

Технології розробки алгоритмів розв'язання інженерних задач

Лекція №1

Викладач: Дрєєв Олександр Миколайович

1. Введення до предмету

1.1. Означення алгоритму

1.2. Властивості алгоритму

1.3. Умови до можливості виконання
алгоритму

1.4. Способи завдання алгоритму. Блок-схема,
функціональна схема, схема потоків
інформації.

1.5. Вимоги до оформлення робіт

Введення до предмету

Означення алгоритму

Алгоритм — скінченна заздалегіть визначена послідовність дій, виконання яких призводить до розв'язання групи задач.

Дії повинні бути зрозумілими виконавцю та виконавець повинен бути здатен їх виконати. Інтелект має здатність створювати нові алгоритми.

Введення до предмету

Властивості алгоритму

Скінченність — для виконання дій потрібна скінченна кількість часу.

1. Підстрибни.

2. Перейди до п.1

Це не є алгоритмом.

Правильність — алгоритм дає правильну відповідь або визначає її відсутність.

Складність — вказує на кількість операцій потрібних для отримання результату та залежність їх кількості від кількості вхідних даних.

Введення до предмету

Властивості алгоритму

Складність задають $O()$ символікою.

1. $i=0$. $S=0$.
2. Число з номером i додаємо до S
3. i збільшуємо на 1
4. Якщо є ще числа, тоді до п.2

Складність $O(n)$

1. Розсадити n чоловік на n стільців так, як вони ще не сідали.
2. Якщо є ще варіанти, то до п.1

Складність $O(n!)$

Введення до предмету

Властивості алгоритму

Зрозумілість — всі команди повинен знати виконавець

Однозначність — всі команди трактуються єдиним чином

Введення до предмету

Умови можливості виконання

- Виконавець розуміє мову
 - Алгоритм забезпечує скінченність кількості кроків (відсутні зациклювання, це наслідок правильності алгоритму)
 - Виконавець має доступ до вхідних даних
 - Виконавцю достатньо пам'яті та часу
 - Алгоритм використовує інші вже відомі алгоритми
 - Алгоритм можна виконати, навіть якщо він дає неправильні результати
-
-

Введення до предмету

Способи запису алгоритму

Описовий, рецепт — алогоритм для людини, команди не однозначні та їх тлумачення залежить від контексту.

Текстовий — алгоритм задається послідовним текстом, можливо по крокам, як на попередніх слайдах.

Блок-схемою — кожна дія відокремлена геометричною фігурою, про зміст якої домовлено заздалегіть; порядок переходу між діями задано лініями та стрілками.

Введення до предмету

Способи запису алгоритму

Алгоритмічною мовою — алогоритм текстовий, але використана штучна мова, наприклад для полегшення порозуміння з автоматом-виконавцем.

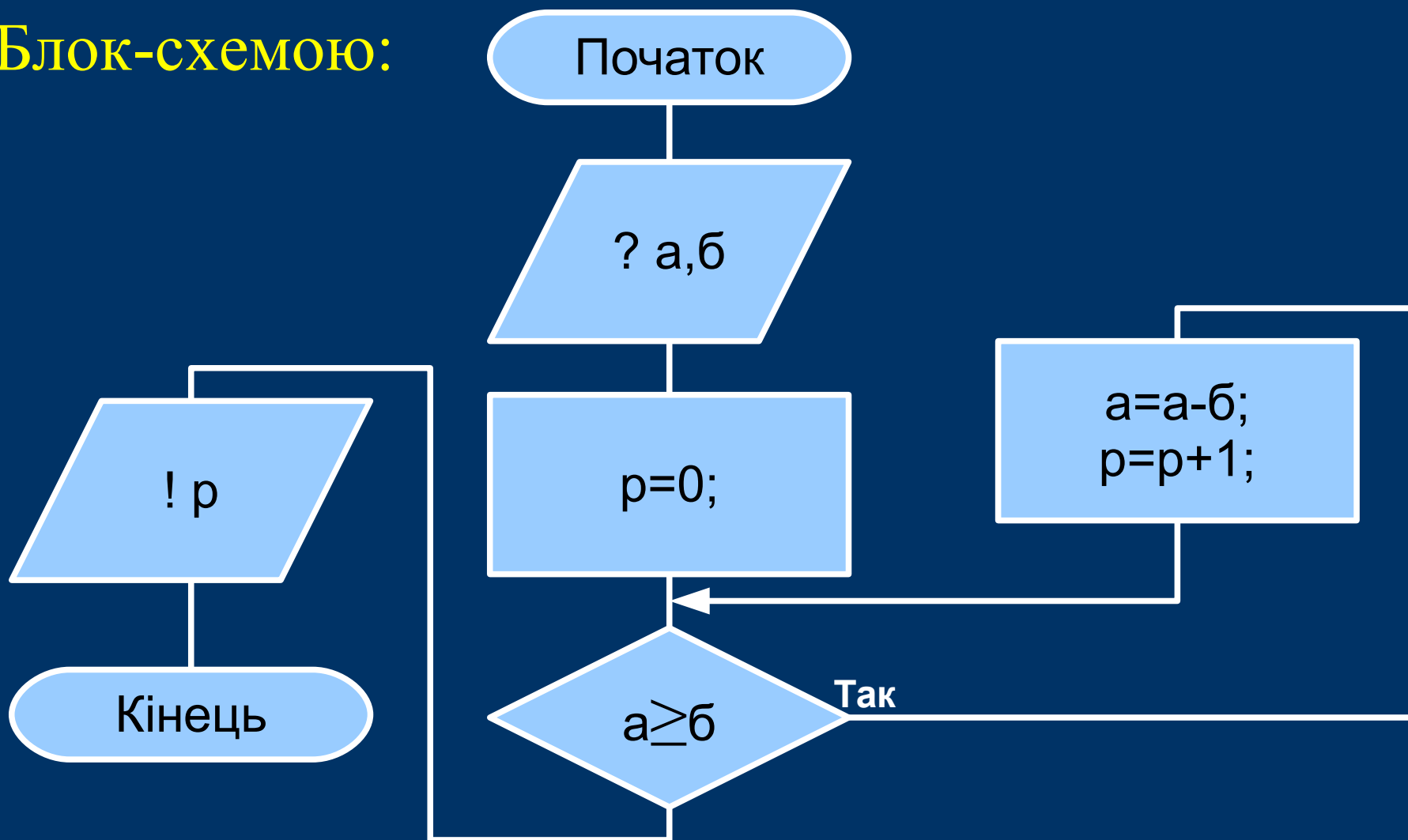
Функціональною схемою — алгоритм задається зв'язками між функціональними блоками; добре пояснює механізм обробки інформації.

Схема потоків інформації — вказує на джерела та приймачі інформації, відображає паралельні та незалежні потоки обробки інформації; акцент на послідовності обробки.

Введення до предмету

Способи запису алгоритму

Блок-схемою:



Введення до предмету

Способи запису алгоритму

Алгоритмічною мовою:

Вхід: a - ділене;

Вхід: b - ділитель;

Вихід: p - результат ділення;

$p=0$;

Поки $a \geq b$ роби

(

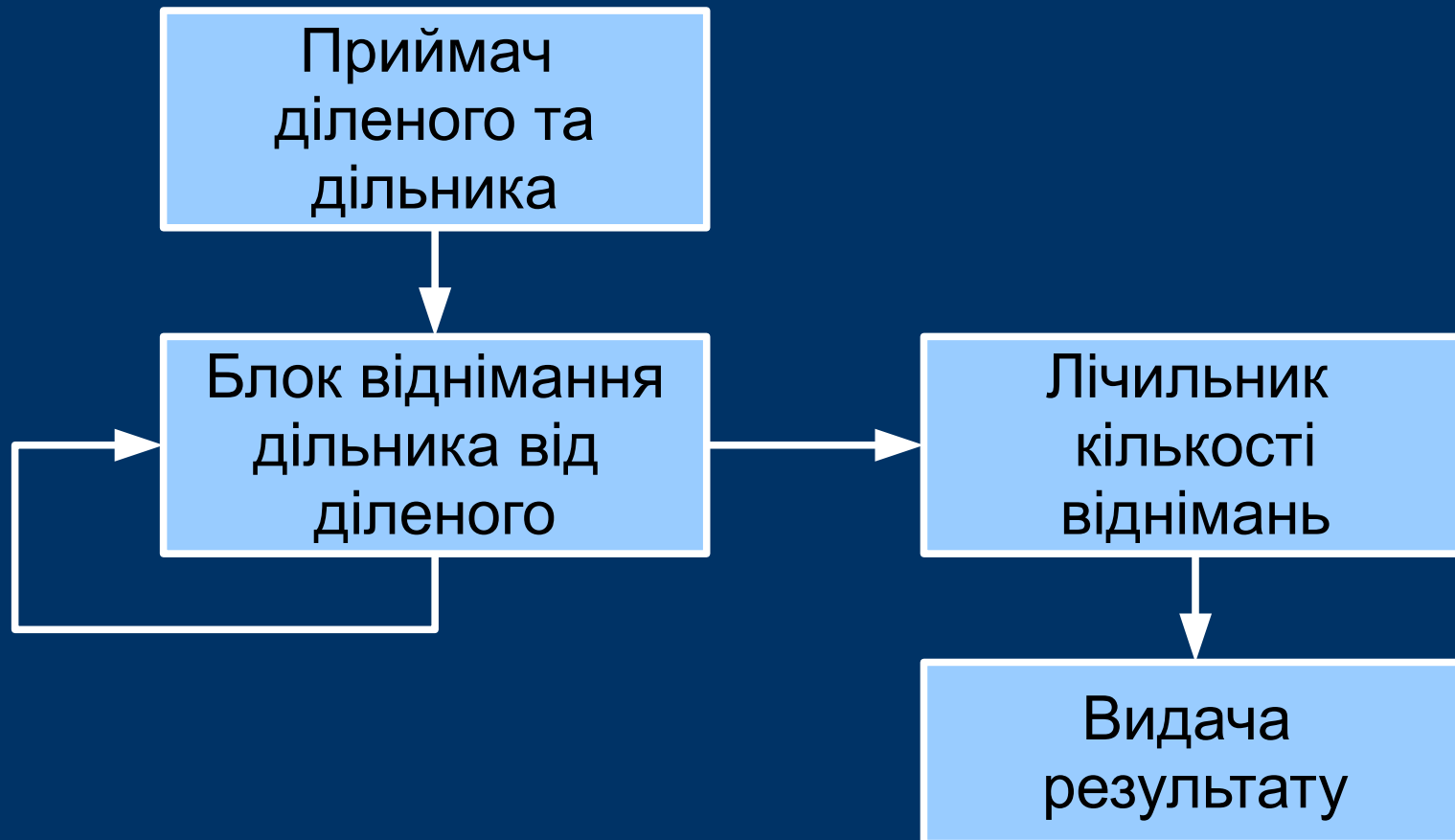
$a=a-b, \quad p=p+1$;

);

Введення до предмету

Способи запису алгоритму

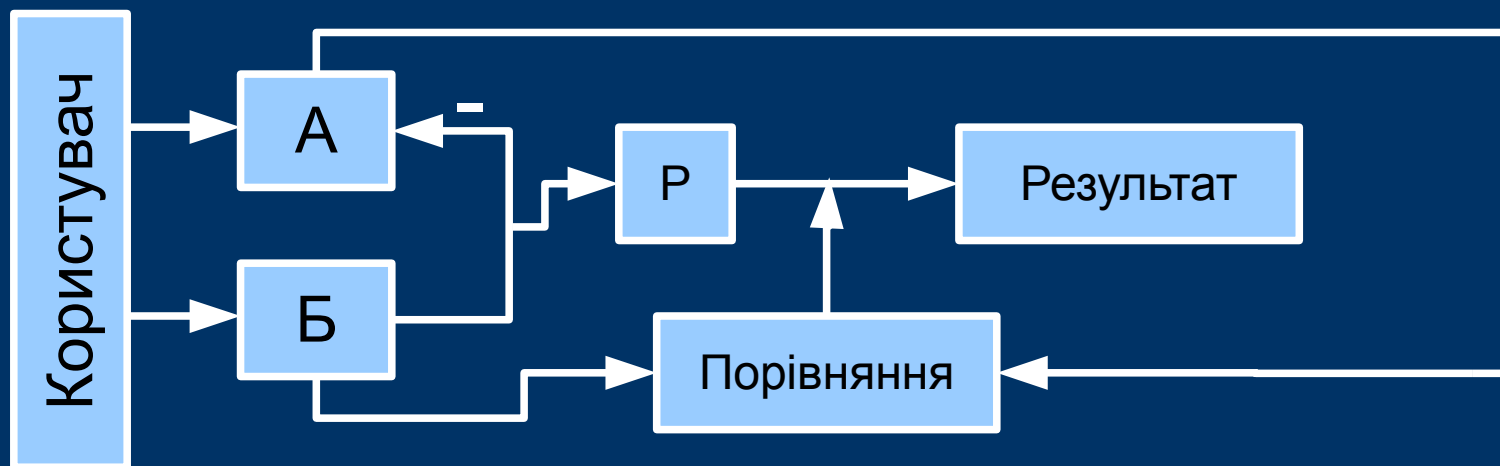
Функціональна схема:



Введення до предмету

Способи запису алгоритму

Схема потоків інформації:



Введення до предмету

Якість алгоритму

Якість алгоритму:

- 1) Мінімальна складність $O()$
 - 2) Універсальність, коло допустимих задач
 - 3) Захист від неправильного використання
 - 4) Максимальна швидкодія
 - 5) Мінімальність кількості команд
 - 6) Мінімальність потоків інформації
 - 7) Поліморфність, допустимість зміни структури
 - 8) Простота
-
-

Введення до предмету

Вимоги до оформлення робіт

Роботи оформлюються у звітах з стандартним титульним аркушом, або в окремому підписаному зошиті.

Робота містить:

- 1) Текст завдання
 - 2) Текстом: основна ідея розв'язання
 - 3) Структура алгоритму
 - 4) Потoki інформації в алгоритмі
 - 5) Блок-схема алгоритму, загальна
-
-

Введення до предмету

Вимоги до оформлення робіт

- 6) Блок-схеми допоміжних алгоритмів (при потребі чи за вимогою викладача)
 - 7) Запис мовою програмування
 - 8) Оцінка складності алгоритму теоретично та експериментально
 - 9) Висновки (чи виконано завдання, як по іншому можна було розв'язати задачу, чи є алгоритм якісним, в яких межах вхідних даних можна використати алгоритм)
-
-

Введення до предмету

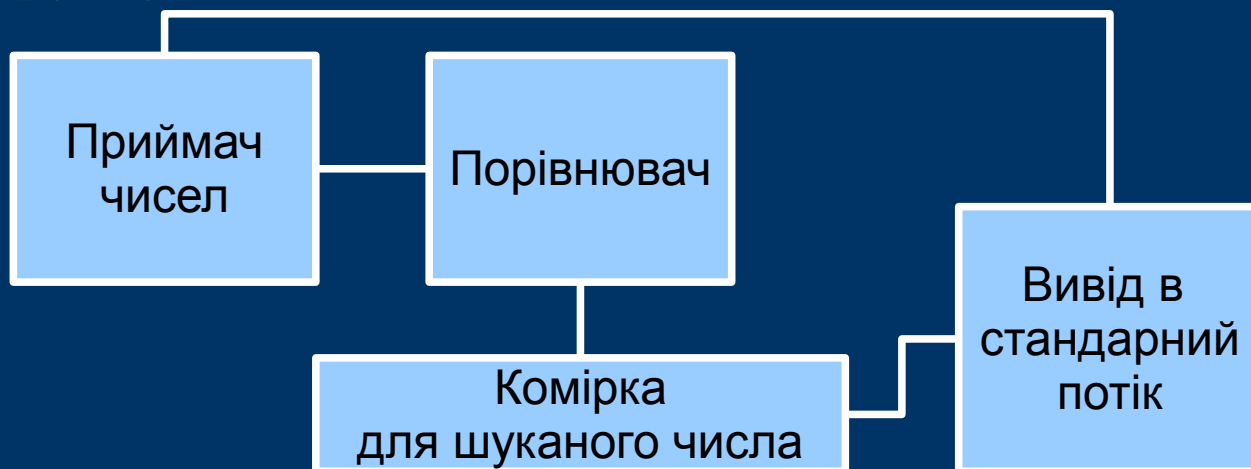
Вимоги до оформлення робіт

Приклад.

Завдання: В послідовності цілих чисел, зі стандартного потоку даних, розділених знаком переведення строки знайти найбільше число і вивести його в стандартний потік даних.

Ідея розв'язання: Перше число вважаємо найбільшим. Поки в потоці вводу даних є числа, читаємо одне, і якщо воно більше за вже знайдене, вважаємо його новим найбільшим числом.

Структурна схема:

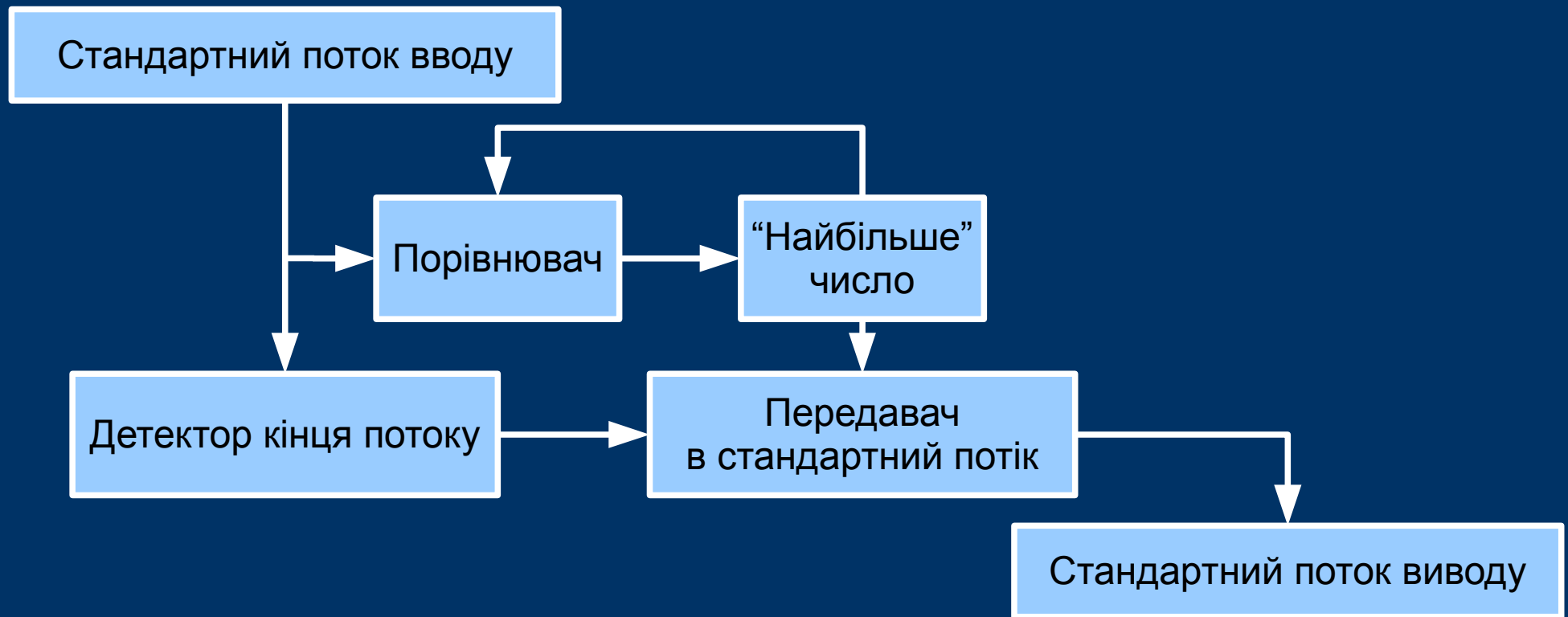


Введення до предмету

Вимоги до оформлення робіт

Приклад.

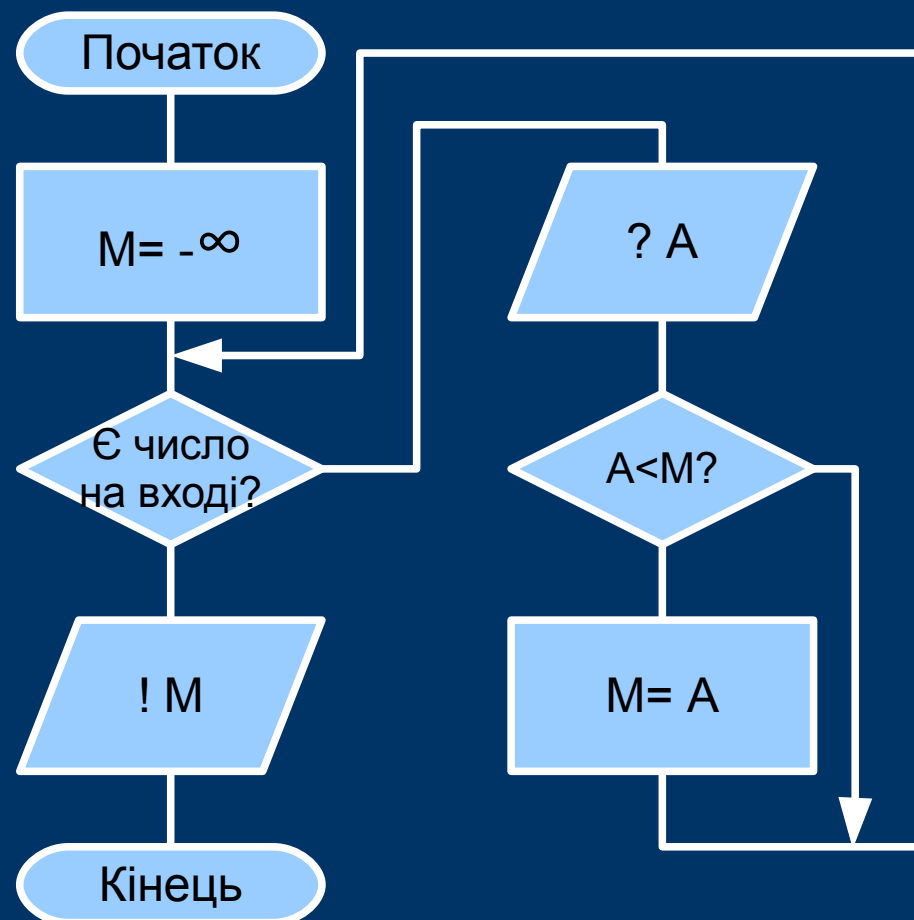
Схема потоків інформації:



Введення до предмету

Вимоги до оформлення робіт

Приклад.
Блок-схема алгоритму:



Введення до предмету

Вимоги до оформлення робіт

```
#include <iostream>
int main(int N,char* Param[])
{
    int M = 0x80000000; //32-х розрядне найменше ціле
    int A = 0;
    while( !std::cin.eof() ) //Доки не кінець потоку вводу, повторити
    {
        std::cin >> A; //Читаємо число
        if( A>M ) M = A; //Якщо більше, то тепер воно в M
    }
    std::cout << M << std::endl; //Вивід найбільшого
    return 0;
}
```

Введення до предмету

Вимоги до оформлення робіт

Файл data.txt з пробними числами:

8

5

45

56

45

32

Запуск програми на виконання:

```
C:\CPR\WIN32>program.exe < data.txt > result.txt
```

Вміст файлу result.txt:

56

Програма працює вірно.

Введення до предмету

Вимоги до оформлення робіт

Оцінка складності:

При виконанні програми, алгоритм порівнює кожний вхідний елемент з одним елементом, тому час виконання програми пропорційний кількості елементів. Складність алгоритму $O(n)$.

data.txt — 1341 цілих чисел

Запуск: `tmetric.exe find_max.exe < data.txt`

Результат: 84 ms

data.txt — 2682 цілих числа

Запуск: `tmetric.exe find_max.exe < data.txt`

Результат: 172 ms

При збільшенні вхідних даних вдвічі, час роботи збільшився вдвічі. Теоритична оцінка збігається з практичною.

Введення до предмету

Вимоги до оформлення робіт

Висновок:

Розроблено алгоритм, програму відшукування максимального числа з набору чисел. Програма пройшла тестові завдання й працює правильно. Алгоритм має лінійну складність. Програма може бути використана як фільтр в формуванні командних файлів.

При пошуку найбільшого числа потрібно перевіряти всі числа, тому алгоритм з меншою складністю створити не можна.

Шляхів удосконалення програми не знайдено.

ВІДПОВІДІ НА КОНТРОЛЬНІ ПИТАННЯ
