної задачі,що розв'язується.

Будь-які відомості можна подати в різній формі, тому існують різні форми інформаційних моделей. В їх числі словесні, графічні, математичні, табличні.

Побудова інформаційної моделі, як і матеріальної, повинна бути пов'язана з метою моделювання. Будь-який реальний об'єкт має велику кількість властивостей, тому для моделювання повинні бути виділені лише ті властивості, які відповідають меті дослідження. Процес виділення суттєвих для моделювання характеристик об'єкта, зв'язків між ними з метою їх дослідження і описання, називається системним аналізом.

Побудувавши інформаційну модель, людина використовує її замість об'єкта-оригінала для вивчення властивостей цього об'єкта, прогнозування його поведінки і ін.

Математичне моделювання сьогодні є суттєвим фактором в різних сферах людської діяльності: у плануванні, прогнозуванні, управлінні, при проектуванні чашин, механізмів та систем. Вивчення реальних явищ за допомогою математичних моделей, як правило, вимагає застосування обчислювальних методів. При цьому широко використовуються методи обчислювальної математики, теорії ймовірностей та математичної статистики, інформатики.

Моделювання, метою якого є одержання чисельних значень параметрів процесу або явища, що моделюється, називається чисельним.

В процесі чисельного моделювання використовуються математичні моделі: аналітичні, обчислювальні, імітаційні.

Аналітична модель зв'язує формульними залежностями різні параметри процесу або явища, що моделюється. Одні параметри при цьому відомі, інші - шукані.

Зрозуміло, що з'ясування великої кількості характеристик різних об'єктів вимагає значних зусиль і тривалих спостережень та всебічного аналізу їх результатів, які практично нездійсненні без використання сучасної обчислювальної техніки. При цьому уявлення людини про об'єкт, що досліджується, подається за допомогою інформаційної моделі, тобто деякого опису об'єкту відповідно до уявлень про нього та його властивості. Для опрацювання інформаційної моделі за допомогою комп'ютера її необхідно подати відповідним чином так, щоб була можливість помістити ці описи до запам'ятовуючих пристроїв комп'ютера.

При опрацюванні моделі за допомогою комп'ютера можна піти одним з двох шляхів - написати спеціальну програму для комп'ютера мовою програмування або скористатися вже розробленим прикладним програмним забезпеченням. При виборі шляху враховуються різні обставини, зокрема: мета дослідження; поставлені завдання; можливості використання програмного забезпечення.

Однак в будь-якому разі необхідно чітко сформулювати всі правила опрацювання моделі, чітко окреслити мету, яку потрібно досягти, питання, які потрібно з'ясувати, та завдання, які потрібно виконати для досягнення поставленої мети.

В природничих науках головну роль відіграють кількісні характеристики реальних об'єктів і використовуються відповідні моделі. В шкільних курсах ці моделі обмежуються, як правило, найпростішими рівняннями чи нерівностями.

В гуманітарних науках велике значення відіграють структурні моделі, основу яких складають виділені об'єкти та зв'язки між ними. Вивчення структурних моделей і

взагалі поняття структури близько підводить до основних концептуальних питань інформатики. Тому дуже важливо звернути увагу на поняття структури та зв'язків між об'єктами.

Можна виділити три типи задач з галузі інформаційного моделювання, які за зростанням ступеня складності для сприйняття учнями розташовуються в такому порядку;

1) задано інформаційну модель об'єкта; потрібно навчитися її аналізувати, робити висновки, використовувати для розв'язування задач;

2) дано набір несистематизованих даних про реальний об'єкт(систему;процес); потрібно систематизувати їх і таким чином створити інформаційну модель;

3) дано реальний об'єкт (процес, систему); потрібно розробити його інформаційну модель.

Інформаційне моделювання –це прикладний розділ інформатики, який пов'язаний з різними предметними галузями: технікою, економікою, природничими, гуманітарними, соціальними науками і ін. Тому розв'язуванням задач третього типу займаються фахівці у відповідних галузях знань. В рамках шкільного курсу інформатики інформаційне моделювання може бути предметом профільного курсу, пов'язаного з іншими шкільними дисциплінами:математикою, фізикою, біологією, економікою і т.д.

Однією з проблем, яка постає при вивченні цього модуля, є поєднання досить консервативної алгоритмічної лінії курсу з динамічними та сучасними лініями виконавця,формалізації та моделювання, інформаційних технологій.

Алгоритмізація як розділ інформатики, який вивчає процеси створення алгоритмів, традиційно відноситься до теоретичної інформатики внаслідок свого фундаментального характеру. Завдяки розвитку інформаційних технологій, і зокрема технологій програмування, з'являється можливість у межах модулю "Моделювання. Основи алгоритмізації" ознайомити з загальнонауковими поняттями інформатики і в той же час формувати та розвивати вміння та навички, необхідні користувачеві під час роботи з сучасним програмним забезпеченням, тобто з'являється можливість зробити цей розділ містком між теоретичною та практичною інформатикою.

Модуль "Моделювання. Основи алгоритмізації " шкільного курсу інформатики має неабияке методологічне значення. Він розкриває важливість алгоритмів, їх роль у функціональному зв'язку понять "інформація-алгоритм-комп'ютер", що визначають процес автоматичного опрацювання інформації. На прикладах демонструється можливість формального виконання алгоритму,елементарність дій, що задаються на виконання виконавцеві за кожною вказівкою алгоритму. Тим самим підкреслюється можливість передавання виконання формально описаного алгоритму виконавцеві-машині, тобто можливість автоматизації діяльності людини на основі алгоритмів

Сучасний підхід до вивчення основ алгоритмізації повинен будуватися на таких основних положеннях:

1.Процес вивчення основ алгоритмізації слід орієнтувати на використання комп'ютера як дидактичного засобу навчання.

2.Метою вивчення основ алгоритмізації є виділення змісту реального об'єкта - алгоритму - і правил його побудови

Поняття алгоритму відіграє провідну роль у формуванні операційного мислення та уявлень учнів про можливість автоматизації різних видів діяльності людини. Введення і розвиток у шкільному курсі інформатики поняття алгоритму та вивчення основних властивостей алгоритмів дозволяє показати одну з найважливіших характеристик алгоритмів - формальність, чисто механічний характер діяльності людини

при їх виконанні, що є основою виконання такого роду операцій за допомогою комп'ютера, тобто основою автоматизації виконання таких операцій.

Таким чином, етап опанування поняття "алгоритм" є першим етапом формування уявлень про автоматичне опрацювання різноманітних даних за допомогою комп'ютера. Варто також звернути увагу і на внутрі-предметні зв'язки - алгоритм можна трактувати як повідомлення про те, як слід розв'язувати задачу, подане у вигляді формально описаного впорядкованого набору вказівок про те, які і в якому порядку слід виконувати операції для того, щоб розв'язати задачу. Слід звернути увагу на те, що якщо поміняти порядок виконання вказівок, то задача швидше за все не буде розв'язана або ж буде розв'язана зовсім інша задача. Володіння поняттям алгоритму одним із найважливіших компонентів інформаційної культури.

Поняття "алгоритм" належить до числа фундаментальних математичних понять і є об'єктом дослідження спеціального розділу математики - теорії алгоритмів.

У змісті навчання основ алгоритмізації можна виділити такі компоненти:

- навчання відомих алгоритмів і їх використання;
- навчання класичних алгоритмів;
- навчання побудови описів алгоритмів як з використанням, так і без використання відомих алгоритмів.

Наведемо один із варіантів класифікації мов, які можуть бути покладені в основу вивчення та застосування для навчання основ алгоритмізації:

- природна мова (словесний опис алгоритму);
- мова графічних схем;
- структура програми;
- мови програмування.

Класифікацію виконавців алгоритмів можна подати так:

- людина;
- комп'ютер, що працює на основі використання компіляторів (інтерпретаторів) мов різних рівнів.

Звідси випливає, що для вивчення відомих алгоритмів і їх використання необхідно передусім вибрати виконавця і вивчити систему його вказівок (мову для запису алгоритмів), тобто набір

операцій, які може виконувати такий виконавець у зв'язку з задачею, яку необхідно розв'язати.

Таким чином, можна сформулювати метод вивчення відомих алгоритмів: повідомляється алгоритм, описаний деякою мовою, а необхідно: 1) записати алгоритм мовою конкретного виконавця (якщо алгоритм повідомлено в словесній формі); 2) виконати алгоритм, використовуючи набір операцій конкретного виконавця.

До класичних алгоритмів можна віднести алгоритми таких типів:

- 1) для створення, зміни і руйнування інформаційних структур (лінійні списки, дерева, графи);
- 2) напівчисельні алгоритми-алгоритми теорії чисел, арифметика багаторазової точності (за Д.Кнутом);
- 3) генерації псевдо випадкових чисел (за Д.Кнутом);
- 4) пошуку і впорядкування (за Д.Кнутом);
- 5) пошуку підрядка в рядкові (за Д.Кнутом);
- 6) стиснення інформації;
- 7) побудови графічних примітивів.

Враховуючи обмеженість навчального часу, всі класичні алгоритми неможливо вивчити на уроках інформатики, з ними слід знайомити учнів на факультативах або при організації профільного навчання.

Наведемо деякі міркування щодо добору вправ для навчання класичних алгоритмів.

1. Вправи на тестування (процес пошуку помилок у програмі). Надається текст програми, в якому описано класичний алгоритм, і словесний опис алгоритму. Потрібно побудувати систему тестів для його перевірки.

2. Вправи на оптимізацію. Для роботи пропонується алгоритм (з можливими помилками). Потрібно виявити помилки і оптимізувати алгоритм за часом виконання.

При навчанні побудови (відкриття) алгоритмів перш за все необхідно з'ясувати питання: в чому полягає процес побудови алгоритму.

Побудувати алгоритм - означає: 1) продумати план деякої майбутньої діяльності; 2) зафіксувати його за допомогою деякої системи позначень, так званою формальною мовою.

Доцільно виділити такі етапи навчання основ алгоритмізації :

1-й етап. Вступ до алгоритмізації. Алгоритми над найпростішими типами даних.

Проектування алгоритмів "згори-донизу". Базові алгоритмічні конструкції: послідовне виконання вказівок, цикл, розгалуження. Побудова алгоритмів методом покрокової деталізації з використанням раніше описаних.

2-й етап. Навчання класичних алгоритмів (раніше описаних) та їх використання.

Зазначимо, що при навчанні основ алгоритмізації потрібно виділяти три класи вправ

1) задано текст задачі і алгоритм її розв'язування, потрібно виконати алгоритм і з'ясувати особливості роботи за алгоритмом (вправи на тестування);

2) задано лише опис алгоритму, потрібно виконати алгоритм і встановити його призначення (вправи на "відгадування" формулювання задачі);

3) задано лише призначення алгоритму, потрібно написати текст алгоритму - скласти алгоритм і виконати його - протестувати (вправи на складання алгоритму).

2. ІНФОРМАЦІЯ ДЛЯ ВИВЧЕННЯ

§1. Необхідність вивчення теми " Моделювання. Основи алгоритмізації "

Вивчення інформатики формує елементи операційного стилю мислення, який полягає в умінні:

- формалізувати задачу;
- виділити в ній логічно самостійні частини;
- визначити зв'язки цих частин;
- спроектувати алгоритм розв'язування за допомогою технологій "згори-донизу" та "знизу-догори";

• дібрати якомога ефективніший шлях отримання розв'язку;

• інтерпретувати та аналізувати результати.

Саме тому формування такого операційного стилю мислення є одним із завдань навчання взагалі і курсу інформатики зокрема. Навчання основ алгоритмізації забезпечує умови для реалізації таких завдань.

Мета навчання основ алгоритмізації - навчити основних способів організації операцій і даних, а також застосування базових алгоритмічних конструкцій при складанні описів алгоритмів розв'язування різноманітних задач.

Під час вивчення основ алгоритмізації основна увага насамперед повинна приділятися:

- виявленню загальних закономірностей і принципів алгоритмізації
- основним етапам розв'язування задач сучасних інформаційних технологій;
- аналізу поставленої задачі, методам формалізації та моделювання реальних процесів та явищ;
- добору виконавця поставленої задачі, виходячи з того, що він є також певним об'єктом із притаманними йому властивостями та набором допустимих операцій, які слід аналізувати з метою правильного та ефективного їх використання;
- методам та засобам формалізованих описів дій виконавця, сучасним засобам їх конструювання та реалізації за допомогою комп'ютера.

§2. Підготовча робота до теми" Моделювання. Основи алгоритмізації "

Змістова лінія моделювання поруч з лінією інформації і інформаційних процесів відноситься до теоретичних основ курсу інформатики. Разом з тим не слід вважати, що тема моделювання носить лише теоретичний характер і відокремлена від всіх інших тем. Програмні засоби інформаційних технологій - СУБД, табличні редактори та інші - слід розглядати як засоби для опрацювання інформаційних моделей. Алгоритмізація має пряме відношення до моделювання.

Головна мета вивчення поняття моделі пов'язана з подальшим розглядом основних етапів розв'язування задач за допомогою комп'ютера.

Доцільно відмітити, що формування правильного розуміння змісту станів розв'язування задач та порядку їх слідування - одна з важливих цілей вивчення курсу інформатики, яка досягається поступово, за мірою вивчення всього навчального матеріалу.

Методика інформаційного моделювання пов'язана з питаннями системології, системного аналізу. Ступінь глибини вивчення цих питань суттєво залежить від рівня підготовленості. Важко сприймається абстрактні, узагальнені поняття. Тому розкриття таких питань повинно спиратися на прості, доступні приклади.

На уроці можна спиратися на те, що поняття алгоритму і блок-схеми не є новими для учнів і потрібно лише систематизувати і уточнити їх знання. Таким чином, не так багато дійсно нових понять потрібно ввести на уроці. Фактично вони стосуються властивостей алгоритму

Щодо походження терміну «алгоритм», то можна зазначити, що він походить від імені видатного вченого Мухамеда аль-Хорезмі (Мухамеда з Хорезму), якому належить значний внесок у розвиток математики. У 825 році він написав «Книгу про індійську арифметику», де було вперше систематично викладено арифметику, засновану на десятковій системі числення із застосуванням нуля, тобто ту арифметику, яка в наш час є звичайною та природною. Арифметика виникла в Індії і тому раніше називалася індійською. В перекладі на латинську «Книги про індійську арифметику» ім'я її автора було написано як «Алгоритми». «Так казав Алгоритми», - починали європейські вчені, посилаючись на викладені в книзі правила виконання арифметичних дій. Ці правила і отримали назву алгоритмів. Тільки у XX-му столітті, у зв'язку зі становленням та розвитком теорії алгоритмів, поняття алгоритму стало вживатися в більш широкому значенні.

Розглядаючи властивості алгоритму, доцільно наголосити надвох важливих обставинах. По-перше, слід зазначити, що через алгоритми передається здобутий досвід

способів діяльності. Використовуючи готовий алгоритм для розв'язання тієї чи іншої задачі, ми спираємося на цей досвід і можемо просуватися далі, вдосконалюючи його або поширюючи на більш складні задачі. По-друге, алгоритм дає рецепт розв'язання задачі у вигляді певної послідовності елементарних дій. Інтелектуальну складову розв'язання задачі сконцентровано на етапі розробки алгоритму, і виконавцеві алгоритму залишається лише точно і формально виконувати заплановані дії. Відокремлення інтелектуальної і операційної складових діяльності є основою автоматизації процесів.

Способи подання алгоритму достатньо розглянути на рівні ознайомлення, далі учні оволодіватимуть на практиці поданням алгоритму у формі блок-схеми. Це надасть їм можливість не тільки засвоїти правила подання алгоритму із застосуванням різних способів, але й переконатися у доцільності застосування того чи іншого способу в залежності від конкретної мети. Для засвоєння матеріалу уроку учням пропонуються завдання теоретичного і практичного характеру. Практичні завдання орієнтовані на розкриття сутності алгоритму і передбачають діяльність учнів з точного виконання приписаних дій для досягнення певної цілі або діяльність із складання і опису алгоритму для розв'язання логічних задач.

§3. Моделювання. Основи алгоритмізації

1. Поняття моделі. Типи моделей.
2. Моделювання як метод дослідження об'єктів. *Практична робота №1.* Побудова інформаційної моделі.
3. Поняття алгоритму. Властивості алгоритмів. Форми подання алгоритму. Виконавець алгоритму. Система команд виконавця алгоритму.
4. Базові структури алгоритмів: слідування, розгалуження, повторення. Графічні схеми базових структур алгоритмів. Поняття про конструювання алгоритмів різними методами.
5. Етапи розв'язування задачі за допомогою комп'ютера. Вхідні дані та результати, їх взаємозв'язок. Поняття про метод розв'язування задачі. Поняття програми.

§1. ПОНЯТТЯ МОДЕЛІ. МОДЕЛЮВАННЯ.

1. Моделі. Види та типи моделей.
2. Інформаційна модель.
3. Карта пам'яті. Редактори карт пам'яті.
4. Математична модель.
5. Алгоритм створення інформаційної моделі.

МОДЕЛІ. ВИДИ ТА ТИПИ МОДЕЛЕЙ

Діяльність людини, спрямовану на вивчення властивостей об'єктів навколишнього світу та їх зв'язків з іншими об'єктами, називають **дослідженням**.

Предметна область – це множина всіх предметів, властивості яких і відношення між якими розглядаються в межах деякого дослідження або в процесі діяльності.

Наприклад, під час дослідження туристичних маршрутів, які можна прокласти для ознайомлення з історією та культурою рідного краю, до предметної області входять транспортні шляхи, визначні місця, історичні та культурні пам'ятки краю.



Модель (від лат. *modulus* – міра, взірець, норма) – це новий об'єкт, який має властивості даного об'єкта, суттєві для певного дослідження.



Авіаційний тренажер – модель кабіни літака

Оригінал – об'єкт, для заміщення якого призначається модель.

Моделювання – процес створення та дослідження моделі об'єкта.

Модель завжди є штучно створеним людиною об'єктом, який дає **ідеалізоване уявлення** про об'єкт-оригінал. Ступінь ідеалізації моделі визначає межі її адекватності, тобто відповідності оригіналу в рамках поставленої задачі.

Моделі створюють у випадках, коли:

- 1) сам об'єкт недоступний або важкодоступний і його неможливо або важко дослідити безпосередньо (**наприклад**, зірка сузір'я Великої Ведмедиці, Сонячна корона, процес розпаду ядра або виверження вулкана);
- 2) дослідження об'єкта можуть призвести до його руйнування (**наприклад**, дослідження міцності мостового переходу при різних навантаженнях або будинків під час землетрусів різної сили);
- 3) виготовлення об'єкта потребує значних коштів (**наприклад**, забудова нового мікрорайону);
- 4) об'єкти мають багато властивостей і деякі з них є несуттєвими для даного дослідження (**наприклад**, колір будинку не впливає на дослідження його міцності)

Види моделей

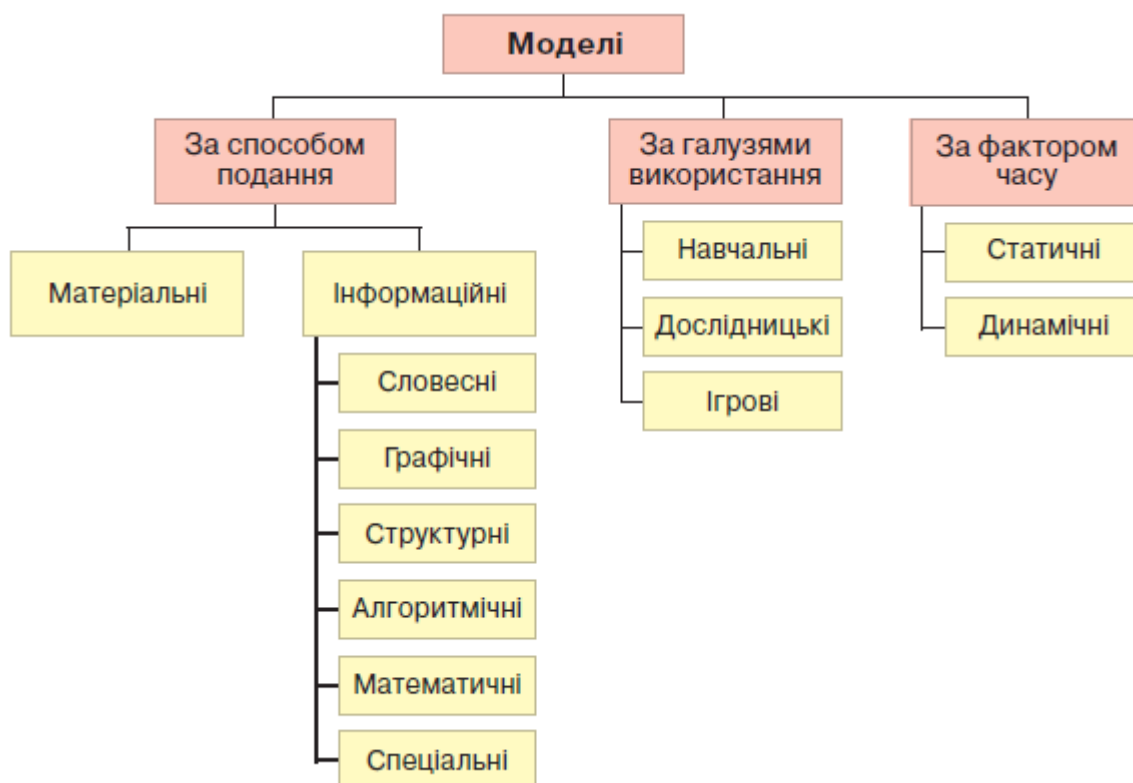
Моделлю предмета може бути його зменшена копія, креслення, яке відображає його структуру чи склад, схема, що відбиває взаємозв'язки окремих елементів. **Наприклад**, макет корабля, креслення будівлі, схема комп'ютера.

Модель явища або процесу відтворює його хід, послідовну зміну стану, етапи розвитку. **Наприклад**, модель еволюції людства, модель розвитку економіки тощо.

Типи моделей

Моделі **класифікують за різними ознаками**:

- 1) за способом подання;
- 2) за галузями використання;
- 3) за фактором часу



За способом подання моделі розподіляють на матеріальні (предметні) та інформаційні.

Матеріальна модель об'єкта – це модель об'єкта, подана у вигляді матеріального об'єкта (предмета).

Матеріальні моделі **призначені** для проведення **практичних досліджень**.

Приклади: іграшки, опудала тварин, манекени, муляжі, глобус, модель водяного млина

Інформаційна модель об'єкта – це модель об'єкта, подана у вигляді його опису.

Інформаційні моделі **призначені** для проведення **теоретичних досліджень**.

Приклади: фізична карта України, рівняння хімічної реакції, математична функція, розповідь про береги Дніпра.

Інформаційні моделі у свою чергу **розподіляють на:**

- **словесні** (наприклад, усні та письмові описи);
- **графічні** (наприклад, рисунки, креслення, піктограми, карти та ін.);
- **структурні** (наприклад, таблиці, графіки залежностей, діаграми, схеми та ін.);
- **алгоритмічні** (наприклад, правила, плани дій та ін.);
- **математичні** (наприклад, формули, рівняння, нерівності, функції та ін.);
- **спеціальні** (наприклад, хімічні формули і рівняння, нотні записи, записи шахових партій та ін.).



Матеріальна модель
земної кулі

Інформаційна модель земної кулі

За галузями використання моделі розподіляють на:

- **навчальні** – створюються для навчання (**наприклад**, наочні посібники, тренажери, що створюють для навчання, наприклад тренажер літака для навчання льотчиків, тренажер для імітації перевантажень під час виведення на орбіту в космічному польоті та ін.);
- **дослідницькі** – створюються для проведення досліджень (**наприклад**, модель річкового теплохода, гелікоптера, гідроелектростанції для проведення досліджень на міцність, плавучість, аеродинамічні властивості, вплив на екологію, модель комп'ютера, прилад, що імітує розряд блискавки та ін., а також формули, рівняння, функції та ін., які можна використати для дослідження об'єктів);
- **ігрові** – створюються для моделювання ситуацій і перевірки різноманітних стратегій поведінки, адаптації до певних умов (**наприклад**, військові, економічні, спортивні, ділові ігри та ін.).

За фактором часу моделі розподіляють на:

- **статичні** – це моделі, в яких зафіксовано стан об'єкта в певний момент часу і подальші зміни цього об'єкта не враховуються, **наприклад** гіпсова модель піраміди Хеопса;
- **динамічні** – це моделі, які призначені для дослідження змінень значень властивостей об'єкта з часом, **наприклад**, модель змінень рівня води в гірських річках під час танення снігів; модель погоди в даному регіоні; модель Сонячної системи, що змінюється залежно від результатів нових досліджень.


ІНФОРМАЦІЙНА МОДЕЛЬ

Інформаційна модель – це сукупність цілеспрямовано відібраної інформації про властивості об'єкта, його стан і зв'язки із зовнішнім світом, які є суттєвими для даного дослідження.

Один і той самий об'єкт може бути представлений різними моделями в різних дослідженнях.

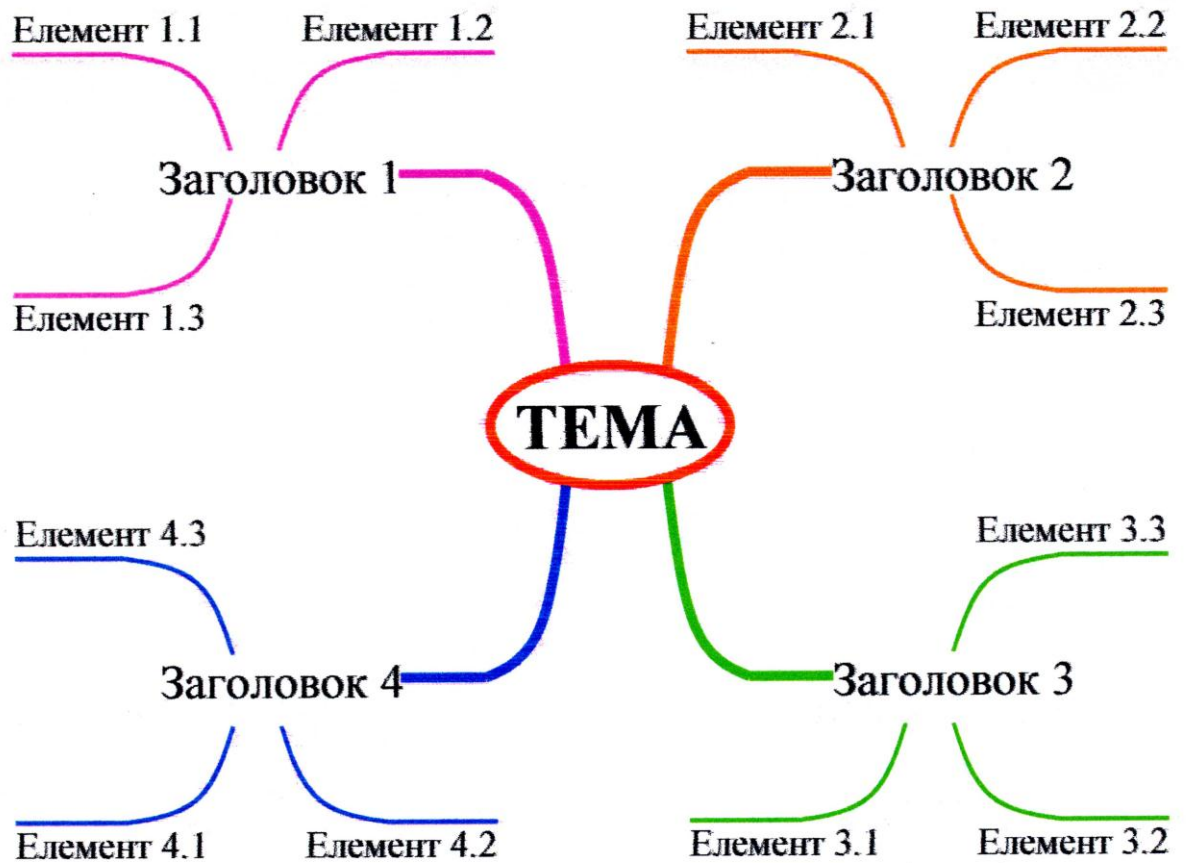
Одна й та сама модель може слугувати для дослідження різних прототипів.

Форми подання інформаційних моделей

Форма подання моделі	Приклад												
Словесна	<p>Письмовий опис речовини</p> <p>Прозора безбарвна рідина без запаху та смаку. У природі існує у трьох агрегатних станах: твердому (лід), рідкому (вода) і газуватому (водяна пара)</p>												
Графічна	<p>Зображення молекули води</p> 												
Структурна	<p>Таблиця</p> <p style="text-align: center;">Найважливіші фізичні константи води</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0ffe0;"> <th>Показник</th> <th>Константа</th> </tr> </thead> <tbody> <tr> <td>Густина</td> <td></td> </tr> <tr> <td>• лід (0 °C)</td> <td style="text-align: center;">917 кг/м³</td> </tr> <tr> <td>• вода (0 °C)</td> <td style="text-align: center;">999 кг/м³</td> </tr> <tr> <td>Температура плавлення</td> <td style="text-align: center;">0 °C</td> </tr> <tr> <td>Температура кипіння</td> <td style="text-align: center;">100 °C</td> </tr> </tbody> </table>	Показник	Константа	Густина		• лід (0 °C)	917 кг/м ³	• вода (0 °C)	999 кг/м ³	Температура плавлення	0 °C	Температура кипіння	100 °C
Показник	Константа												
Густина													
• лід (0 °C)	917 кг/м ³												
• вода (0 °C)	999 кг/м ³												
Температура плавлення	0 °C												
Температура кипіння	100 °C												
Математична	<p>Формула для обчислення об'єму води в акваріумі</p> $V = abc,$ <p>де a, b, c – довжина, ширина та висота акваріума (у см)</p>												
Спеціальна	<p>Формула молекули води</p> H_2O <p>Рівняння хімічної реакції утворення води</p> $2\text{H}_2 + \text{O}_2 = 2\text{H}_2\text{O}$												

КАРТА ПАМ'ЯТІ. РЕДАКТОРИ КАРТ ПАМ'ЯТІ

Карта пам'яті – це інформаційна модель, подана в структурній формі у вигляді схеми, інформація якої зображується ієрархічно у вигляді радіального рисунка (ближче до центру – більш важливі поняття, далі від центру – менш важливі поняття), в якій за допомогою ліній відображаються зв'язки між різними поняттями та окремими частинами теми.



Карта пам'яті

Карти пам'яті також називають **ментальними, асоціативними картами, картами розуму, картами знань**. Часто використовують англійську назву **mind map** (англ. mind map – карта розуму).

Автором теорії карт пам'яті є британський психолог **Тоні Бьюзен** (1942 р.н.). Він розробив методику запам'ятовування та організації мислення, завдяки якій установив рекорд у запам'ятовуванні великих обсягів даних.



Тоні Бьюзен

Використовуючи карти пам'яті, ми **отримуємо** такі **можливості**:

- навчання: створення конспектів лекцій, читання книг та підручників, написання рефератів;
- запам'ятовування: підготовка до контрольних робіт, запам'ятовування списків, дат, номерів телефонів;
- планування часу, виступу, розробка нових ідей, подій, складання різних планів;
- генерування ідей;
- прийняття рішень;
- структурування та опрацювання даних.

Редакторами карт пам'яті є програми **FreeMind**, **Freeplain**, **Edraw Mind Map**, **TheBrain**, **XMind** та інші. Створювати карти знань можна також з використанням інтернет-ресурсів, наприклад **Bubbl.us** (*bubbl.us*), **Mind42.com** (*mind42.com*), **Glinkr** (*glinkr.net*) та інших.

Можливості редакторів карт пам'яті:

- створення карти пам'яті;
- редагування та форматування карти пам'яті;
- додавання графічних об'єктів до вузлів карти пам'яті;
- додавання нотаток до вузлів карти пам'яті;
- створення гіперпосилань на допоміжні ресурси;
- керування показом карти пам'яті;
- пошук даних у карті пам'яті;
- відкриття, збереження, друкування карти пам'яті та інші.

МАТЕМАТИЧНА МОДЕЛЬ

Математична модель – це інформаційна знакова модель, яка являє собою математичний опис взаємозалежностей між параметрами об'єкта.

Математичні моделі **подаються сукупністю математичних формул**.

Математичні моделі **є основою комп'ютерного моделювання**.

АЛГОРИТМ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ МОДЕЛІ

1. Визначити мету створення моделі.
2. Визначити предметну область дослідження, для якого створюється модель.
3. Виділити ті властивості об'єкта або явища, які є суттєвими для дослідження.
4. Установити взаємозв'язки між значеннями вибраних властивостей, між даним об'єктом або явищем та іншими об'єктами та явищами предметної області, що є суттєвими для дослідження.
5. Вибрати форму подання інформаційної моделі.
6. Вибрати засіб, з використанням якого буде подано модель.
7. Зобразити модель відповідно до вибраної форми подання, відобразивши в ній суттєві властивості та встановлені взаємозв'язки.



Рис. 2.5. Алгоритм побудови інформаційної моделі

ПОБУДОВА ІНФОРМАЦІЙНИХ МОДЕЛЕЙ

Задача1.

Побудуйте інформаційну модель океанічних западин.

Розв'язування.

1. **Мета створення моделі** – вивчення океанічних западин та їх розподілу по океанах.

2. **Предметна область** охоплює всі океани Землі. Міжнародна гідрографічна організація з 2000 року виділяє п'ять океанів. Саме такою будемо вважати **предметну область** під час побудови нашої моделі.

Для створення моделі використаємо дані, отримані з інтернет-енциклопедії Вікіпедія.

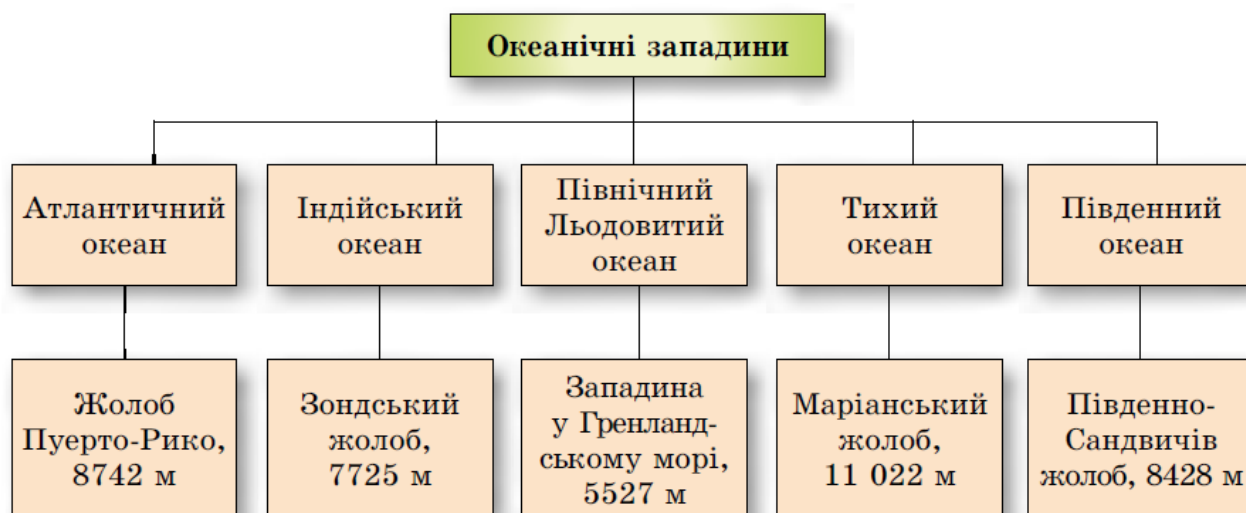
3. **Суттєвими властивостями** об'єктів моделювання є назви западин, їх глибина, належність западин до океанів. **Несуттєві властивості** – солоність води в западинах, морські мешканці западин, океанічні течії та інші.

4. У моделі суттєвими є взаємозв'язки між западинами та океанами, у яких розміщені ці западини.

5. Виберемо для цієї моделі **структурну форму подання у вигляді схеми**.

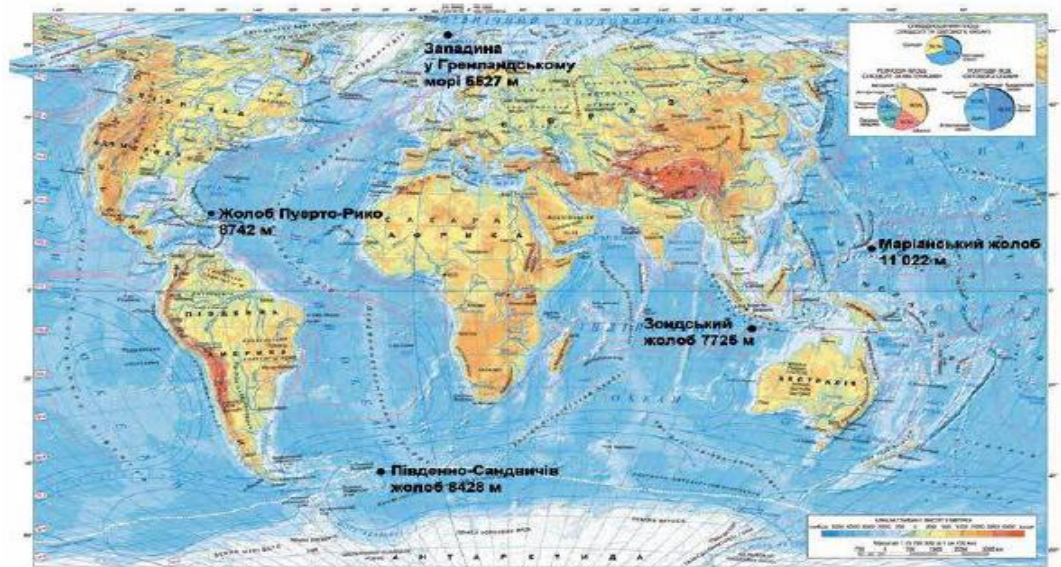
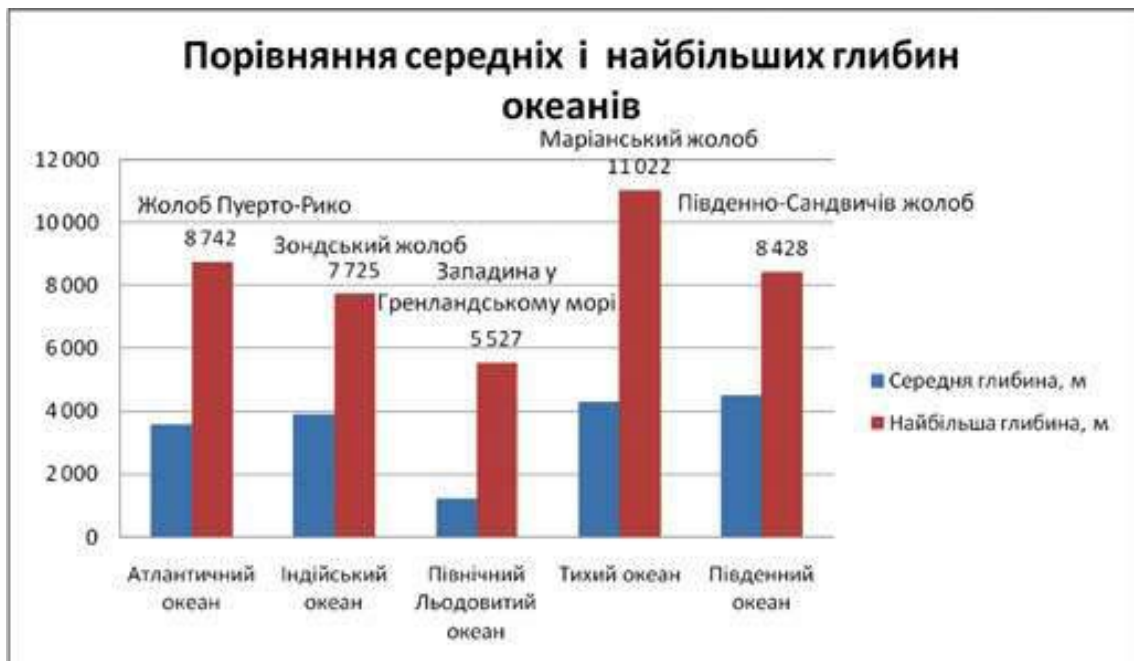
6. Зобразити структурну модель можна в **текстовому процесорі**, який містить засоби для вставлення схем.

7. Як **результат**, отримуємо модель, наведену на рисунку.



Для моделі можуть використовуватися інші форми подання: таблиця, у якій наведено дані про западини, діаграма порівняння середніх і найбільших глибин океанів, карта з нанесеними позначеннями глибин океанічних западин тощо.

<i>Океан</i>	<i>Найбільша глибина, м</i>	<i>Назва або розміщення западини</i>
Атлантичний	8742	Жолоб Пуерто-Рико
Індійський	7725	Зондський жолоб
Північний Льодовитий	5527	У Гренландському морі
Тихий	11 022	Маріанський жолоб
Південний	8428	Південно-Сандвичів жолоб



Задача 2.

Побудуйте інформаційну модель об'єкта ромашка лікарська.

Розв'язування.

1. **Мета побудови моделі** – вивчення будови рослин.
2. **Предметна область** охоплює рослинний світ.
3. **Суттєвими властивостями** є будова органів рослини: коріння, стебла, листків, квітів. Несуттєві властивості: запах, час цвітіння, лікарські властивості, місця розповсюдження тощо, – оскільки вони не стосуються будови рослини.
4. Для цієї моделі **суттєвим** є взаємне розташування органів рослини.
5. **Виберемо** для цієї моделі **графічну форму** подання, яка наочно відобразить суттєві властивості об'єкта дослідження.

6. **Засіб** для подання графічної моделі – **графічний редактор**.

7. Під час відтворення графічної моделі потрібно зобразити органи рослини за таким описом: *стебло* – пряме, циліндричне, голе, розгалужене; *листки* – чергові, розсічені; *квітки* – білі пелюстки, жовті серединки; *коріння* – стрижневе, малорозгалужене, тонке, неглибоко проникає у ґрунт.

Як **результат**, отримаємо графічне подання моделі, зображене на рисунку.



Задача 3. Побудуйте математичну модель розв'язання задачі.

У одній коробці в мене є жуки,
у другій такій коробці — павуки.
Небагато їх, і легко полічити:
павуків з жуками разом шість.
Став лічити, скільки всього ніг,
але швидко це зробити я не зміг.
Сорок ніг я налічив, нарешті, там
і загадку всім задати хочу вам:
— Відгадайте, скільки маю я жуків
і окремо скільки в мене павуків?

Розв'язування.

Розв'яжемо цю задачу в загальному вигляді, позначивши кількість жуків через x . Тоді кількість павуків буде $(6 - x)$. Кількість ніг у жуків $6x$, а в павуків — $8(6 - x)$. Враховуючи, що разом у жуків і павуків 40 ніг, складемо рівняння:

$$6x + 8(6 - x) = 40$$

Одержане **рівняння є математичною моделлю** для розв'язання даної задачі.

Розв'язавши рівняння, визначимо, що жуків було 4, а павуків — 2.

ПИТАННЯ

1. Вставте пропущені слова у наведених нижче визначеннях:

а) Предметна область – це _____ всіх _____, властивості яких і відношення між якими розглядаються в межах деякого дослідження або в процесі діяльності.

б) Модель – це _____, який має властивості даного об'єкта, суттєві для певного дослідження.

в) Оригінал – це об'єкт, для _____ якого призначається модель.

г) Моделювання – _____ створення та дослідження моделі об'єкта.

Відповіді:

а) множина, предметів;

б) новий, об'єкт;

в) заміщення;

г) процес.

2. Визначте, для яких випадків могла бути створена модель літака:

а) сам об'єкт недоступний або важкодоступний і його неможливо або важко дослідити безпосередньо;

б) дослідження об'єкта можуть призвести до його руйнування;

- в) виготовлення об'єкта потребує значних коштів;
- г) об'єкти мають багато властивостей і деякі з них є несуттєвими для даного дослідження.

Відповіді: а, б, в, г.

3. Визначте, які з наведених моделей є матеріальні, а які інформаційні:

- а) рисунок піраміди;
- б) намальований план школи;
- в) формула площі прямокутника;
- г) нотний запис пісні;
- д) макет забудови ділянки;
- е) кулінарний рецепт;
- є) хімічний дослід;
- ж) модель літака;
- з) комп'ютерна модель автомобіля;
- и) модель атома;
- і) запис шахової партії;
- ї) проведення фізичного дослідження;
- й) рівняння хімічної реакції;
- к) опудало птаха;
- л) сценарій проведення свята;
- м) глобус?

Відповідь:

Матеріальні: д, є, ж, і, к, м

Інформаційні: а, б, в, г, з, и, і, й, л

4. Визначте, до якого типу за способом подання належать зазначені моделі в класифікації:

- а) список учнів вашого класу з відомостями про кожного;
- б) іграшкова залізниця;
- в) макет шкільного будинку;
- г) план проведення дня народження;
- д) звіт про похід;
- е) перелік необхідних покупок у магазині?

Відповідь:

Матеріальні: б, в

Інформаційні: а, г, д, е

5. Визначте, до якого типу належать наведені нижче інформаційні моделі:

- а) карта України;
- б) сценарій п'єси;
- в) формула закону Ома;
- г) ноти нової пісні;
- д) рівняння хімічної реакції;
- е) таблиця результатів чемпіонату України з баскетболу;
- є) автобіографія?

Відповідь:

Словесні: є

Графічні: а

Структурні: є

Алгоритмічні: б

Математичні: в

Спеціальні: г, д

6. Визначте, яка модель створюється для перевірки різноманітних стратегій поведінки.

- а) навчальні;
- б) дослідницькі;
- в) ігрові.

Відповідь: в

7. Вставте пропущені слова у наведених нижче визначеннях:

а) Карта пам'яті – це _____ модель, подана в _____ формі у вигляді схеми, інформація якої зображується ієрархічно у вигляді _____ рисунка (ближче до центру – більш важливі поняття, далі від центру – менш важливі поняття), в якій за допомогою _____ відображаються зв'язки між різними поняттями та окремими частинами теми.

б) _____ модель – це інформаційна знакова модель, яка являє собою математичний опис взаємозалежностей між параметрами об'єкта.

Відповідь:

- а) інформаційна, структурній, радіального, ліній;
- б) математична.

8. Складіть алгоритм створення інформаційної моделі.

- б) Визначити мету створення моделі.
- е) Визначити предметну область дослідження, для якого створюється модель.
- г) Виділити ті властивості об'єкта або явища, які є суттєвими для дослідження.
- е) Установити взаємозв'язки між значеннями вибраних властивостей, між даним об'єктом або явищем та іншими об'єктами та явищами предметної області, що є суттєвими для дослідження.

- а) Вибрати форму подання інформаційної моделі.
- д) Вибрати засіб, з використанням якого буде подано модель.
- в) Зобразити модель відповідно до вибраної форми подання, відобразивши в ній суттєві властивості та встановлені взаємозв'язки.

Відповідь: б, е, г, е, а, д, в.

ЗАВДАННЯ

1. Побудуйте карту пам'яті з відомостями про *Велике переселення народів* за даними, які отримано із сайту **Школяр України** (shkolyar.in.ua/narodjennya-serednyovichniyi-uevropy7):

У IV–VII ст. відбулося Велике переселення народів, яке призвело до зіткнення між двома світами – римським (де відбувався поступовий занепад імперії внаслідок економічної кризи, міжусобних воєн, послаблення старої віри, скорочення населення) та варварським (де не існувало держав і законів). Це сприяло загибелі Західної Римської імперії в 476 році. На уламках імперії утворилися п'ять великих варварських королівств: Вандальське, Вестготське, Остготське, Франкське і Бургундське.

На карті пам'яті відобразіть період, коли відбувалося Велике переселення народів, його причини та наслідки. Збережіть створений документ.

- 2. Складіть карту пам'яті «Складові комп'ютера».
- 3. Побудуйте карту пам'яті про агрегатні стани води.
- 4. Створіть інформаційну модель кубіка Рубіка.
- 5. Створіть інформаційну модель об'єктів «квітка» і «кавун» з точки зору художника, медика, ботаніка, постачальника, покупця.

6. Створіть інформаційну модель об'єкта «дачний будинок» з точки зору художника, будівельника, покупця.

7. Відстань між містами A і B дорівнює 50 км. З цих міст одночасно назустріч один одному виїхали два велосипедисти зі швидкостями 15 км/год і 12 км/год відповідно. Створіть математичну модель для визначення відстані між ними у будь-який момент часу до їхньої зустрічі.

8. Один оператор може набрати рукопис за 20 год, а другий – за 30 год. Створіть математичну модель для визначення частини рукопису, яку вони наберуть через x год, якщо працюватимуть одночасно.

ПРАКТИЧНА РОБОТА **«Побудова інформаційної моделі»**

***Увага!** Під час роботи з комп'ютером дотримуйтеся правил безпеки та санітарно-гігієнічних норм.*

Створіть інформаційну модель для вивчення вимірювальних приладів на уроках фізики.

1. Зобразіть у текстовому процесорі або в редакторі карт пам'яті структурну модель для вивчення вимірювальних приладів. Відобразіть у моделі назви та призначення приладів (для вимірювання часу: метроном, секундомір, годинник; для вимірювання лінійних розмірів: лінійка; для вимірювання об'єму: мензурка); одиниці вимірювання відповідних фізичних величин. Збережіть отриманий документ у вашій папці у файлі з іменем **практична 1_1**.

2. Зобразіть у графічному редакторі графічну модель одного з вимірювальних приладів. Збережіть отримане зображення у вашій папці у файлі з іменем **практична 1_2**.

3. Створіть презентацію *Вимірювальні прилади*:

1) Скопіюйте з текстового документа структурну модель вимірювальних приладів і вставте її на другий слайд презентації.

2). Уставте на третій слайд презентації створену графічну модель вимірювального приладу.

3). На четвертому слайді презентації запишіть одиниці вимірювання часу (години, хвилини, секунди) і співвідношення між ними.

4). Збережіть отриману презентацію у вашій папці у файлі з іменем **практична 1_3**.

6. Ривкінд Й.Я. **Інформатика: підручник для 7 класів загальноосвітніх навчальних закладів** / Й.Я. Ривкінд, Т.І. Лисенко, Л.А. Чернікова, В.В. Шакоцько – К.: «Генеза», 2015

7. Електронний ресурс. Режим доступу: <https://yadi.sk/d/gFRB0m6m9ZAKe>

§2. ОСНОВИ АЛГОРИТМІЗАЦІЇ

Поняття алгоритму. Властивості алгоритмів. Форми подання алгоритму. Виконавець алгоритму. Система команд виконавця алгоритму

Кожна людина щодня зустрічається з безліччю задач від найпростіших і добре відомих до дуже складних. Для багатьох задач існують визначені правила (інструкції, команди), що пояснюють виконавцю, як розв'язувати дану проблему. Ці правила людина може вивчити чи заздалегідь сформулювати сама в процесі розв'язування задачі. Чим точніше описані правила, тим швидше людина опанує ними і буде ефективніше їх застосовувати. У нашому житті ми постійно складаємо опис деякої послідовності дій для досягнення бажаного результату, тому поняття алгоритму не є для нас чимось новим і незвичайним. Ви вже не раз зустрічалися з алгоритмами в інших шкільних предметах. Наприклад, у хімії отримання тієї чи іншої сполуки можна описати за допомогою алгоритму. Але найбільше прикладів алгоритмів у математиці — науці, у якій власне й зародилося це поняття. По суті, математика вивчає різні алгоритми і створює нові. До алгоритмів зі шкільного курсу математики належать правила виконання арифметичних дій, правила знаходження розв'язків рівнянь тощо. У вигляді алгоритмів можна сформулювати правила побудови різних геометричних фігур (згадайте задачу на побудову), а також рекомендації щодо розв'язування типових задач.

До слова «алгоритм» близькі за значенням слова: спосіб, рецепт. Однак алгоритми в інформатиці — це не тільки рецепти розв'язування задач. Алгоритми розробляють, насамперед, із метою автоматизації дій виконавця. Кожний з нас використовує сотні різних алгоритмів (алгоритми виконання арифметичних дій, розв'язування задач, прибирання квартири, миття посуду, готування їжі - рецепти тощо). Отже, давайте сформулюємо визначення, що ж таке алгоритм.

Алгоритм – скінчена однозначно визначена послідовність операцій, формальне виконання яких приводить до розв'язання певної задачі за кінцеве число кроків;

зрозуміле і точне розпорядження виконавцю виконати послідовність дій, спрямованих на досягнення зазначеної мети чи на розв'язання поставленої задачі.

Термін «алгоритм» походить від імені видатного вченого середньовічного Сходу **Мухаммеда бен-Муси аль –Хорезмі (783-850)**, який у своїх наукових працях з математики, астрономії та географії описав і використовував індійську позиційну систему числення, а також сформулював у загальному вигляді правила виконання чотирьох основних арифметичних дій. Європейські вчені ознайомилися з його працями завдяки їх перекладам на латину. При перекладі ім'я автора було подано як **Algorithmus**. Звідки й пішло слово «алгоритм».

Вчинки людей підпорядковані досягненню конкретної мети. Все залежить від того, якої мети ми намагаємося досягнути. **Наприклад:** миття посуду (виконавці - людина або посудомийна машина), покупка деяких товарів (один із членів родини), розв'язування математичної задачі (учень або комп'ютер) тощо.

Виконавець алгоритму – жива чи нежива істота, яка може виконати всі вказівки заданого алгоритму

У кожного виконавця може бути свій набір команд. Наприклад, речення «Розфарбувати!» може бути командою для людини, яка має в руках пензлик із фарбою, знає, що треба розфарбувати, й уміє це робити. В іншому разі виконавець не зможе виконати таку команду.

Система команд виконавця – сукупність припустимих команд виконавця.

Виконавець має свій набір команд, кожна з яких йому зрозуміла і він навчений їх виконувати. Наприклад, виконавець службовий собака вміє виконувати певні команди людини. А інший виконавець – дворовий собака – деякі команди виконати не зможе.

Припустимі команди – команди, які можуть бути виконані виконавцем.
Неприпустимі команди – команди, які не можуть бути виконані виконавцем.

Поняття алгоритму в інформатиці є фундаментальним, тобто таким, котре не визначається через інші ще більш прості поняття (для порівняння у фізиці - поняття простору і часу, у математиці - точки). Будь-який виконавець (і комп'ютер зокрема) може виконувати тільки обмежений набір операцій (екскаватор копає яму, вчитель вчить, комп'ютер виконує арифметичні дії). Тому алгоритми повинні мати наступні властивості.

Властивості алгоритмів

1. **Дискретність (скінченність).** Виконання кожного алгоритму повинно завершуватися за скінченну кількість кроків. Тільки закінчивши виконання однієї команди, виконавець переходить до виконання іншої.

2. **Результативність.** Виконання алгоритму завжди повинно приводити до певного результату. Воно не може закінчуватися невизначеною ситуацією або ж не закінчуватися взагалі. Будь-який алгоритм передбачає, що його виконання при допустимих початкових даних за число кроків приведе до очікуваного результату.

3. **Формальність.** Будь-який виконавець, який володіє заданою системою команд виконавця, може виконати заданий алгоритм, не вдаючись до суті задачі. Ця властивість має особливе значення для автоматизованого виконання алгоритмів.

4. **Визначеність (однозначність).** Будь-який алгоритм потрібно описати так, щоб під час його виконання у виконавця не виникло двозначних указівок. Тобто різні виконавці згідно з алгоритмом повинні діяти однаково та одержати один і той самий результат.

5. **Масовість.** За допомогою створеного алгоритму можна розв'язувати цілий клас задач.

6. **Зрозумілість.** В алгоритмі повинні бути лише ті вказівки, які знайомі виконавцеві.

Процес алгоритмізації – визначення елементарних дій та порядку їх виконання для розв'язання поставленої задачі. Різні способи записування алгоритмів застосовується для подання алгоритму у вигляді, який однозначно розуміється і розробником, і виконавцем алгоритму.

Форми подання алгоритмів

1. Словесно-природні мови: усна, письмова
2. Словесно-формульна - мова математичних формул, мова хімічних процесів, навчальна алгоритмічна мова.
3. Графічна-метод блок схем, метод структурних схем.
4. Однією з мов програмування.
5. Інші-нотна грамота

Перш ніж приступити до складання блок-схеми, необхідно:

1. Регламентувати складу входу і виходу, тобто визначити імена вхідних даних, проміжних і вихідних результатів.
2. Дати найменування основному алгоритму і допоміжним алгоритмам.

Приклад алгоритму

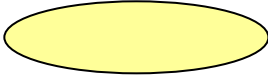

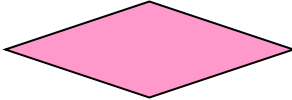
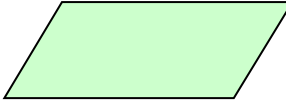

Опишемо алгоритм поділу відрізка АВ навпіл за допомогою циркуля й лінійки.

1. Установіть ніжку циркуля в точці А.
2. Другу ніжку циркуля встановіть у точці В.
3. Окресліть коло.
4. Установіть ніжку циркуля в точці В, не змінюючи його розхил.
5. Окресліть коло.
6. Через точки перетину кіл за допомогою лінійки проведіть пряму.
7. Серединою відрізка АВ є точка перетину проведеної прямої з цим відрізком.

Графічний спосіб представлення алгоритмів є більш компактним та наочним в порівнянні з словесним чи формульним. При графічному представленні алгоритм

створюється у вигляді послідовності зв'язаних між собою функціональних блоків, кожен з яких відповідає одному декільком дій.

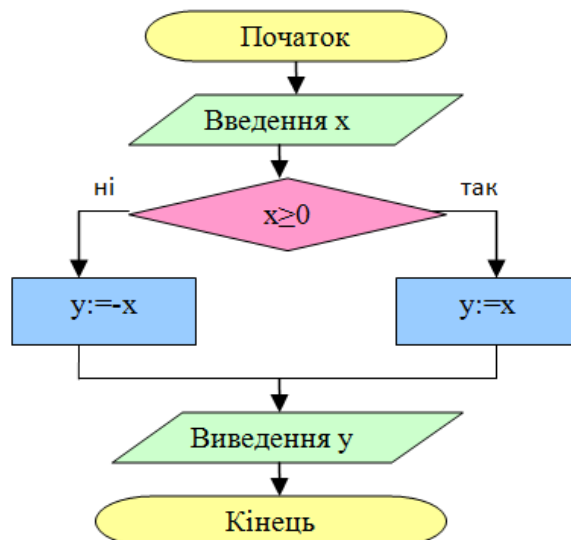
Блок-схема алгоритму – це графічне зображення алгоритму у вигляді спеціальних блоків з необхідними словесними поясненнями.

Найменування	Позначення	Функція
Термінатор		Блок Початок або кінець алгоритму визначає початок та кінець алгоритму
Процес. (Обчислення)		Блок процес визначає зміну значення, форми подання або розташування даних
Розгалуження. (Перевірки умови)		Блок перевірки умови визначає подальші кроки виконання алгоритму залежно від виконання умови
Дані		Блок введення-виведення визначає введення даних в програму або виведення (результатів)
Модифікатор.		Блок модифікатор дозволяє компактно відобразити цикл

Розглянемо словесний спосіб запису на простому прикладі. Знайти модуль величини $X(|X|)$ і надати цього значення змінній Y . Під час побудови алгоритму скористаємося визначенням модуля: $|X| = X$ при $X \geq 0$ і $|X| = -X$ при $X < 0$. Алгоритм можна записати у такий спосіб.

1. Початок.
2. Вести числове значення величини X .
3. Якщо $X \geq 0$, то Y надати значення X , інакше Y надати значення $-X$.
4. Вивести значення Y .
5. Кінець.

Наведемо графічний спосіб представлення цього алгоритму.



Питання для самоперевірки

1. Що таке алгоритм? Дайте визначення цього поняття.
2. Назвіть виконавців для таких алгоритмів:
 - a) Спосіб розв'язання задачі, що записує на дошці вчитель;
 - b) Інструкція про те, як здійснити розбір слова за будовою;
 - c) Інструкція виготовлення годівниці для птахів.
3. Назвіть властивості алгоритму та поясніть їхнє значення.
4. Чи можуть різні виконавці отримати різні результати роботи одного алгоритму?
5. Які переваги графічного зображення алгоритмів перед словесним записом?
6. Назвіть компоненти блок-схем алгоритмів.

Базові структури алгоритмів: слідування, розгалуження, повторення.

Графічні схеми базових структур алгоритмів.

Поняття про конструювання алгоритмів різними методами

Навіть ще не маючи досвіду в створенні алгоритмів, ми інтуїтивно розуміємо, що вони відрізняються за своєю структурою. Так, є алгоритми, що виконуються за будь-яких обставин. Але таке трапляється нечасто, оскільки людина завжди коригує свої плани залежно від навколишніх умов, і тому виникає ситуація «якщо трапиться...», «якщо зустрінуся...», «якщо встигну...».

Для опису логічно зумовленого ходу виконання дій при складанні будь-якого алгоритму використовуються одні й ті самі елементи, що називаються базовими структурами. Їх усього три: **слідування, розгалуження, повторення**. Із цих елементів можна скласти будь-який алгоритм.

1. Базова структура слідування - це така форма організації дій, при якій всі дії виконуються одна за однією по черзі. Вказівка S подається у вигляді послідовності двох (або більше) виконуваних одна за одною простіших вказівок S_1, S_2, \dots, S_n



2. Базова структура розгалуження (розвилка) - це така форма організації дій, при якій в залежності від виконання чи невиконання деякої умови виконується чи одна, чи інша послідовність дій.

Умова - означає перевірку значення логічного виразу та вибір одного з двох варіантів дій, залежно від значення логічного виразу. В логічному виразі можуть використовуватись логічні операції «не» (інверсія), «і» (кон'юнкція), «або» (диз'юнкція).

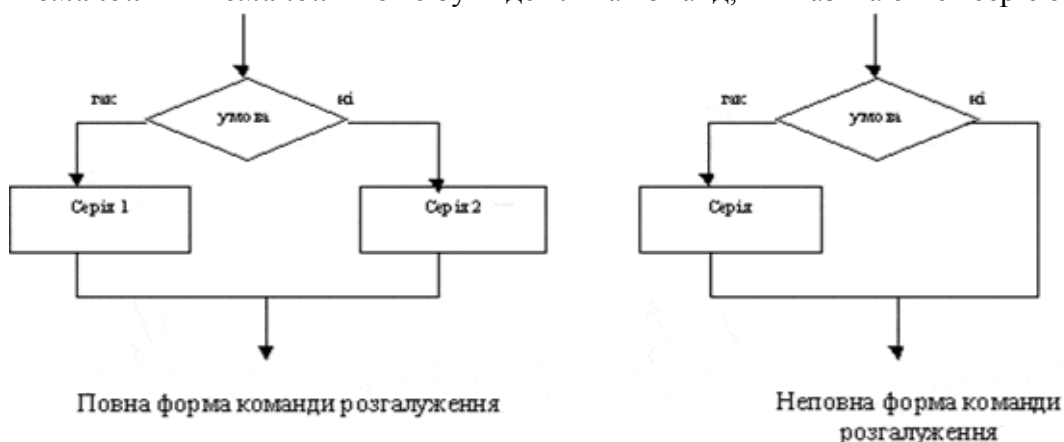
Умова називається **складною**, якщо вона містить кілька простих умов об'єднаних між собою логічними операціями «не», «і», «або». Логічна операція ("**або**") використовується у тих випадках, коли необхідно виконання хоча б однієї з умов, тобто хоча б одна з умов являється істиною. Наприклад, діти залишаються вдома (не йдуть до школи), якщо "сьогодні вихідний" або "сьогодні канікули" або "сьогодні свято" або "дитина хвора". Ясно, що зовсім не обов'язково, щоб сьогодні були одночасно і канікули, і свято, і вихідний та ще й хвороба, щоб не піти до школи. Логічна операція ("**і**") навпаки, використовується лише в тих випад-

ках, коли тільки одночасне виконання всіх умов призводить до результату. Наприклад, тільки у випадку вже зроблених домашніх справ *та* виконаних домашніх завдань *та* гарної погоди у нас є надія, що мама дозволить погуляти з подругою. Логічна операція ("не") використовується в тих випадках, коли є логічне заперечення.

Логічний вираз може набувати одного з двох значень—істина чи хиба.

Команда **повне розгалуження** — це вказівка виконати одну з двох команд:*команду1* або *команду2* залежно від істинності (true) чи хибності (false) деякої умови. Якщо *умова* істинна, то виконується *команда1* і на цьому виконання розгалуження закінчується. Якщо ж *умова* хибна, то виконується *команда2* і на цьому виконання команди розгалуження закінчується.

Окремим випадком розгалуження є **неповне розгалуження**, коли в разі хибностіумови ніякі операції в розгалуженні взагалі не виконуються. Повне розгалуження завжди можна подати у вигляді слідування двох неповних. *Умова* може бути складною - утворена за допомогою логічних операцій «не», «та», «або», а замість *команди1* чи *команди2* може бути декілька команд, які називаються серією команд.



3. Базова структура повторення - забезпечує багаторазове виконання деякої сукупності дій, яка називається **тілом циклу**.

Циклом називається така форма організації дій, при якій одна й та ж послідовість дій виконується декілька разів (або ні разу), доки виконується деяка умова. Кожне виконання тіла циклу супроводжується перевіркою умови завершення циклу і називається його **ітерацією**. Якщо умова завершення хибна, то тіло циклу виконується ще раз, якщо істинна, то виконання циклу припиняється і здійснюється перехід до виконання наступної за циклом команди. Замість умови завершення циклу може перевірятися її заперечення — **умова продовження**, хибність якої призводить до припинення виконання циклу, а істинність - до продовження. Змінні, значення яких модифікуються в тілі циклу і впливають на істинність умови завершення, називаються **параметрами циклу**. Виконанню будь-якого циклу має передбачати ініціалізацію його параметрів.

Розрізняють :

1. цикли з заданим числом повторень (цикли з параметром);
2. цикли з невідомим числом повторень (цикли з «передумовою» та «післяумовою»)

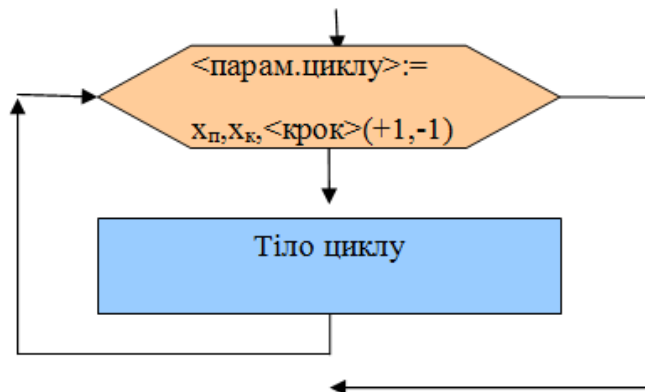
Цикл з параметром — цикл, в якому параметр циклу змінює своє значення від заданого початкового значення до кінцевого значення з деяким кроком, і для кожного значення цієї змінної тіло циклу виконується один раз.

Цикл з передумовою- коли ми спочатку перевіряємо умову, а потім виконуємо деяку послідовність дій;

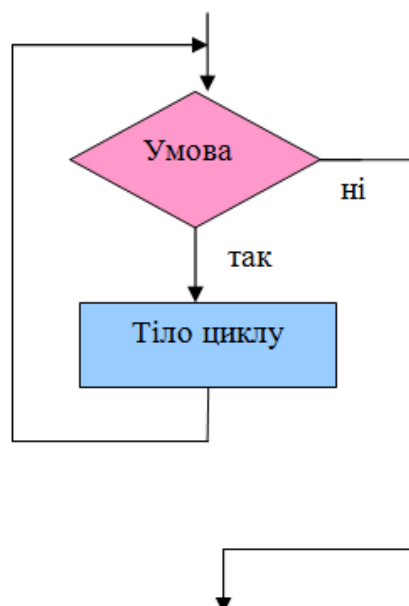
Цикл с післяумовою- спочатку ми виконуємо хоч один раз необхідну послідовність дій, а потім перевіряємо, чи не досягли ми бажаного результату.

Мовою блок-схем типи циклів виглядають наступним чином:

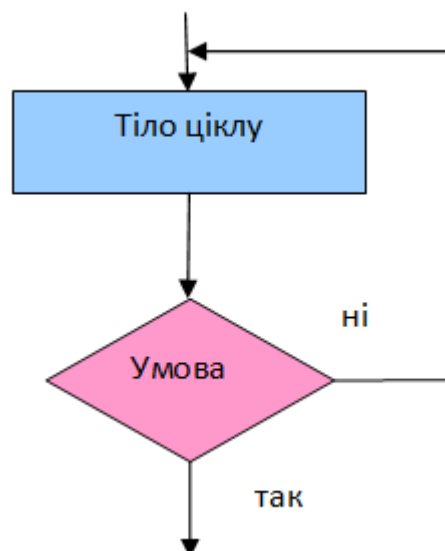
Цикл з параметром



Цикл з передумовою



Цикл з післяумовою



У **циклі з передумовою** спочатку варто визначити, істинна чи хибна умова. Якщо вона істинна, то виконати **команди**, які утворюють тіло циклу, і знову повертаються до визначення істинності умови. Якщо ж умова хибна, то виконання **тіло циклу** завершується.

У **циклі з післяумовою** спочатку виконується **команди**, які утворюють тіло циклу, а потім визначається істинність умови виходу із циклу. Якщо умова хибна, то знову виконуються тіло циклу, і знову визначається істинність умови. Якщо ж умова істинна, то виконання вказівки завершується.

У структурі **цикл з передумовою** тіло циклу може не виконуватися жодного разу. У структурі ж **цикл з післяумовою** тіло циклу буде виконуватися принаймні один раз.

Технологічний підхід до побудови алгоритму

Будь-яке масове виробництво завжди спирається на технологію. Технологія дозволяє розподілити складний виробничий процес на окремі етапи й організувати роботу різних груп виконавців над кінцевим продуктом. Так створюються літаки й автомобілі, будуються споруди, виготовляються продукти харчування тощо. З поширенням сфери застосування комп'ютера виникла індустрія програмних засобів, а з нею й технологія виробництва програмних продуктів.

Основою будь-якого програмного продукту є алгоритм. Познайомимося з **технологією розробки алгоритму**, яка ґрунтується на **основних методах сучасної технології проектування алгоритмів** :

1. Метод **структурного проектування**. Будь алгоритм може бути побудований з комбінацій трьох базових структур: лінійний алгоритм(слідування), що розгалужується алгоритм(розвилка) і циклічний алгоритм (повторення).

2. Метод **спадного проектування**. Спочатку виділяються головні функції, потім другорядні.

3. Метод **покрокової деталізації(методом «зверху донизу»)**;

4. Метод **модульності**. Організація алгоритму у вигляді відносно незалежних частин — модулів Модуль - логічно зв'язаний фрагмент алгоритму, що виконує одну функцію і складається з скінченного числа кроків.

Проектування алгоритму за методом покрокової деталізації розпочинається з визначення основних складових поставленої задачі. Назвемо їх підзадачами. Далі здійснюється аналіз кожної з виділених підзадач і визначаються їх складові і т. д. Урешті-решт задача буде подана як сукупність простих елементів, і в подальшій деталізації не буде потреби. Головна перевага описаного методу полягає в тому, що в процесі проектування закладається правильність алгоритму: адже на кожному кроці фактично вирішується, чи можна цю задачу подати як сукупність більш простих задач. Поступове і цілком прозоре зменшення складності задачі зводить її до такого стану, коли алгоритм можна написати без помилок. Спроектований у такий спосіб алгоритм складається з окремих модулів, які відповідають виділеним підзадачам.

Модуль — це логічна частина алгоритму, яка є відносно незалежною, має певне цільове призначення й вирішує тільки одну чітко сформульовану задачу.

Іноді в процесі проектування алгоритму з'ясовується, що одну й ту саму підзадачу необхідно вирішувати на різних етапах розв'язання задачі. Модуль для такої підзадачі створюється один раз, а використовується кожного разу, коли це потрібно. Незалежність модулів дозволяє розподілити роботу зі складання алгоритму між різними

групами виконавців, а в подальшому використовувати розроблені модулі як готові цеглинки для конструювання нових алгоритмів.

Технологічно доцільно при написанні алгоритму спиратися на застосування уніфікованих алгоритмічних структур — слідування, розгалуження, повторення. Це дозволяє використовувати стандартні прийоми програмування, що сприяє зменшенню помилок у програмах.

Кожний алгоритм можна подати у вигляді комбінації трьох базових алгоритмічних структур, що є їх основною властивістю. Використання цього принципу покладено в основі структурного підходу до побудови алгоритмів. Розроблені алгоритми мають чітко виражену структуру.

Використання алгоритмічних базових структур дозволяє:

1. Зробити алгоритми читабельними та зрозумілими.
2. Доводити правильність алгоритмів (тобто довести, що кожен алгоритм має одну точку входу і одну точку виходу, не містить нескінченних циклів, а також показати, що результати, які отримаємо за допомогою алгоритму відповідають шуканим).
3. Оцінити ефективність алгоритмів (тобто дати оцінку алгоритму з точки зору обчислювальних затрат).

Лінійний алгоритм-алгоритм, в якому використовується тільки структура слідування.

Алгоритм з розгалуженням- алгоритм, в основі якого лежить структура розгалуження.

Циклічний алгоритм-алгоритм, в основі якого лежить структура повторення.

Питання для самоперевірки

1. Поясніть призначення основних алгоритмічних структур.
2. Який алгоритм називається лінійним?
3. Які є різновиди алгоритмів із розгалуженою структурою?
4. Наведіть приклади циклічних процесів.
5. Опишіть відмінності між циклами з передумовою, післяумовою, з параметром.
6. В яких циклах тіло циклу може не виконуватися жодного разу? Поясніть відповідь.
7. Порівняйте виконання основних алгоритмічних структур.
8. Поясніть сутність методу покрокової деталізації під час складання алгоритму.

Етапи розв'язування задачі за допомогою комп'ютера. Вхідні дані та результати, їх взаємозв'язок. Поняття про метод розв'язування задачі. Поняття програми.

1. Метод розв'язання задачі.
2. Основні етапи розв'язання прикладних задач із використанням комп'ютера.
3. Визначення понять «мова», «програма», «мова програмування», «алгоритмічна мова».
4. Класифікація мов програмування.

Метод - це загальний спосіб, сукупність прийомів, що використовуються для розв'язання задач певного типу.

У багатьох задачах знаходження точного розв'язку є неможливим або надто складним. До таких задач розроблені наближені методи (їх ще називають чисельними методами).

Існують й інші методи розв'язування задач: впорядкування, пошук даних, оптимізація.

Вибір методу зумовлюється аналізом вхідних даних конкретної задачі й умовами застосування кожного з наявних методів.

Вибір методу розв'язання задачі за допомогою комп'ютера має спиратись на отримання швидкого результату з мінімальними затратами оперативної пам'яті.

Нині важко уявити собі життя сучасної людини без комп'ютера. Люди використовують його для розв'язання найрізноманітніших задач: від виконання складних обчислень у наукових дослідженнях та економіці до виконання кропіткої домашньої роботи (наприклад: такі процеси як прання білизни, приготування їжі, миття посуду з використанням сучасної побутової техніки зараз комп'ютеризовані. Комп'ютеру можна доручити навіть функції домашнього секретаря). Комп'ютер — це помічник людини, без нього неможлива обробка величезного потоку інформації, який кожного дня все зростає: будь то оформлення складної документації, створення та обробка графічних зображень, розв'язування математичних задач, отримання даних з будь-якої теми, тощо. Це далеко не повний перелік всіх можливостей нашого помічника.

Для розв'язання цих задач комп'ютер озброєний найрізноманітнішим програмним забезпеченням, яке поділяється на чотири великих категорії: операційні системи, системні утиліти, системи програмування, прикладне програмне забезпечення. Отже, користувач аналізує завдання, яке необхідно розв'язати, та обирає оптимально придатний програмний засіб, який є в арсеналі засобів його комп'ютера. Однак є велика кількість задач, для розв'язання яких не існує відповідного програмного забезпечення, або існуюче програмне забезпечення з певних причин не влаштовує. У цьому випадку користувач може самостійно написати програму для виконання поставленої задачі.

Постановка (або формулювання) — це перший крок у розв'язанні будь-якої задачі. На цьому етапі слід чітко з'ясувати: що дано і що треба знайти. Тобто чітко уявити суть задачі, необхідні початкові дані для її розв'язання, а також те, що можна вважати за очікуваний результат.

Побудова інформаційної математичної моделі — це другий крок розв'язування задачі. Це дуже відповідальний етап, оскільки не завжди в умові задачі міститься формула, яку можна застосувати в програмі. Для цього створюється інформаційна математична модель об'єкта.

Розробка алгоритму — третій крок у розв'язуванні задач. Алгоритм розробляється на основі побудованої математичної моделі. В ньому можна використати вже відомі методи розв'язування отриманих математичних співвідношень, причому за наявності кількох методів розв'язування, необхідно їх проаналізувати та обрати оптимальний. Якщо існуючими методами не можна розв'язати задачу, то треба розробити власний метод.

Під час створення складних алгоритмів застосовується **метод покрокової деталізації**, який полягає в тому, що складна задача розбивається на прості підзадачі, кожна з яких, в свою чергу, може розбиватися на ще простіші. Такий підхід дозволяє розбити алгоритм на окремі частини — модулі, реалізацію кожного з таких модулів можна доручити окремому програмісту. В цьому випадку програміст концентрується на розв'язанні окремої підзадачі, використовуючи для цього свої методи.

Останнім етапом у методі покрокової розробки є об'єднання окремих модулів у єдине ціле. Для цього між усіма модулями мають бути встановлені зв'язки, тобто узгоджена передача інформації від одних модулів до інших. Це дуже кропітка робота і від оптимальності вибору вхідних та вихідних параметрів окремих модулів залежить оптимальність роботи всієї програми.

Алгоритм, призначений для комп'ютерної реалізації, має бути записаний однією з **мов програмування**. На даному етапі розвитку комп'ютерної техніки існує така різноманітність мов програмування, що програміст завжди може обрати оптимальний варіант для отримання результату розв'язку. А враховуючи можливість розбиття алгоритму на окремі модулі, реалізацію кожної підзадачі можна виконати різними засобами.

Отже, мова програмування обрана, програма написана. Тепер програму необхідно **налагодити та протестувати**. Це наступний крок розв'язування задачі. Під налагоджуванням програми розуміють процес випробування роботи програми з виправленням виявлених при цьому помилок. Виправити **синтаксичні помилки**, пов'язані з порушенням правил написання програм, допоможе середовище програмування, але **семантичні (логічні) помилки**, пов'язані з неправильним змістом дій та використанням недопустимих значень величин, виправити набагато важче. В цьому допоможуть правильно підібрані тести.

Останній крок — це **використання програми для отримання результатів**. На цьому етапі необхідно ще раз перевірити правильність очікуваних результатів. Якщо отримані результати є помилковими, слід повернутися до одного з попередніх етапів (іноді, навіть, до самого першого — постановки задачі) і ще раз перевірити правильність проведених робіт. Можливо, що деякі етапи потребують переробки або доопрацювання.

Тепер програму можна експлуатувати і, навіть, пропонувати іншим користувачам, доповнивши її необхідною документацією.

Отже, розв'язання задач із використанням комп'ютера характеризується декількома етапами, деякі з яких виконуються безпосередньо людиною, інші — людиною і машиною. Кожен із цих етапів важливий по-своєму, і тільки правильне виконання кожного з етапів приводить до розв'язання задачі в цілому.

Етапи розв'язання задач за допомогою комп'ютера

Етап	Опис етапу
Постанова задачі	Формулювання задачі в термінах деякої галузі знань, опис початкових даних та очікуваних результатів, формулювання мети задачі
Побудова інформаційної моделі	Опис реального об'єкта дослідження в припустимих для реалізації задачі термінах, щоб звести дослідження реального об'єкта до розв'язання задачі на моделі
Вибір методу формалізованої задачі	Метод розв'язуваної задачі — це сукупність теоретичних принципів і практичних

		приймів, що використовуються під час розв'язування задачі
Розробка алгоритму розв'язуваної задачі		Алгоритм більшою мірою визначається обраним методом, хоча той самий метод може бути реалізований за допомогою різних алгоритмів. Під час складання алгоритму необхідно враховувати всі його властивості
Вибір програмного забезпечення		Визначення необхідного прикладного програмного забезпечення (якщо воно є) або розробка нового програмного забезпечення (розробка алгоритму, вибір системи програмування, написання й тестування програми)
Виконання алгоритму на комп'ютері та отримання комп'ютерної моделі	Настроювання алгоритму або програми	Перевірка правильності роботи програми за допомогою тестів і виправлення виявлених помилок. Тест — це спеціально дібрані вхідні дані для отримання певного результату
	Аналіз отриманих результатів	Аналіз результатів, отриманих на моделях і на реальних об'єктах, для виправлення помилок і доопрацювання розробленої прикладної програми, що пройшла тести на моделі

Мова — це система знаків (символів, жестів, міміки, положень перемикача тощо) для подання інформації та обміну нею. Це загальне визначення охоплює і природні, і штучні (формальні) мови.

Алгоритмічна мова — це мова, призначена для подання алгоритму у вигляді послідовності вказівок для виконання їх виконавцем. Алгоритмічна мова, як і кожна мова, має свій словник. Його основу складають слова, що використовуються для запису команд, які входять до системи команд виконавця.

Мови програмування — це алгоритмічні мови, призначені для опису алгоритмів, що орієнтовані для виконання на комп'ютері, або система позначень для точного опису алгоритму, який необхідно виконати за допомогою комп'ютера.

Програма — це алгоритм, записаний мовою програмування.

Класифікація мов програмування

За ступенем залежності від апаратних засобів	За принципом програмування	За орієнтацією на клас задач
---	-----------------------------------	-------------------------------------

<p>Низького рівня (мова асемблера — машинно-залежна мова, що описує дії в термінах команд процесора та програми, які транслюють арифметичні вирази (автокоди))</p>	<p>Процедурні мови. Програма, написана процедурною мовою, описує, що необхідно робити для розв'язання задачі, використовуючи при цьому такі основні поняття, як слідування, розгалуження, цикл</p>	<p>Універсальні мови призначені для розв'язування різноманітних задач</p>
<p>Високого рівня (машинно-незалежні мови, наближені до природних)</p>	<p>Непроцедурні мови. Програма, написана непроцедурною мовою, описує, що потрібно робити для розв'язання задачі, використовуючи такі поняття, як підстановка, розгалуження, рекурсія</p>	<p>Спеціалізовані мови враховують специфіку предметної області</p>
	<p>Об'єктно-орієнтовані оперують такими поняттями, як об'єкти та класи об'єктів</p>	

Приклад

Формулювання задачі: Дано три числа. Потрібно перевірити, чи існує трикутник, довжини сторін якого збігаються з цими числами. У разі позитивної відповіді, обчислити площу трикутника .

1) Вказуємо вихідні дані і необхідні результати.

Які у нас вихідні дані?

Вихідні дані – числа a, b, c . Вони можуть набувати будь-яких значень.

Що ми повинні отримати у кінці?

Необхідні результати:

- відповідь на запитання : «Чи є твердження «Існує трикутник, довжини сторін якого дорівнюють a, b, c » істинним чи хибним?»;

-у випадку, якщо такий трикутник існує - обчислити його площу S .

Обов'язково врахувати, що площа трикутника повинна бути невід'ємною величиною.

2) Для побудови алгоритму потрібно визначити, за яких умов можливе отримання необхідних результатів, а за яких – ні. Вибір умов для відповіді на питання про існування трикутника із даними довжинами сторін.

Яка це умова?

Одночасне виконання трьох нерівностей $a < b + c, b < a + c, c < a + b$.

Формули, за якими обчислюється площа трикутника.

Яка підійде формула?

Формула Герона $S = \sqrt{p(p-a)(p-b)(p-c)}$, де $p = \frac{a+b+c}{2}$.

3) Вибираємо мову програмування або конкретне програмне забезпечення (оскільки цю задачу можна зробити як в, наприклад, Pascal так і в Excel)

4) Комп'ютер перероблює текст програми, написаний обраною мовою чи програмним забезпеченням на машинну мову. І якщо щось буде незрозуміло, комп'ютер видає помилки, які ми маємо виправити.

5) Тестуємо програму. Вводимо підібрані числа замість a, b, c . Числа підбираємо таким чином, щоб у нас утворювався трикутник, а при іншому наборі – ні. Дивимось, чи правильно програма працює, тобто - чи правильно дає відповідь на питання? Якщо трикутник утворюється, то перевіряємо: чи правильно рахує площу (число, обраховане вами, повинно співпадати з числом, яке видає комп'ютер).

Питання для самоперевірки

1. Що таке метод розв'язання задачі?
2. Назвіть етапи рішення задач при допомозі комп'ютера.
3. Які помилки називають синтаксичними/логічними?
4. Що таке програма?

ДОДАТОК № 1 (завдання)

2. Структура слідування

- 1) Скласти алгоритм переправи вовка, кози і капусти через річку, якщо човен вміщує лише двох і не можна залишати наодинці вовка з козою, а козу з капустою.
- 2) Скласти алгоритм переправи трьох ревних джентльменів і їх жінок через річку, якщо човен вміщує лише двох, а жодний джентльмен не залишить свою дружину наодинці з іншими джентльменами.
- 3) Дехто має напувати двох коней. Він має 8-літрове відро, наповнене водою, і два порожніх відерця об'ємом у 3 і 5 літрів. Скласти алгоритм, за яким можна розділити воду нарівно у два відра.
- 4) Обчислити відстань між двома точками з координатами $(X_1; Y_1)$ і $(X_2; Y_2)$ за формулою $L = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}$.
- 5) Знайти площу кільця між колами з радіусами R_1 і R_2 , $R_1 < R_2$ за формулою $S = 3.14 * (R_2^2 - R_1^2)$

3. Структура розгалуження

- 1) Складіть алгоритми розв'язання задач про монети: серед 4 монет є одна фальшива, яка відрізняється від справжніх за вагою. За два виваження на терезах без важелів знайти фальшиву монету.
- 2) Скласти алгоритм обчислення значення функції:
$$Y = \begin{cases} \sqrt[4]{3x^2 + 1}, & \text{якщо } x \leq 0; \\ 2x, & \text{якщо } 0 < x < 5; \\ (x - 1)^2, & \text{якщо } x \geq 5. \end{cases}$$
- 3) Скласти алгоритм знаходження розв'язку лінійного рівняння $ax + b = 0$.
- 4) Скласти алгоритм обчислення значення функції $\text{sign}(x)$ «знак числа».
- 5) Скласти алгоритм визначення координатної чверті, якій належить точка M з координатами x і y .

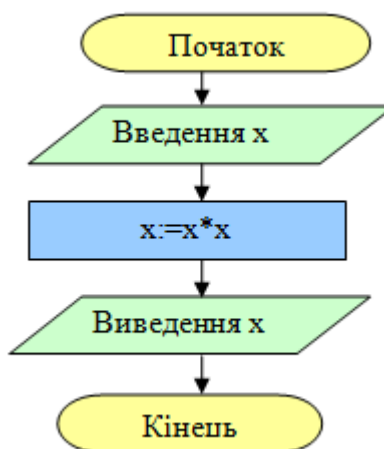
4. Структура повторення

- 1) За відомою легендою, винахідник шахів запросив у царя за цю гру таку винагороду: «Поклади на першу клітинку шахів одне зернятко, на другу — два і на кожну наступну клітинку — вдвоє більше, ніж на попередню». Складіть блок-схему структурного алгоритму підрахування кількості зерен, які необхідно покласти на шахову комірку з номером n ;
- 2) Скласти алгоритм знаходження найбільшого спільного дільника двох натуральних чисел M і N .
- 3) Скласти алгоритм обрахунку значення:
$$e^x = x + \frac{x}{1!} + \frac{x}{2!} + \frac{x}{3!} + \dots + \frac{x}{n!} + \dots$$
з точністю 0.01. поки $\frac{x}{n!} \geq 0.01$
- 4) Скласти алгоритм знаходження суми членів ряду:
$$S = \frac{1}{a} + \frac{1}{a^2} + \dots + \frac{1}{a^n}$$
- 5) Скласти алгоритм знаходження суми цілих додатніх чисел менших n .

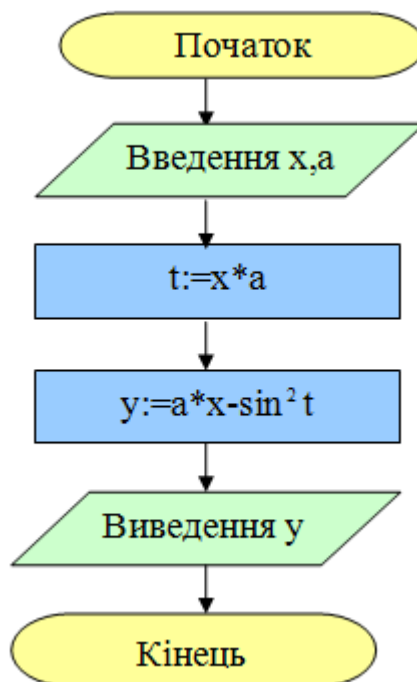
ДОДАТОК №2 (Класифікація завдань)

2. Лінійний алгоритм

1) Алгоритм обчислення квадрата числа.

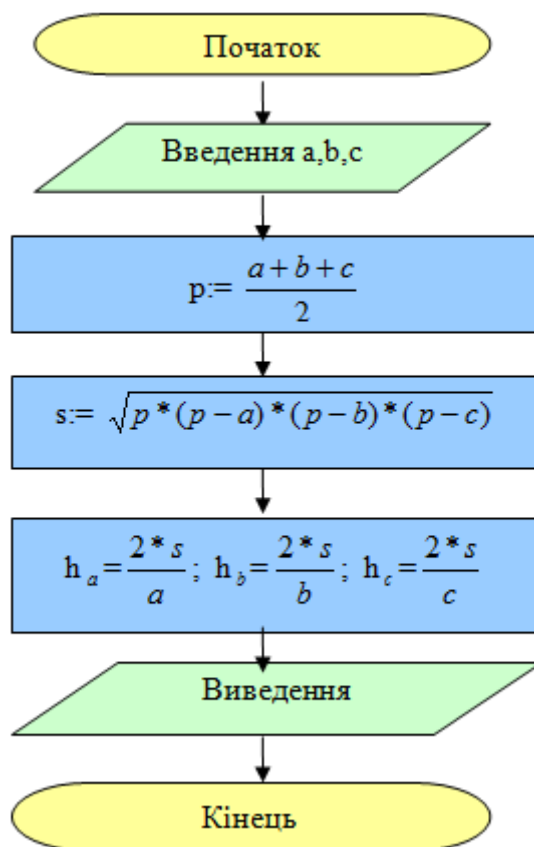


2) Алгоритм обчислення значення функції $y = ax - \sin^2 ax$



3) Алгоритм обчислення висот трикутника з сторонами a, b, c по формулам :

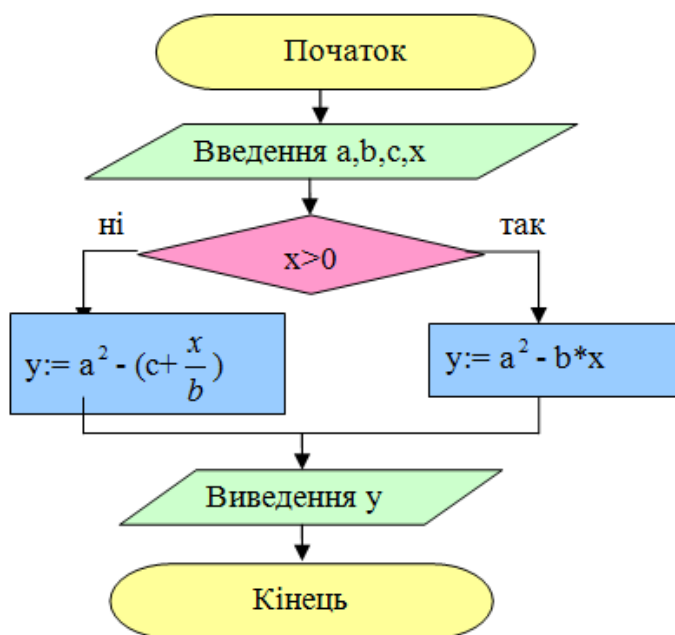
$$h_a = \frac{2 * s}{a}; h_b = \frac{2 * s}{b}; h_c = \frac{2 * s}{c}; \text{ де } s = \sqrt{p * (p - a) * (p - b) * (p - c)}; p = \frac{a + b + c}{2}$$



3. Алгоритм з розгалуженням

1) Алгоритм обчислення значення виразу:

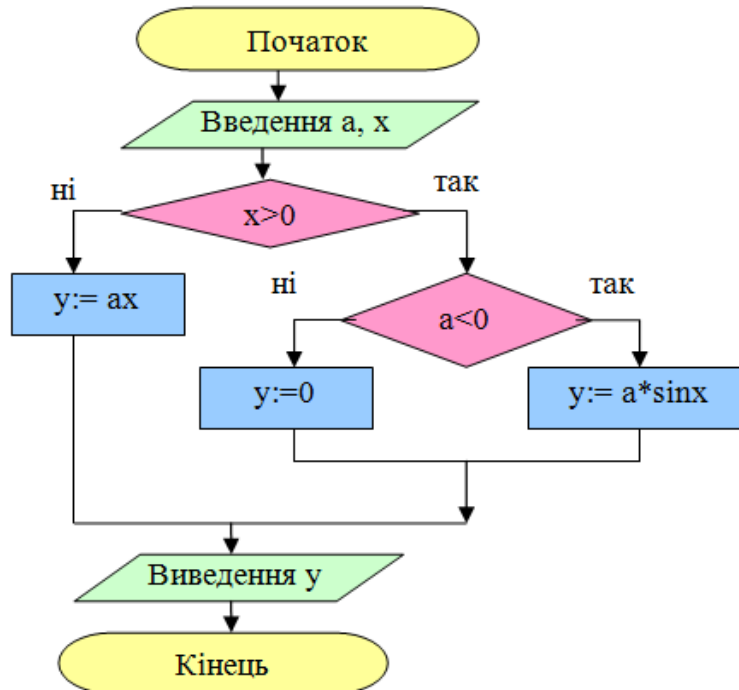
$$y = \begin{cases} a^2 - b*x, & \text{якщо } x > 0 \\ a^2 - (c + \frac{x}{b}), & \text{якщо } x \leq 0 \end{cases}$$



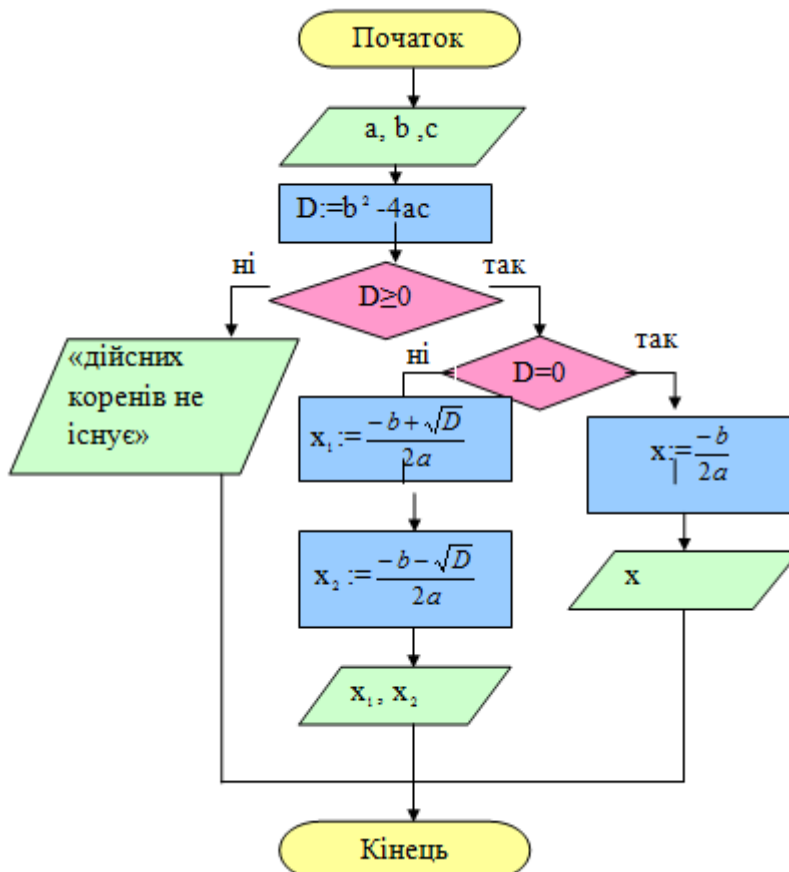
2) Алгоритм обчислення значення виразу:

$$y = \begin{cases} ax, & \text{якщо } x > 0 \text{ і } a \leq 0 \\ a \cdot \sin x, & \text{якщо } x > 0 \text{ і } a < 0 \\ 0, & \text{якщо } x \leq 0 \end{cases}$$

Цей алгоритм є прикладом алгоритму з перевіркою складної умови: якщо $x > 0$, здійснюється перевірка умови $a < 0$. Якщо умова $x < 0$ не виконується, змінній y присвоюється значення 0.



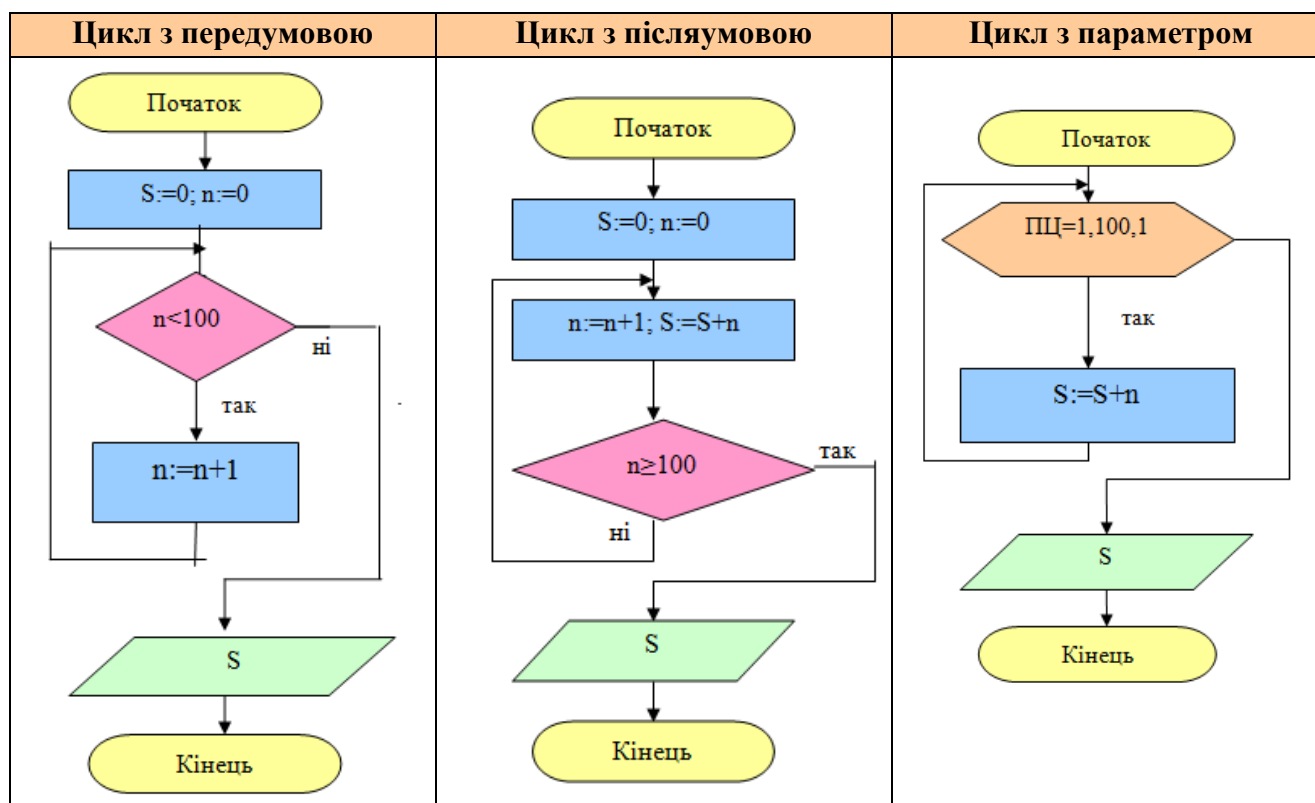
3) Розв'язати квадратне рівняння $ax^2 + bx + c = 0$



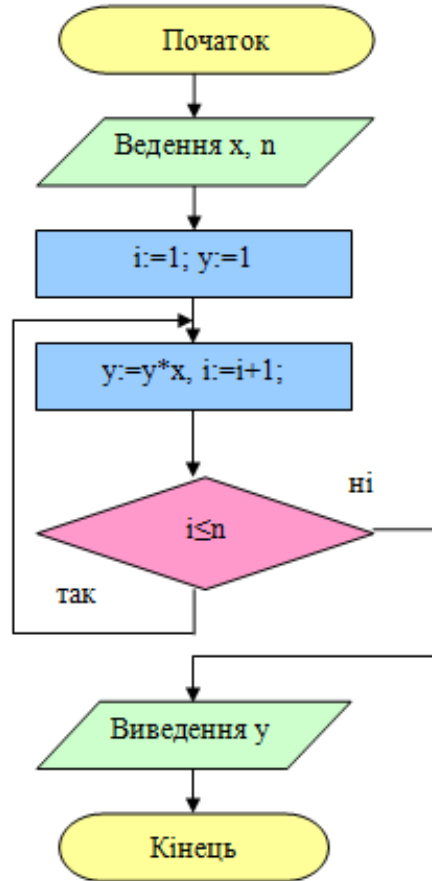
4. Циклічний алгоритм

1) Знайти суму перших ста натуральних чисел.

Словесний спосіб	Формульно-словесний		
	Цикл з передумовою	Цикл з післяумовою	Цикл з параметром
<p>1.Надати сумі S по-чаткове значення 0</p> <p>2.Надати числу n початкове значення 0</p> <p>3.У циклі повторювати: -збільшувати поточне значення числа n на 1; -збільшити поточне значення суми S на n</p> <p>4.Цикл закінчити, коли n набуде значення 100 і воно буде додано до суми S.</p> <p>5.Вивести обчислене значення S.</p>	<p>1.$S:=0$</p> <p>2.$n:=0$</p> <p>3.Доки $n < 100$ Викону вати :</p> <p>$n:=n+1$; $S:=S+n$</p> <p>4.Вивес ти S.</p>	<p>1.$S:=0$</p> <p>2.$n:=0$</p> <p>Викону вати</p> <p>$n:=n+1$; $S:=S+n$</p> <p>До $n \geq 100$</p> <p>4.Вивес ти S.</p>	<p>1.$S:=0$</p> <p>2.Для n Від 1 До 100 Кроком 1</p> <p>Виконувати $S:=S+n$</p> <p>3. Вивести S.</p>



1) Алгоритм обчислення значення функції $y=x^n$, де n натуральне число і $n=4$.



1-й цикл	2-й цикл	3-й цикл	4-й цикл
$x=3, n=4$ $i=1, y=1$ $y=3$ $i=2$ $2 \leq 4$ так	$y=9$ $i=3$ $3 \leq 4$ так	$y=27$ $i=4$ $4 \leq 4$ так	$y=81$ $i=5$ $5 \leq 4$ Ні Виведення 81 Кінець

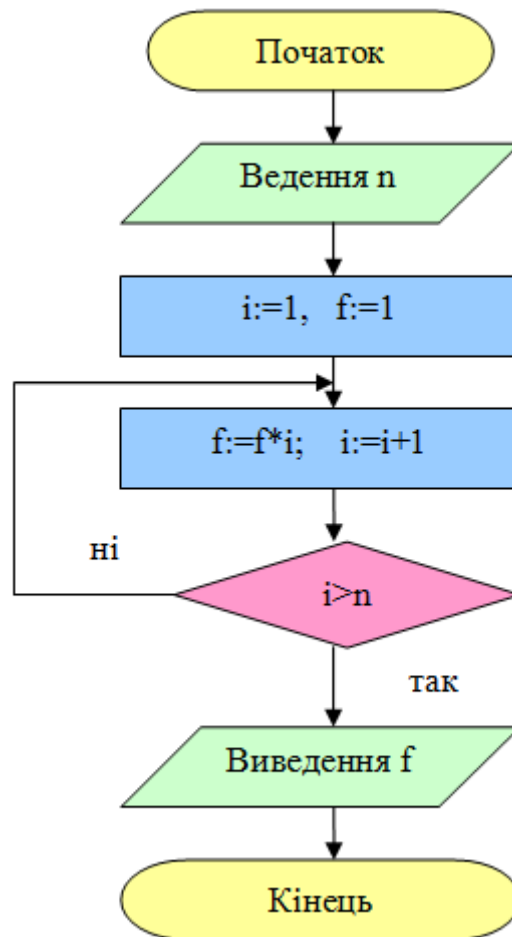
3) Знайти $n!$

$$1! = 1$$

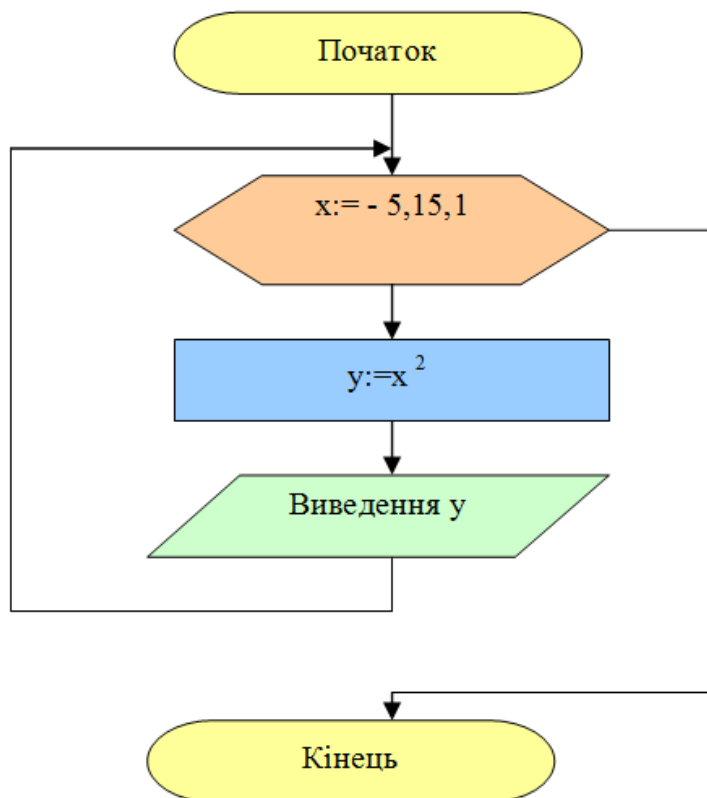
$$2! = 1 * 2 = 2$$

$$3! = 1 * 2 * 3 = 6$$

$$4! = 1 * 2 * 3 * 4 = 24$$



2) Алгоритм знаходження значення функції $y=x^2$ для x від -5 до 5 з кроком 1



3. КОНТРОЛЬНО-ОЦІНЮВАЛЬНИЙ БЛОК

Алгоритм, його властивості й форми подання

- 1) Вставте слова, пропущені у наведеному нижче визначенні алгоритму.
«Алгоритм — це _____ і _____ припис про здійснення скінченної послідовності визначених дій з метою розв'язання задачі певного типу або досягнення поставленої мети».
- а) чіткий;
 - б) точний;
 - в) зрозумілий для виконавця;
 - г) короткий;
 - д) простий;
 - е) доступний.
- 2) Чітка послідовність дій, яка приведе до бажаного результату називається:
- а) виконавцем;
 - б) алгоритмом;
 - в) системою команд виконавця;
 - г) командами;
- 3) Який з об'єктів може бути виконавцем:
- а) тварина;
 - б) карта;
 - в) комп'ютер;
 - г) людина;
 - д) книга;
- 4) Який з документів є алгоритмом ?
- а) рецепт приготування страви;
 - б) розклад уроків;
 - в) порядок дій при побудови бісектриси кута з використанням циркуля та лінійки;
 - г) правила техніки безпеки;
- 4) Властивостями алгоритму є:
- а) зрозумілість;
 - б) повнота;
 - в) лаконічність;
 - г) масовість;
 - д) дискретність;
 - е) детермінованість;
 - є) формальність;
 - ж) скінченність;
 - з) правильність.
- 5) Зрозумілість алгоритму означає, що:
- а) кожна його команда є зрозумілою для всіх;
 - б) кожна команда алгоритму є зрозумілою для його виконавця;
 - в) кожна його команда є простою для виконання;
 - г) виконавець розуміє й може виконати кожен команду алгоритму;
 - д) кожна команда алгоритму може бути виконаною.
- 6) Детермінованість алгоритму означає, що:
- а) алгоритм дає виконавцю все, що йому потрібно для розв'язання завдання;
 - б) алгоритм обмежує дії виконавця;

- в) алгоритм однозначно визначає всі дії виконавця;
 г) алгоритм гарантує розв'язання задачі за скінченну кількість кроків.
- 7) Дискретність алгоритму означає, що він подає процес розв'язання задачі у вигляді:
- а) сукупності окремих простих дій;
 б) визначеної послідовності окремих простих дій;
 в) певної множини зрозумілих дій;
 г) скінченної множини дій, які виконавець може виконати.
- 8) Масовість алгоритму означає, що його можна застосовувати:
- а) для розв'язання будь-яких задач;
 б) для розв'язання будь-яких задач певного класу;
 в) для розв'язання задач певного класу з будь-якими вхідними даними;
 г) для розв'язання задач певного класу, які мають вхідні дані, що належать до кола допустимих для даного алгоритму.
- 9) Блок-схема це-
- а) графічне зображення алгоритму у вигляді спеціальних блоків з необхідними словесними поясненнями;
 б) словесне зображення алгоритму у вигляді спеціальних блоків з необхідними поясненнями;
 в) це зображення кодів алгоритму у вигляді спеціальних блоків з необхідними словесними поясненнями;
 г) це неповне зображення алгоритму у вигляді спеціальних блоків з необхідними словесними поясненнями;

10) Який вираз обчислюється за допомогою дій: $a=x^2$; $b=\frac{a}{x}$; $c=b+5$; $y=c^2$

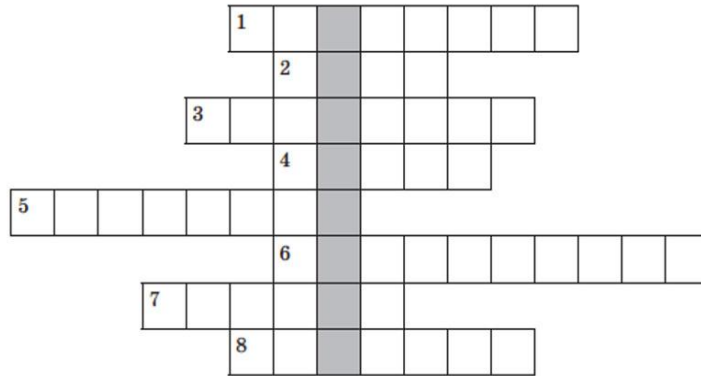
а) $y=(x+5)^2$; б) $y=\frac{(x+5)^2}{x}$; в) $y=\left(\frac{x+5}{x}\right)^2$; г) $y=x^2+5$

11) Проаналізуйте алгоритм обчислення виразу $f=2(x+5)-3x$, якщо $x=5$, за діями. Визначте вхідні, проміжні, вихідні величини та їхні поточні значення кожному кроці, заповніть таблицю.

Алгоритм	Поточні значення	Вхідні величини:
$x=5$	$x=5$	Проміжні величини
$a=x+5$		
$b=2a$		Вихідні величини
$c=3x$		
$f=b-c$		

12) Розв'язати кросворд

- Команда іншим словом. (Вказівка)
- Чітка послідовність дій. (План)
- Алгоритм, складений для комп'ютера. (Програма)
- Зміна властивостей об'єкта, взаємодія між об'єктами, утворення нового об'єкта або знищення існуючого об'єкта. (Подія)
- Електронний пристрій для обробки інформації. (Комп'ютер)
- Того, хто виконує алгоритм, називають... (Виконавець)
- Предмети, явища і процеси, які розглядаються як єдине ціле. (Об'єкти)
- Точна, зрозуміла вказівка для виконання якоїсь дії. (Команда)



Базові структури алгоритмів

1) Алгоритм, який приписує виконання певної послідовності дій при будь-яких допустимих вхідних даних, називається:

- а) лінійним;
- б) алгоритмом із розгалуженням;
- в) алгоритмом із повторенням.

2) Алгоритм, який приписує виконання тих чи інших дій залежно від результату перевірки умови, називається:

- а) лінійним;
- б) алгоритмом із розгалуженням;
- в) алгоритмом із повторенням.

3) Алгоритм, який приписує повторне виконання дій і за яким кількість повторень визначається заданою умовою, називається:

- а) лінійним;
- б) алгоритмом із розгалуженням;
- в) алгоритмом із повторенням.

4) Як можна прочитати базову структуру розгалуження? Вставте пропущені слова. «... перевірка умови дає результат ..., ... виконати дію 1, ... виконати дію 2».

- а) доки;
- б) якщо;
- в) повторювати;
- г) то;
- д) інакше;
- е) до;
- є) «так»;
- ж) «ні».

5) Як можна прочитати базову структуру циклу з передумовою? Вставте пропущені слова.

«... перевірка умови дає результат ..., виконувати дію».

- а) доки;
- б) якщо;
- в) повторювати;
- г) то;
- д) інакше;
- е) до;
- є) «так»;
- ж) «ні».

б) Як можна прочитати базову структуру циклу з післяумовою? Вставте пропущені слова.

«... дію ... одержання результату ... при перевірці умови».

- а) доки;
- б) якщо;
- в) повторювати;**
- г) то;
- д) інакше;
- е) до;
- є) «так»;
- ж) «ні».

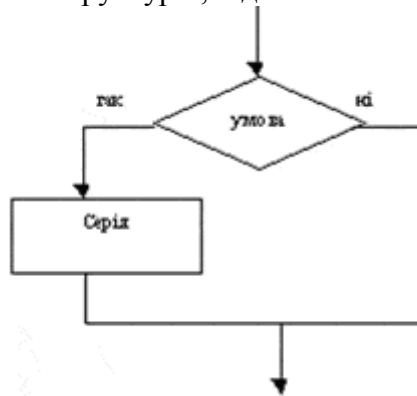
7) Що означає проектування алгоритму методом «зверху донизу»?

- а) поступове роздроблення задачі на все більш прості складові;**
- б) запис алгоритму, починаючи з верхнього рядка і до самого нижнього;
- в) управління процесом проектування алгоритму з верхньої до нижньої ланки.**

8) Модульна організація алгоритму — це:

- а) організація алгоритму з крупних блоків;
- б) організація управління алгоритмом за допомогою модулів;
- в) складання алгоритму з окремих відносно незалежних частин.**

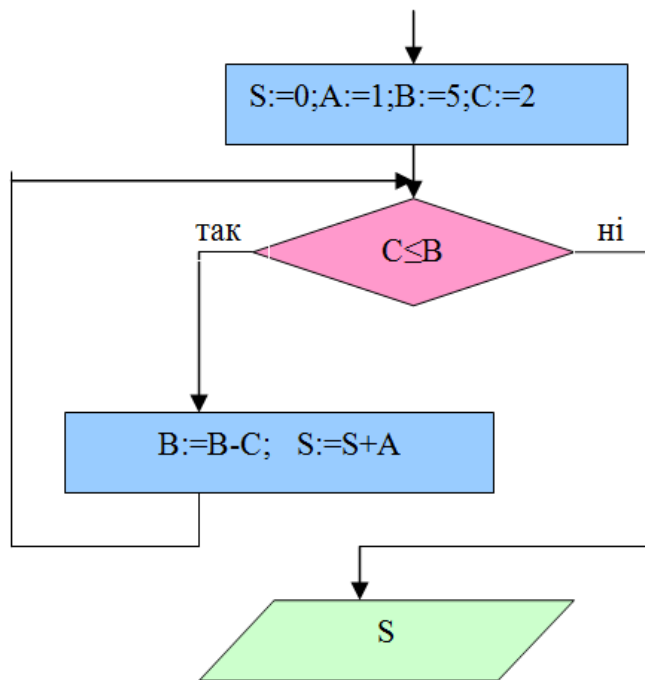
9) Визначте тип алгоритмічної структури, поданої схематично:



- а) слідування;
- б) повторення;
- в) неповна форма розгалуження;**
- г) повна форма розгалуження

10) Після виконання фрагмента алгоритму, поданого схематично, для заданих значень змінних буде виведено число:

- а) 0; б) 1; в) 2; г) 3;

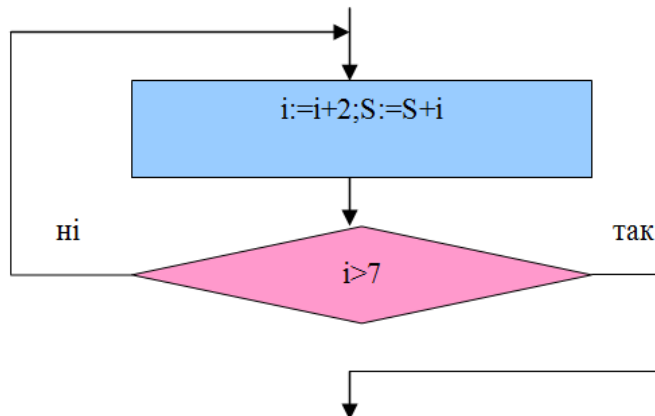


11) Скільки разів буде перевірено умову повторення після виконання фрагменту алгоритму, поданого в пункті 10?

а) жодного; б) один; в) двічі; г) тричі;

12) Скільки разів виконується тіло циклу для заданих значень змінних $S=0, i=1$ після виконання фрагмента алгоритму, поданого схематично?

а) один; б) двічі; в) тричі; г) жодного



Задача. Підрахуйте вартість обгортки подарунка у вигляді куба з ребром 50 см, якщо 1 м² кольорового паперу коштує 5 грн.

4. ЛІТЕРАТУРА

1. Білоусова Л.І. **Інформатика в таблицях і схемах** / Л.І. Білоусова, Н.В. Олєфіренко. – Харків: ТОРСІНГ ПЛЮС, 2014. – 112 с.
2. **Інформатика: 11 клас: підручник для загальноосвітніх навчальних закладів: рівень стандарту** / Й.Я. Ривкінд, Т.І. Лисенко, Л.А. Чернікова, В.В. Шакотько; за загальною редакцією М.З. Згуровського. – К.: Генеза, 2011. – 304 с.: іл.
3. Казанцева О.П. **Інформатика: підручник для 7 класу загальноосвітніх навчальних закладів** / О.П. Казанцева, І.В. Стеценко, Л.В. Фурик. – Тернопіль: Навчальна книга – Богдан, 2015. – 180 с.
4. Морзе Н.В. **Інформатика: підручник для 7 класу загальноосвітніх навчальних закладів** / Н.В. Морзе, О.В. Барна, В.П. Вембер, О.Г. Кузьмінська – К.: Видавничий дім «Освіта», 2015
5. Мукосеєнко О.А. **Конспекти-сходинки як засіб побудови молодшими школярами багатоступінчастих карт пам'яті на уроках інформатики** / О.А. Мукосеєнко // Комп'ютер у школі та сім'ї. - 2015. - № 7. - С. 26-29.
6. Ривкінд Й.Я. **Інформатика: підручник для 7 класів загальноосвітніх навчальних закладів** / Й.Я. Ривкінд, Т.І. Лисенко, Л.А. Чернікова, В.В. Шакотько – К.: «Генеза», 2015
7. Електронний ресурс. Режим доступу: <https://yadi.sk/d/gFRB0m6m9ZAKe>
8. Олєхник С.Н., Нестеренко Ю.В., Потапов М.К. **Старинные занимательные задачи.**, М., 1985.
9. Шост Д.М. **Інформатика. Pascal. Зошит-конспект. 10-11 класи**

5. ПОСИЛАННЯ НА САЙТИ

- http://info.ho.ua/2012_2013_11kl_st_tema3.htm
http://uchitelska.at.ua/publ/starsha_shkola/informatika_11_klas/urok_5_etapi_rozv_jazuvanija_zadachi_za_dopomogiju_komp_jutera_ponjattja_programi/92-1-0-2762