

# Алгоритм

---

Підготував студент 16 – А групи  
Лозінський Едуард

# Що таке алгоритм

---

- **Алгоритм** – це скінчена послідовність вказівок (команд), формальне виконання яких дозволяє за обмежений час отримати розв'язок задачі.
- Сам термін “*алгоритм*” утворився в результаті перекладу на європейські мови імені арабського математика IX століття *Аль-Хорезмі*, який описав правила (алгоритми) виконання основних арифметичних операцій у десятковій системі числення.

- 
- У своїй практичній діяльності люди постійно мають справу із алгоритмами (послідовностями вказівок, інструкціями, правилами тощо). Для прикладу можна назвати приготування кулінарної страви згідно з рецептом, користування міжміським телефоном-автоматом, пошук слова у словнику, розв'язування квадратного рівняння.

# Властивості алгоритмів

---

- *Скінченність.* Виконання кожного алгоритму повинно завершуватись за скінченне число кроків.
- *Результативність.* Виконання алгоритму завжди повинно призводити до певного результату. Воно не може закінчуватись невизначеною ситуацією або ж не закінчуватися взагалі.

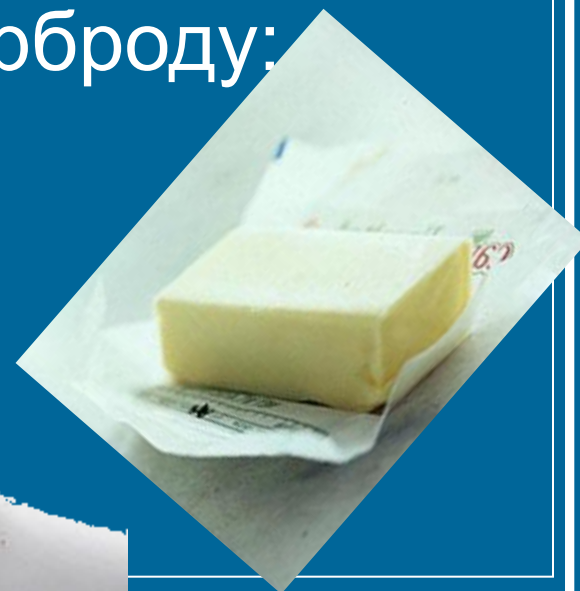
- *Формальність.* Виконавець відповідно до алгоритму повинен одержати результат, не вникаючи в його суть. Очевидно, що комп'ютери не можуть розуміти суть завдань і окремих вказівок алгоритму.
- *Визначеність.* Будь-який алгоритм повинен бути описаний так, щоб при його розшифруванні у виконавця не виникло двозначних вказівок. Тобто різні виконавці згідно з алгоритмом повинні діяти однаково та прийти до одного і того ж результату

- 
- *Масовість.* За допомогою складеного алгоритму повинен розв'язуватись цілий клас задач.
  - *Зрозумілість.* В алгоритмі повинні бути лише операції, які знайомі виконавцеві. Досконалим виконавцем алгоритмів обробки інформації є комп'ютер, робота якого здійснюється під керівництвом програм.

- 
- Алгоритми можна описувати за допомогою слів, спеціальних мов, використовуючи спеціальні формули, таблиці, графіки, блок-схеми, інші засоби. Алгоритм записується засобами мови, зрозумілої виконавцю. Для людини – це *природна мова*.

Для того, щоб краще зрозуміти, що таке алгоритм, опишемо процес приготування бутерброду, або іншими словами – алгоритм приготування бутерброду:

- Відріж хліб
- Намасти маслом
- Смачного





# Спроба №2

- Відріж 1 шматочок хліба
- Намасти маслом одну сторону
- Смачного



# Спеціально для \*... комп'ютера

- 1) В праву руку візьми за ручку ніж, в лівій руці тримай хліб.
- 2) Гострою стороною ножа відріж від хліба шматочок товщиною 1 см, а довжиною 10 см. Все відклади в сторону.
- 3) Візьми масло.
- 4) Гострою стороною ножа намасти шматочок хліба маслом з однієї сторони.
- 5) Кінець роботи.



# Бургомістр і алгоритм

---

- В одному німецькому місті бургомістр вночі зіштовхнувся з перехожим і набив собі гулю. Вранці він написав наказ: «Всім мешканцям міста вночі ходити з ліхтарями». Ввечері він пішов перевірити, як виконується його наказ. І знову набив гулю. «Чому ти без ліхтаря?» — «Ось він». — «Чому він без свічки?» — «Наказу не було».

- 
- Наступного дня з'явився наказ: «У ліхтарях повинна бути свічка». Знову бургомістр пішов перевіряти виконання свого наказу і знову набив гулю. «Чому без ліхтаря?» — «Ось він». — «Чому ліхтар без свічки?» — «Ось вона». — «Чому вона не запалена?» — «Не було наказу».

- І тільки на третій день вийшов вичерпний наказ: перехожі в темну пору доби повинні ходити з ліхтарями, у ліхтарях повинна бути свічка, свічка повинна бути запалена.



# Базові алгоритмічні структури

---

- Слідування
- Розгалуження
- Повтор

# Слідування

---

- Операція слідування подається у вигляді послідовності двох (або більше) простих операцій, що виконуються одна за одною. Якщо алгоритм складається лише з послідовності простих операцій, його називають простим або лінійним алгоритмом.



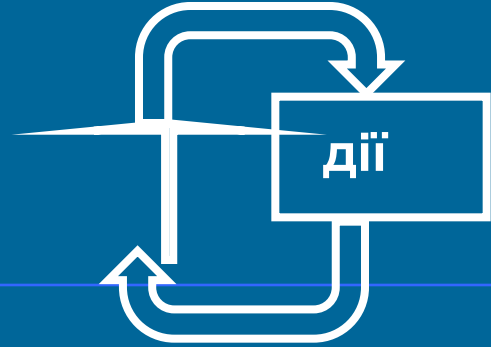
# Розгалуження (вибір)



- Операція розгалуження – це вказівка виконати одну з двох команд: команду1 або команду2, залежно від істинності чи хибності деякого твердження  $P$ . Якщо твердження  $P$  істинне, то виконується команда1. Якщо твердження  $P$  хибне, то виконується команда2. Особливим випадком розгалуження є неповне розгалуження, коли у разі хибності твердження  $P$  ніякі операції взагалі не виконуються.



# *Повторення (цикл)*



- Повторення команди або групи команд певну кількість разів або до виконання певної умови

За допомогою комбінацій цих трьох базових структур можна подати будь-який алгоритм.

# Блок-схема алгоритму

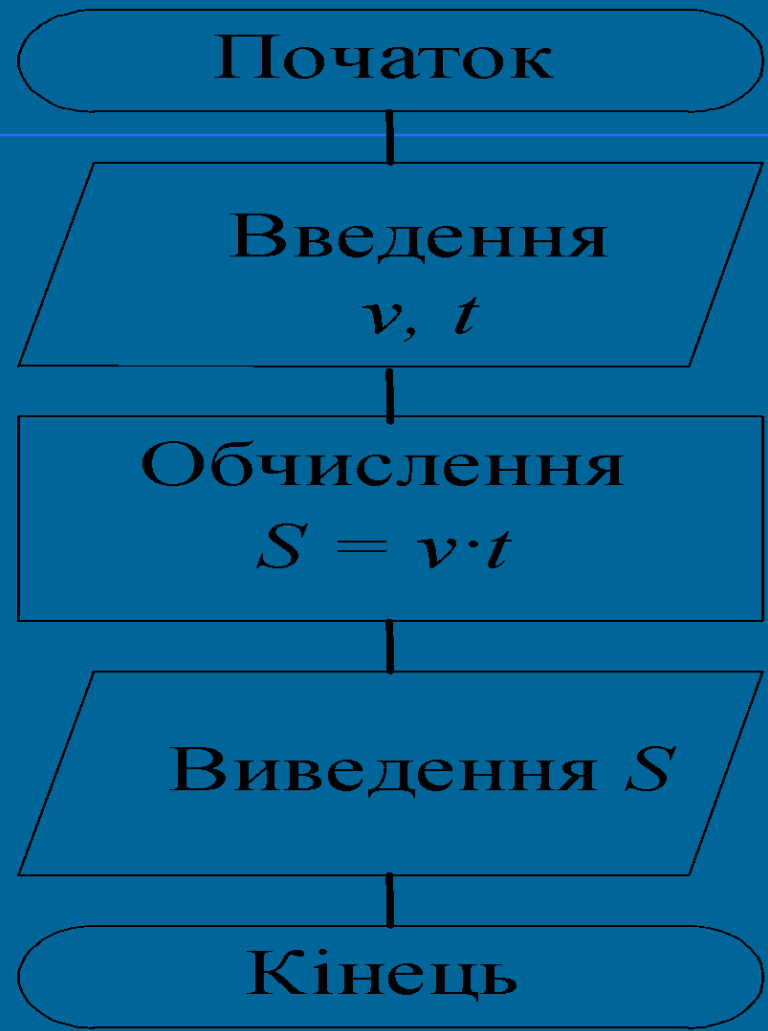
---

- Графічне зображення, на якому окремі дії алгоритму зображуються за допомогою геометричних фігур, а послідовність виконання дій вказується за допомогою ліній зі стрілками, які з'єднують ці фігури.
- Блок-схеми дозволяють наочно зобразити структуру алгоритму. На такій схемі добре видно послідовність виконання дій, а також цикли і розгалуження.

- 
- Геометричні фігури у блок-схемах називають блоками. Вони позначаються символами, які мають стандартне зображення і призначення.
  - У професійному програмуванні використовується до 30 різноманітних стандартних символів для зображення блок-схем.

# «Обчислити шлях за швидкістю $v$ і часом руху $t$ »

- Словесний запис алгоритму задачі буде таким:
  1. Ввести швидкість  $v$  і час руху  $t$ .
  2. Обчислити шлях за формулою  $S = v \cdot t$ .
  3. Вивести шлях  $S$ .



## *Алгоритм «Відгадай число»:*

---

- 1. Задумай будь-яке число.
- 2. Додай до нього 12.
- 3. Від результату відніми 7.
- 4. Відніми від результату задумане число.
- 5. Одержано число 5.

«Як перевезти по одному через річку без втрат вовка, козу і капусту»:

---

1. Переправити на той берег козу, вовка залишити з капустою;
2. Повернутись, взяти вовка, переправитись з ним до кози;
3. Забрати козу і повернутись назад до капусти;
4. Залишити козу, забрати і перевезти капусту до вовка;
5. Повернутись і забрати козу.

## *Алгоритм «Користування телефоном»:*

---

- 1. зняти трубку;
- 2. почувши гудок, набрати номер;
- 3. якщо з'єднання відбулось — говорити;
- 4. якщо з'єднання не відбулось — покласти трубку і перейти до п.1.

Алгоритм знаходження найбільшого спільного дільника (НСД) двох натуральних чисел вперше описав Евклід:

- 1. Порівняй числа  $a$  і  $b$ .
- 2. Якщо  $a = b$ , то  $a$  найбільший спільний дільник.
- 3. Якщо  $a > b$ , то замінити  $a$  на  $a - b$ .
- 4. Якщо  $a < b$ , то замінити  $b$  на  $b - a$ .
- 5. Перейти до п. 1.



# Домашнє завдання: блок-схема

---

- Візьми лопату
  - Постав лопату
  - Візьми відро
  - Постав відро
  - Візьми саджанець
  - Постав саджанець
  - Викомай ямку
  - Засип ямку
  - Постав у ямку
  - Полий водою
  - Пройди вперед
- Є кілька умов:
    - В руках у садівника може бути лише 1 предмет
    - Перед засипанням ямки із саджанцем потрібно полити його водою
    - Після засипання ямки із саджанцем потрібно полити його водою – щоб він розквітнув
    - Перед тим, як перейти до наступного саджанця – не забути лопату!

- 
- Для комп'ютера мова складається з нулів та одиниць. Використання такої мови для складання програм є неефективним. Тому використовуються спеціальні мови – *мови програмування.*

- 
- Мова програмування дозволяє записувати команди у такій формі, щоб їх можна було автоматично замінити на машинні коди. Це перетворення здійснюється автоматично за допомогою спеціальних програм-перекладачів, які називаються трансляторами.

# Мова програмування Паскаль

---

- Одна із найпопулярніших мов програмування - це мова *Паскаль*, яку створив у 1968 році швейцарський вчений Ніклаус Вірт. Вона дозволяє записувати команди, завдяки яким комп'ютер може розв'язувати математичні задачі, обробляти тексти, будувати зображення на екрані дисплея.

- 
- Усі слова, з яких складають програми, поділяють на службові (зарезервовані) та ідентифікатори користувача.
  - *Службові слова* мають наперед визначене значення і використовуються для формування структури програми, здійснення описів, позначення операцій, формування керуючих конструкцій.

- 
- Наприклад, службовими словами для мови Паскаль є: *and, or, not, program, label, const, type, var, procedure, function, array, record, set, file, begin, end, if, then, else, for, to, downto, do, repeat, until, while, case, goto* тощо.

- 
- *Ідентифікатор* - це послідовність латинських літер, цифр і знаку підкреслення, яка розпочинається з латинської літери. У мові Паскаль максимальна довжина ідентифікатора становить 127 символів. У ідентифікаторах не можна використовувати знак пробілу.

# Структура програми на мові Паскаль

---

- *Program* *назва*;
- *Var* *розділ опису змінних*;
- *Begin*  
*команди*;
- *End.*



# Найпростіша програма

---

- Program simple;
- Begin
- Write ('hello');
- End.