

План лабораторного заняття № 6-7

Тема № 3. ОСНОВИ АЛГОРИТМІЗАЦІЇ: БАЗОВІ АЛГОРИТМІЧНІ СТРУКТУРИ

Навчальний час: 4 год.

Міжпредметні зв'язки: Зв'язок із елементами знань і умінь таких навчальних дисциплін як „Алгоритмізація та програмування”, „Інформатика” та „Об'єктно-орієнтоване програмування”.

Мета і завдання лабораторного заняття: набуття практичних навичок складання блок-схем та роботи із відповідними сервісами.

Питання для перевірки базових знань за темою лабораторного заняття:

1. Які алгоритми називаються лінійними?
2. Які алгоритми називаються розгалуженими?
3. Які алгоритми називають циклічними?
4. Які конструкції є базовими для зображення алгоритму?
5. Яким властивостями володіє алгоритм?
6. Які є сервіси для створення блок-схем?
7. Що таке цикл?
8. Що таке розгалуження?
9. Яке призначення програми Microsoft Visio?

Завдання:

I. У програмі Microsoft Visio побудувати блок-схему алгоритму:

1. знаходження більшого з трьох чисел a , b , і c . Результат присвоїти змінній u ;
2. виведення на екран значення функції $y=x^2$ для x від -5 до 5 з кроком 1 ;
3. знаходження суми добутоків всіх сусідніх одноцифрових чисел;
4. визначення чи попадає точка (x,y) в коло радіуса 50 і центром в точці $(0,0)$;
5. визначення чи поміститься квадрат із стороною A в колі радіуса R ;
6. встановлення чи існує трикутник із сторонами A , B , C ;
7. знаходження суми всіх чисел кратних трьом від 10 до N ;
8. знаходження суми всіх парних чисел від N до 50 ;
9. знаходження суми кубів всіх двохзначних чисел кратних 5 ;
10. знаходження суми коренів квадратних цілих чисел від 0 до 50 ;
11. знаходження добутку всіх чисел, які діляться без остачі на 4 і лежать в межах від -20 до 20 ;
12. знаходження суми всіх чисел кратних трьом з відрізка $[n,50]$;
13. знаходження суми всіх парних чисел від N до 50 ;

$$Y = \begin{cases} \frac{abx^2}{b+x}, & x = 2. \\ a + \sin(bx), & 2 < x \leq 8. \\ \sqrt{|ab+x|}, & x > 8. \end{cases};$$

14. обчислення значення виразу:

15. обчислення значення виразу:

$$Y = \sqrt[4]{100};$$

16. виведення на екран усіх дільників заданого числа, що є простими числами (1 та саме число не враховувати) або вивести повідомлення, що введене число є простим;

17. обчислення значення виразу $Y = a + \sin(bx)$ для заданих значень коефіцієнтів a і b та аргумента x , що змінюється від -4 до 6 з кроком 2 ;

18. визначення суми чисел натурального ряду доти, доки остаточне значення суми не перевищить 1000 ;

19. знаходження суми всіх непарних чисел від -100 до 100 ;

20. виведення на екран чисел, протилежних до чисел кратних трьом, від 1 до 50 ;

21. виведення на екран чисел, протилежних до чисел кратних трьом, від 1 до 50 ;

22. визначення чи знаходиться точка $M(a,b)$ всередині прямокутника зі сторонами A, B . Початок координат співпадає з точкою перетину діагоналей прямокутника і сторони паралельні осям координат;

23. підрахунок кількості пар взаємно обернених чисел серед трьох чисел a, b, c ;

24. знаходження суми кубів всіх двохзначних чисел кратних 6 ;

25. обчислення добутку членів послідовності $D = -1 * (1/2) * (-1/3) * (1/4) * (-1/5) * \dots * (-1/2N-1) * (1/2N)$;

26. обчислення суми членів послідовності $S = 1 - 2 + 3 - 4 + 5 - 6 + \dots - 2N$.

II. Для коду програми, написаної мовою Pascal, C++ тощо, скласти блок схему за допомогою відповідних сервісів.

III. Оформити звіт про виконання лабораторної роботи.

Звіт про виконання роботи подається у друкованому форматі на папері формату A4.

Оформлення звіту: шрифт – Times New Roman; розмір шрифту – 14 кегель; інтервал між рядками – півтора; абзац – 12,5 мм, поля: верхнє і нижнє – 20 мм, лівє – 25 мм, правє – 15 мм; нумерація сторінок – по центру нижнього поля.


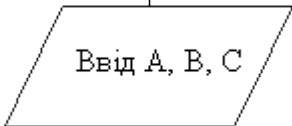
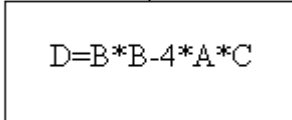
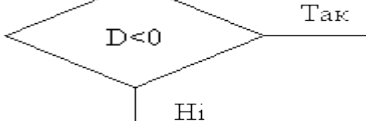
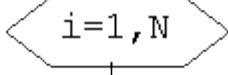
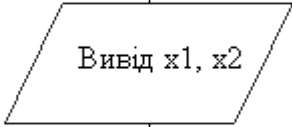
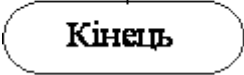
Звіт повинен мати наступну структуру:

- титульна сторінка.
- формулювання завдання.
- результати виконання завдання.
- література.

Теоретичні відомості та методичні вказівки для виконання

Блок-схема алгоритму – це графічне представлення логічної структури алгоритму, де кожний етап обробки інформації зображається у вигляді геометричних символів (блоків).

Існують правила зображення блок-схем алгоритмів. Кожен алгоритм має початок та кінець. Кожна команда алгоритму представляється у вигляді геометричних символів, які мають певну конфігурацію, в залежності від характеру дій, що будуть виконуватись. Геометричні символи з'єднуються між собою лініями або стрілками, які вказують порядок виконання дій.

Вигляд блоку	Призначення
	Початок алгоритму
	Блок вводу вхідних даних
	Блок обробки інформації. В блоці пишуться формули.
	Блок умови. В цьому блоці пишеться умова, в залежності від якої вибираються напрямки дії алгоритму.
	Заголовок циклу FOR
	Блок виведення результату
	Кінець алгоритму

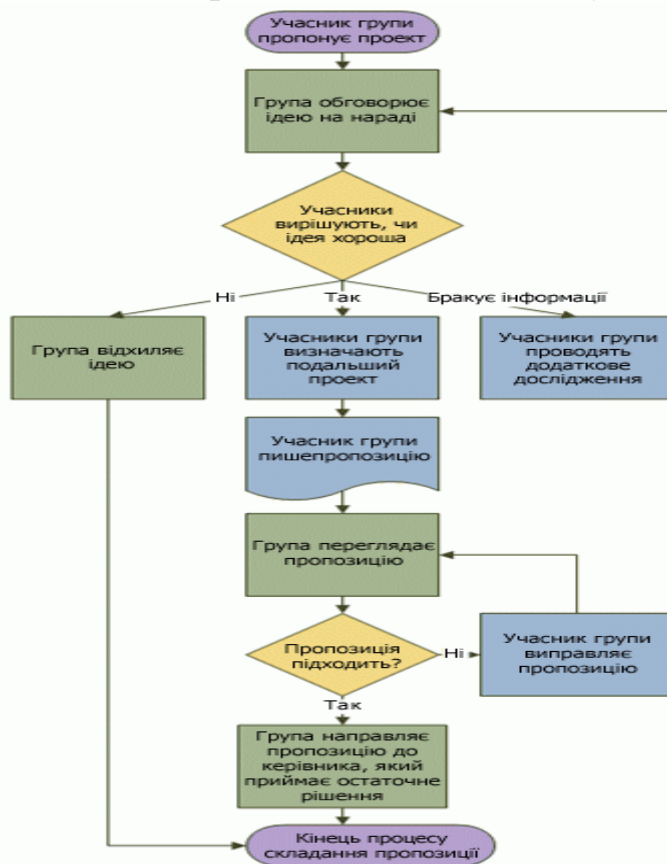
Створення простої блок-схеми у Microsoft Visio:

1. Запустити програму «Visio».

2. Обрати категорію **Блок-схема**.
3. Двічі клацнути **Проста блок-схема**.
4. Перетягнути на сторінку креслення фігури блок-схеми, що відповідають кожному кроку процесу, який документується.
5. Поєднати фігури блок-схеми, тримаючи стрілку миші над першою фігурою, а потім натиснувши маленьку стрілку, що з'явилася і показує на фігуру, яку необхідно з'єднати. Якщо друга фігура розташована на іншій сторінці, ніж інша фігура, натиснути і тримати маленьку стрілку, тягнути її до другої фігури, і відпустити з'єднувач на середині другої фігури.
6. Для введення тексту до фігури або сполучної лінії їх потрібно виділити. Після введення тексту клацнути будь-де в пустому місці сторінки.
7. Щоб змінити напрямок стрілки сполучної лінії, потрібно виділити сполучну лінію, а потім на вкладці **Фігури** у групі **Стилі фігур** натиснути **Лінія**, навести вказівник на пункт **Стрілки** та вибрати потрібний напрямок стрілки.

Виправлення інтервалів та вирівнювання

1. Натиснути сполучення клавіш Ctrl + A, щоб виділити усе на сторінці креслення.
2. На вкладці **Основне** в групі **Упорядкування** натиснути кнопку **Розташування**, а потім вибрати **Автоматично вирівняти та розташувати**. Якщо ця команда не спрацьовує, натиснути сполучення клавіш Control+Z, щоб скасувати дію, а потім спробувати використати інші параметри під кнопками **Вирівнювання** та **Розташування**.



Призначення фігур блок-схеми

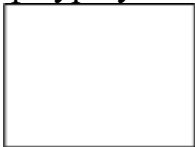
Під час відкриття шаблону **Проста блок-схема** також відкривається трафарет **Фігури простої блок-схеми**. Кожна фігура шаблону зображає окремий етап процесу. Однак, стандартне універсальне значення фігур відсутнє – будь-яка фігура може мати будь-яке значення, зрозуміле тим, хто створює і читає блок-схему. У більшості блок-схем використовується три або чотири фігури, якщо немає вагомий причини використовувати більше фігур.

Проте фігури Visio мають назви, які пояснюють їхнє використання. Найпоширеніші з них наведено нижче.

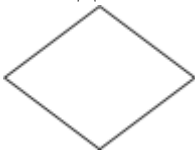
- **Початок/Кінець** Ця фігура використовується для позначення першого та останнього кроку процесу.



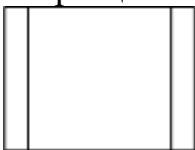
- **Процес** Ця фігура зображає етап у процесі. Це найбільш використовувана фігура у майже усіх процесах.



- **Рішення.** Ця фігура позначає точку, де наступний крок залежить від результатів рішення. Можуть мати місце кілька результатів, але у більшості випадків їх тільки два – так чи ні.



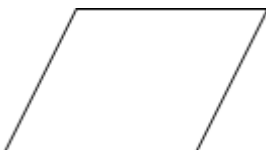
- **Підпроцес** Ця фігура використовується для позначення кількох етапів, які разом створюють підпроцес, визначений в іншому місці, часто на іншій сторінці того самого документа. Це зручно, коли схема велика та складна.



- **Документ.** Ця фігура зображає крок, результат якого – створення документа.



- **Дані.** Ця фігура вказує на надходження інформації до процесу ззовні або на її вихід із процесу. Ця фігура також використовується для зображення матеріалів та інколи називається фігура «Ввід/Вивід».



- **Посилання в межах сторінки.** Це маленьке коло вказує на те, що наступний (або попередній) крок розташований в іншому місці на цій сторінці креслення.

Особливо це зручно для великих блок-схем, на яких в іншому разі довелося б використовувати довгу сполучну лінію, яку важко простежити.



- **Посилання на іншу сторінку.** Після перетягування цієї фігури на сторінку креслення відкривається діалогове вікно, в якому можна створити набір гіперпосилань між двома сторінками блок-схеми або між фігурою підпроцесу та окремою сторінкою блок-схеми, на якій зображено кроки такого підпроцесу.



Форми контролю знань – презентація виконаних завдань у вигляді програми, презентації, обговорення виконаних завдань.

Рекомендована література до теми лабораторного заняття:

Основна та допоміжна література:

1. Алгоритми і структура даних: Навчальний посібник / В.М.Ткачук. - Івано-Франківськ : Видавництво Прикарпатського національного університету імені Василя Стефаника, 2016.-286 с.
2. Алгоритми та структури даних. Навчальний посібник / Т. О. Коротєєва. Львів : Видавництво Львівської політехніки, 2014. - 280 с.
3. Глоба Л. С. Розробка інформаційних ресурсів та систем [Електронний ресурс] : конспект лекцій / Л. С. Глоба, Т. М. Кот. - Київ : НТУУ "КПІ", 2014. - 318 с.
4. Шевчук І. Б. Інформаційні технології в регіональній економіці: теорія і практика впровадження та використання : монографія. Львів : Видавництво ННБК "АТБ", 2018. 448 с.

Інтернет ресурси:

1. AvtoShema 2.2 : [Електрон. ресурс]. - Режим доступу: <https://www.softportal.com/software-6628-avtoshema.html>
2. Створення та збереження документів Microsoft Office Visio. Створення простих схем : [Електрон. ресурс]. - Режим доступу: http://ito.vspu.net/ENK/2011-2012/spec_informat/LABS/lr_vizio1.htm
3. Microsoft Visio : [Електрон. ресурс]. - Режим доступу: https://uk.wikipedia.org/wiki/Microsoft_Visio
4. Блок-схеми онлайн : [Електрон. ресурс]. - Режим доступу: <http://geek-nose.com/blok-sxemy-onlajn-kak-strukturno-predstavit-informaciyu/>
5. Мойсеенко Е.В., Лаврушина Е.Г. Информационные технологии в экономике : [Електрон. ресурс]. - Режим доступу: https://abc.vvsu.ru/books/inform_tehnolog/default.asp
6. Построитель блок-схем для Pascal : [Електрон. ресурс]. - Режим доступу: https://www.studmed.ru/bs-v10-postroitel-blok-shem-dlya-pascal_9aebfd8f021.html

7. Рагулин П.Г. Информационные технологии: электронный учебник : [Электрон. ресурс]. - Режим доступа: http://window.edu.ru/catalog/pdf2txt/007/41007/18312?p_page=4
8. Створення простої блок-схеми : [Електрон. ресурс]. - Режим доступу: <https://support.office.com/uk-ua/article/%D0%A1%D1%82%D0%B2%D0%BE%D1%80%D0%B5%D0%BD%D0%BD%D1%8F-%D0%BF%D1%80%D0%BE%D1%81%D1%82%D0%BE%D1%97-%D0%B1%D0%BB%D0%BE%D0%BA-%D1%81%D1%85%D0%B5%D0%BC%D0%B8-e207d975-4a51-4bfa-a356-eeec314bd276>
9. Теоретические основы технологии программирования: [Электрон. ресурс]. - Режим доступа: <http://bourabai.kz/alg/technology.htm>
10. Технология программирования : [Электрон. ресурс]. - Режим доступа: https://studref.com/441961/informatika/tehnologiya_programmirovaniya
11. Установка Visio : [Электрон. ресурс]. - Режим доступа: <https://support.office.com/ru-ru/article/%D0%A3%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-visio-f98f21e3-aa02-4827-9167-ddab5b025710>
12. Visustin — генератор блок-схем : [Электрон. ресурс]. - Режим доступа: <https://www.aivosto.com/visustin-ru.html>
13. AFCE Редактор Блок-Схем (Algorithm Flowchart Editor) 0.9.8 : [Электрон. ресурс]. - Режим доступа: <https://lumpics.ru/algorithm-flowchart-editor/>
14. Редактор для построения блок схем. Блок схемы онлайн: как структурно представить информацию : [Электрон. ресурс]. - Режим доступа: <https://geekpad.ru/the-editor-for-building-block-diagrams-block-diagrams-online-how-to-structurally-provide-information.html>

Обладнання заняття, ТЗН тощо: ноутбук, ПЕОМ.

Завдання студентам на самостійне опрацювання навчального матеріалу, рішення задач, розв'язання вправ для підготовки до наступного лабораторного заняття.

1. Представлення інформації у вигляді структур даних.
2. Розробка програм з застосуванням структур.

Укладач: _____ **Шевчук І.Б., доцент, д.е.н., доцент**
 (підпис) (ПБ, посада, науковий ступінь, вчене звання)