МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"



Звіт

з лабораторної роботи № 1-2-

З дисципліни «Програмування , частина 2»

Виконав:

Ст. гр. ТР-12

Дербаба Олександра

Прийняв:

Асистент. Каф. ТК

Гордійчук-Бублівська О.В.

Львів - 2023

**Лабораторна робота №1 -2**

«Основи алгоритмізації»

**Мета роботи**: ознайомитися із загальними принципами розробки алгоритмів, навчитися виконувати задачі щодо побудови різних типів алгоритмів.

**Теоретичні відомості**

Поняття алгоритму інтуїтивно зрозуміло та часто використовується в математиці та комп'ютерних науках. Говорячи неформально, алгоритм - це довільна коректно визначена обчислювальна процедура, на вхід якої подається деяка величина або набір величин, а результатом виконання якої є вихідна величина або набір значень.

Таким чином, алгоритм є послідовністю обчислювальних кроків, які перетворюють вхідні величини у вихідні. Алгоритм можна також розглядати як інструмент, який призначений для вирішення коректно поставленої обчислювальної задачі. У постановці задачі в загальних рисах визначаються відношення між входом та виходом.

В алгоритмі описується конкретна обчислювальна процедура, за допомогою якої можна досягнути виконання вказаних відношень. Можна навести загальні риси алгоритму:

• а. Дискретність інформації. Кожний алгоритм працює із даними: вхідними, проміжними, вихідними. Ці дані представляються у вигляді скінченних слів деякого алфавіту.

• b. Дискретність роботи алгоритму. Алгоритм виконується по кроках та при цьому на кожному кроці виконується тільки одна операція.

• c. Детермінованість алгоритму. Система величин, які отримуються в кожний (не початковий) момент часу, однозначно визначається системою величини, які були отримані в попередні моменти часу.

• d. Елементарність кроків алгоритму. Закон отримання наступної системи величин з попередньої повинен бути простим та локальним.

• e. Виконуваність операцій. В алгоритмі не має бути не виконуваних операцій. Наприклад, неможна в програмі призначити значення змінній «нескінченність», така операція була би не виконуваною. Кожна операція опрацьовує певну ділянку у слові, яке обробляється.

• f. Скінченність алгоритму. Опис алгоритму повинен бути скінченним.

• g. Спрямованість алгоритму. Якщо спосіб отримання наступної величини з деякої заданої величини не дає результату, то має бути вказано, що треба вважати результатом алгоритму.

• h. Масовість алгоритму. Початкова система величин може обиратись з деякої потенційно нескінченної множини. Розглянемо для прикладу задачу сортування послідовності чисел у зростаючому порядку. Ця задача часто виникає на практиці і, фактично, буде центральною проблемою першого розділу даного курсу.

**Хід роботи:**

1. Ознайомитися з теоретичними відомостями.

2. Побудувати блок-схему у середовищі Microsoft Visio для вирішення такого завдання: «Знайти найменше із трьох заданих чисел a,b,c. (Вхідними даними є числа a,b,c, вихідними- найменше число).»

Початок

a, b, c

ТАК

a<b і a<c

Вивести a

Ні

a<b і a<c

Вивести b

ТАК

Ні

Вивести c

Кінець

3. Побудувати блок-схему у середовищі Microsoft Visio для вирішення такого завдання: «Дано два кола з центрами в початку координат і радіусами, рівними 2 і 4 відповідно. Ввести координати точки і визначити чи попадає вона в коло першого радіуса чи знаходиться поза ним, але попадає в коло більшого радіуса , чи знаходиться за межами кіл. Вивести 1 для першого випадку, 2 – для другого, в протилежному випадку – 0.»

ТАК

ТАК

Ні

Ні

Кінець

Вивести 1

Вивести 0

Вивести 2

a<=R1

a<=R2

a=sqrt(x^2+y^2)

Початок

Ввести x і y

R1=2, R2=4

x, y, a

4. Побудувати блок-схему у середовищі Microsoft Visio для вирішення такого завдання: «Представити алгоритм обчислення суми і добутку тільки додатніх чисел із n вхідних, і виводу отриманих значень суми і добутку .» »

ТАК

ТАК

Ні

Ні

Кінець

Вивести s i p

i++

s=s+a\_i, p=p\*a\_i

a\_i<0

i<n

Ввести n, числа від a\_1 до a\_n

n, s=0, p=1, a[], i=1

Початок

Висновок: у цій лабораторній роботі я ознайомитися із загальними принципами розробки алгоритмів, навчитися виконувати задачі щодо побудови різних типів алгоритмів.