

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені ТАРАСА ШЕВЧЕНКА

В.А.Бородін

Методичні рекомендації з курсу «Мова
програмування C++»

xxxxx
"zzzz"
2021

Методичні рекомендації з курсу «Мова програмування C++» / – К.: КНУ
"Київський університет", 2021. – 101 с.

Рецензенти

кандидат фізико-математичних наук, Семенович К.О

кандидат технічних наук, Знаковська Є. А.

[illegible]

Зміст

1	Лінійні програми на С. Введення/виведення. Дійсний тип даних.	7
	Контрольні запитання:	7
	Завдання для аудиторної роботи:	7
	Завдання для самостійної роботи:	8
	Додаткові задачі:	10
2	Використання математичної бібліотеки С. Створення власних функцій	11
	Контрольні запитання:	11
	Завдання для аудиторної роботи:	11
	Завдання для самостійної роботи:	12
3	Цілі типи С. Умовні конструкції.	14
	Контрольні запитання:	14
	Завдання для аудиторної роботи:	14
	Завдання для самостійної роботи:	15
4	Цикли	18
	Контрольні запитання:	18
	Завдання для аудиторної роботи:	18
	Завдання для самостійної роботи:	19
	Додаткові задачі:	22
5	Цикли. Рекурентні співвідношення. Рекурсія	23
	Контрольні запитання:	23
	Завдання для аудиторної роботи:	23
	Завдання для самостійної роботи:	24
	Додаткові задачі:	27
6	Бітові операції	29
	Контрольні запитання:	29
	Завдання для аудиторної роботи:	29
	Завдання для самостійної роботи:	30
7	Статичні масиви. Лінійні масиви	32
	Контрольні запитання:	32
	Завдання для аудиторної роботи:	32
	Завдання для самостійної роботи:	33
	Додаткові задачі:	35
8	Статичні масиви. Багатовимірні масиви	36
	Контрольні запитання:	36
	Завдання для аудиторної роботи:	36
	Завдання для самостійної роботи:	37
	Додаткові задачі:	39

9	Динамічні масиви. Робота з вказівниками	40
	Контрольні запитання:	40
	Завдання для аудиторної роботи:	40
	Завдання для самостійної роботи:	41
	Додаткові задачі:	43
10	Структури. Створення власного типу	44
	Контрольні запитання:	44
	Завдання для аудиторної роботи:	44
	Завдання для самостійної роботи:	45
	Додаткові задачі:	47
11	Робота з бінарними файлами на C	48
	Контрольні запитання:	48
	Завдання для аудиторної роботи:	48
	Завдання для самостійної роботи:	49
	Додаткові задачі:	52
12	Робота з потоками введення/виведення C++.	54
	Контрольні запитання:	54
	Завдання для аудиторної роботи:	54
	Завдання для самостійної роботи:	55
	Додаткові задачі:	57
13	Робота з класом рядок на C++	59
	Контрольні запитання:	59
	Завдання для аудиторної роботи:	59
	Завдання для самостійної роботи:	60
	Додаткові задачі:	61
14	Робота з текстовими файлами	62
	Контрольні запитання:	62
	Завдання для аудиторної роботи:	62
	Завдання для самостійної роботи:	62
	Додаткові задачі:	65
15	Створення власних класів. Інкапсуляція.	66
	Контрольні запитання:	66
	Завдання для аудиторної роботи:	66
	Завдання для самостійної роботи:	67
	Додаткові задачі:	70
16	Робота з класами. Наслідування та поліморфізм.	71
	Контрольні запитання:	71
	Завдання для аудиторної роботи:	71
	Завдання для самостійної роботи:	72
17	Перетворення типів та робота з виключеннями	77
	Контрольні запитання:	77

Завдання для аудиторної роботи:	77
Завдання для самостійної роботи:	78
18 Створення шаблонів функцій та шаблонів класів	81
Контрольні запитання:	81
Завдання для аудиторної роботи:	81
Завдання для самостійної роботи:	82
19 Стандартна бібліотека C++. Послідовні контейнери та контейнери-адаптори.	85
Контрольні запитання:	85
Завдання для аудиторної роботи:	85
Завдання для самостійної роботи:	86
20 Стандартна бібліотека C++. Асоціативні контейнери.	90
Контрольні запитання:	90
Завдання для аудиторної роботи:	90
Завдання для самостійної роботи:	91
Додаткові задачі:	94
21 Стандартна бібліотека C++. Алгоритми та функтори.	95
Контрольні запитання:	95
Завдання для аудиторної роботи:	95
Завдання для самостійної роботи:	96

ВСТУП

Мета цього посібника – надати студенту завдання для того, щоб практично опанувати потрібні навички програмування на мовах С та С++ в рамках дисципліни «Мова програмування С++». Теми обиралися автором таким чином, щоб якомога більш швидким темпом здобути навички для практичного програмування за відведену кількість занять, тому деякі теми та розділи програмування на С та С++, які автор вважає занадто складними або не обов'язковими з точки зору практики програмування, не входять до цього задачника, а винесені на самостійну роботу або як завдання на курсові проекти.

Автор орієнтувався на стандарти С99 та С++ 98, хоча студенти мають можливість розв'язувати задачі використовуючи інші версії компіляторів мови.

Завдання посібника розподілені на 21 лабораторну роботу, кожна з яких присвячена певній темі, що вивчається в дисципліні. Завдання та теми підбиралися таким чином, щоб вивчення синтаксису мови виходило поступовим, тому розраховується на послідовне виконання лабораторних робіт.

Матеріал кожної лабораторної роботи посібника складається з чотирьох блоків: контрольних запитань, завдань для аудиторної роботи та завдань для самостійної роботи та додаткових задач. Додаткові задачі є задачами підвищеної складності. Під час підготовки до практичного заняття, студент повинен опрацювати блок контрольних запитань та знати відповіді на них. Блок завдань для аудиторної роботи містять перелік типових задач відповідної теми. Ці завдання студент має виконати протягом практичного заняття самостійно або під керівництвом викладача. Завдання для самостійної роботи студент виконує самостійно та звітує про їхнє виконання викладачу. Додаткові задачі можуть бути дані студентам, які хочуть розв'язувати більш складні задачі, що можуть вимагати додаткових знань мови або алгоритмів які не вивчалися в курсі.

Розділ 1

Лінійні програми на С. Введення/виведення. Дійсний тип даних.

Контрольні запитання:

- Як запустити програму на С через консоль? На С++? Як створити проект у вашому улюбленому середовищі?
- Як ініціалізувати дійсне та подвійне дійсні числа в С без попереджень компілятора?
- Як вивести дійсне число в мові С? Як вивести його в десятковому вигляді? З заданою точністю?
- Як ввести дійсне число на С? Як ввести його в експоненційному вигляді? Які розміри дійсних чисел в байтах на С/С++ бувають?
- Як ввести два дійсних числа через пробіли в одному рядку? А якщо роздільник — 2 пробіли? А якщо кома?
- Як ввести два дійсних числа в різних рядках?

Завдання для аудиторної роботи:

1. Обчисліть наступні математичні вирази та виведіть результати:
 $2+31; 45*54-11; 15/4; 15.0/4; 67\%5; (2*45.1 + 3.2)/2;$
2. Ініціалізуйте наступні числа як дійсні, подвійні дійсні та довгі дійсні:
 $10^{-4}, 24.33E5, \pi, e, \sqrt{5}, \ln(100)$ та виведіть їх з 2 знаками після коми.
3. Вивести на екран текст:
а)
- a - a - a
a | a | a
- a - a - a,
де a – введена з клавіатури цифра.
4. Обчислити силу притягання F в науковому (експоненційному) форматі між двома тілами, що мають маси m_1, m_2 на відстані r . Вказівка. Шукана сила визначається за формулою $F = \gamma \frac{m_1 * m_2}{r^2}$, де

Розділ 1. Лінійні програми на С. Введення/виведення. Дійсний тип даних.

$\gamma = 6.673 * 10^{-11} \text{ Н*м}^2/\text{кг}^2$. всі потрібні змінні присвоюються всередині програми. Результат вивести в окремому рядку вигляду «F=***», де замість зірок представлення в науковому (експоненційному) вигляді.

5. Дано дійсне число x . Користуючись лише операцією множення, отримати:

- | | |
|------------------------------|--------------------------------|
| а) x^4 за дві операції; | г) x^{15} за п'ять операцій; |
| б) x^6 за три операції; | д) x^{28} за шість операцій; |
| в) x^9 за чотири операції; | е) x^{64} за шість операцій. |

6. Ввести дійсне число градусів Цельсія C (на екрані повинна бути підказка, що ввести) та обчислити й вивести число F в дійсному форматі – та сама температура в градусах Фаренгейта за формулою $F = \frac{9C}{5} + 32$. Результат вивести в окремому рядку вигляду «F=***», де замість зірок представлення числа в найкоротшому вигляді з можливих.
7. Ввести дійсне число x та підрахуйте без та за допомогою математичних функцій мови С його цілу та дробову частину, найменше ціле число, що більше x та найбільше ціле, що менше x , а також його округлене значення. Перевірте результат роботи для від'ємного числа.
8. Ввести в двох різних рядках послідовно два дійсних числа та обчислити значення їх різниці та добутку. Результат вивести в десятковому представленні (з фіксованою крапкою).
9. Ввести два дійсних числа записаних через пробіли в одному рядку та обчислити значення їх середнього арифметичного та середнього гармонічного. Результат вивести в науковому та десятковому представленні.

Завдання для самостійної роботи:

10. Задайте в програмі довільні 5 цілих та 5 дійсних чисел. Вивести на екран таблицю з цих значень у вигляді, слідкуючі за "красою" виводу:

x		1		2		3		4		5	
-		-		-		-		-		-	
y		3.0		1.0		5.0		4.0		2.1	

11. Ініціалізувати два довільні рядки та вивести їх в одному рядку та поставивши між ними кому та пробіл, а перед та після три окличних знаки. Приклад:

!!! Hello , World! !!!

12. Наближено визначити період обертання Землі навколо Сонця, використовуючи ланцюговий дріб

$$T = 365 + \frac{1}{4 + \frac{1}{7 + \frac{1}{1 + \frac{1}{3}}}}$$

Результат вивести в форматі плаваючої крапки.

13. Обчислити значення функції десяткового логарифму для даного числа та вивести результат з точністю до 3 знаків після десяткової крапки.
14. Тіло починає рухатися без початкової швидкості з прискоренням a . Обчислити: відстань, яку воно пройде за час t від початку руху та час, за який тіло досягне швидкості v .
15. Обчислити кінетичну енергію тіла масою m , що рухається зі швидкістю v відносно поверхні Землі.
16. Вивести на екран таблицю:

x	1	2	3	4	5
F(x)	y	y	y	y	y

де замість символу y - значення у форматі з плаваючою крапкою з точністю до двох знаків після крапки або ціле, вирівняне по центру, функцій:

а) $F(x) = \exp^{-x^2}$; б) $F(x) = \sqrt{x}$.

17. Ввести дійсне число від 0 до 10000 та вивести його 8 ступінь з точністю до 20 знаків до десяткової коми та 4 значками після десяткової коми.
18. Позиція у грі «Хрестики-нулики» представлена в програмі за допомогою 9 символів виду ' ', 'O', 'X'. Показати на екрані позицію у грі «Хрестики-нулики». Наприклад, для позицій { 'O', 'X', ' ', ' ', ' ', 'X', 'O', 'X', 'O', ' ' } вона буде:

O	X
X	O

X | O |

Додаткові задачі:

20. Три дійсні числа вводяться як рядок вигляду:

A=xxx.xxx, B=xxExxx C=xxx.xxxx (тут "A=", "B=", "C=" символи, що повинні бути присутніми на консолі але ігноруються при введенні. Обчисліть їх середнє арифметичне та середнє гармонічне та виведіть у науковому та форматі з фіксованою крапкою.

21. Вивести на екран текст:

- | - | a | - | -

- | a | a | a | -

a | a | a | a | a

де a – введене з клавіатури дійсне число менше ніж 100.

22. Ввести користуючись лише однією функцією вводу ціле число записане в шістнадцятковому вигляді та вивести його зменшене на одиницю в шістнадцятковому та десятковому вигляді.
23. Дійсне число записано в рядку, при цьому перед ним може стояти будь-яка послідовність з пробілів та символів «*». Ввести його користуючись лише одним викликом функції вводу та виведіть значення його кубу.
24. Введіть два натуральних числа n, m та виведіть числа m, m^2 в різних рядках на відстанях від лівого краю консолі рівних n та $2n$ відповідно не користуючись циклами.

Розділ 2

Використання математичної бібліотеки C. Створення власних функцій

Контрольні запитання:

- Як підключити математичні функції та скомпілювати програму, що використовує \sin та \arctan ?
- Як узнати скільки максимальна кількість значущих цифр в даному дійсному типі? Максимальну експоненту та мантису?
- Як записати власну функцію на C? Як запустити її зі сталими аргументами та як з аргументами, що є змінними, в програмі?
- Що таке головна функція (драйвер функція)?
- Як перевірити роботу функції в головній функції якщо ми знаємо її значення в деяких точках?

Завдання для аудиторної роботи:

- 1) Ввести дійсне число x та обчислити значення функції тригонометричного косинуса для нього.
- 2) Обчислити гіпотенузу c прямокутного трикутника за катетами a та b .
- 3) Обчислити площу трикутника S за трьома сторонами a , b , c .
- 4) Напишіть функцію, яка за найменшу кількість арифметичних операцій, обчислює значення многочлена для введеного з клавіатури значення x :

а) $y = x^4 + 2x^2 + 1$;

б) $y = x^4 + x^3 + x^2 + x + 1$;

в) $y = x^5 + 5x^4 + 10x^3 + 10x^2 + 5x + 1$;

г) $y = x^9 + x^3 + 1$;

д) $y = 16x^4 + 8x^3 + 4x^2 + 2x + 1$;

е) $y = x^5 + x^3 + x$.

- 5) Напишіть функцію $Rosenbrock2d(x, y) = 100(x^2 - y)^2 + (x - 1)^2$ та перевірте її результат на довільних трьох парах дійсних чисел.

- 6) Трикутник вводиться координатами своїх вершин, які вводяться так: в першому рядку через пробіл два дійсних числа — координати точки А, далі пропускається рядок, в третьому рядку через пробіл два дійсних числа — координати точки В, знов пропускається рядок, через пробіл координати точки С. Підрахувати площу трикутника. (Вказівка: напишіть функції підрахунку довжини відрізка та функції обчислення площі трикутника за довжинами сторін)

Завдання для самостійної роботи:

- 7) Обчислити площу еліпса за координатами його радіусів.
- 8) В трикутнику відомо довжини всіх сторін. Обчислити довжини його:
- а) медіан,
 - б) бісектрис,
 - в) висот.
- 9) Трикутник заданий величинами своїх кутів та радіусом вписаного кола. Обчисліть його площу.
- 10) Трикутник заданий довжиною своїх сторін. Знайти та вивести величину кутів трикутника у радіанах та градусах.
- 11) Знайти об'єм циліндра, якщо відомо його радіус основи та висоту.
- 12) Знайти об'єм конуса, якщо відомо його радіус основи та висоту.
- 13) Знайти об'єм тора з внутрішнім радіусом r і зовнішнім радіусом R .
- 14) Знайти корені квадратного рівняння з коефіцієнтами a, b, c , якщо відомо, що обидва корені в ньому існують. Перевірте ваш розв'язок на коефіцієнтах рівняння $a = 3, b = 100, c = 2$.
- 15) Скласти функцію для обчислення значення многочлена від двох змінних для введеної з клавіатури пари чисел (x, y) :
- а) $f(x, y) = x^3 + 3x^2y + 3xy^2 + y^3$;
 - б) $f(x, y) = x^2y^2 + x^3y^3 + x^4y^4$;
 - в) $f(x, y) = x + y + x^2 + y^2 + x^3 + y^3 + x^4 + y^4$.
- 16) Обчислити відстань від точки (x_0, y_0) до:
- а) заданої точки (x, y) ;
 - б) заданої прямої $ax + by + c = 0$;

в) точки перетину прямих $x + by + c = 0$ і $ax + y + c = 0$, де $ab \neq 1$.

- 17) Напишіть власні функції, що обчислюють наступні вирази та відповідні власні функції, що будуть рахувати похідні даних функцій (Приклад, функція $f(x) = \text{identity}(x) = x$, її похідна $g(x) = \text{identity_derivative}(x) = 1$) :

а) $f(x) = \text{th}(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$;

б) $f(x) = \text{bent}(x) = \frac{\sqrt{x^2 + 1} - 1}{2} + x$;

в) $f(x) = \text{softSign}(x) = \frac{x}{1 + |x|}$;

г) $f(x) = \text{arctg}(x) = \text{tg}^{-1}(x)$;

д) $f(x) = \text{gauss}(x) = e^{-x^2}$;

е) $f(x) = \text{softPlus}(x) = \ln(1 + e^x)$;

є) $f(x) = \text{sigmoid}(x) = (1 + e^{-x})^{-1}$;

ж) $f(x) = \text{invsqrt}(x, \alpha) = \frac{x}{\sqrt{1 + \alpha x^2}}$;

з) $f(x) = \text{sigmweight}(x) = x * (1 + e^{-x})^{-1}$.

Розділ 3

Цілі типи C. Умовні конструкції.

Контрольні запитання:

- Які типи цілих чисел використовуються в C/C++?
- Які варіанти використання булевого типу є в C?
- Як перевести число із знакового до беззнакового типу? Як навпаки?
- Як ввести найдовше можливе ціле число? Як узнати його розмір в байтах?
- Як з'ясовує скільки байтів на цілий та довгий цілий тип виділяє компілятор, а також чи підтримує він довгий тип та скільки на нього виділяється байтів?
- Як виконати цілочисельне ділення в C? Як поділити не цілочисельно два цілих числа?
- Як коректно та без попереджень компілятора ініціалізувати довге натуральне число? Натуральне коротке? Ціле довге?
- Як коректно та без попереджень ввести та вивести натуральне число? Натуральне коротке? Ціле довге?
- Як записати умовне розгалуження в C/C++?
- Які типи умовних виразів на C/C++? Напишіть два варіанти з ними для пошуку мінімуму двох чисел. Напишіть за допомогою виразу альтернативи функцію, що повертає парність цілого числа.

Завдання для аудиторної роботи:

- 1) Дано натуральне тризначне число. Знайти:
 - а) кількість одиниць, десятків і сотень цього числа;
 - б) суму цифр цього числа;
 - в) число, утворене при читанні заданого числа справа наліво.
- 2) Ввести натуральне тризначне число. Якщо в ньому всі 3 цифри різні, то вивести всі числа, які утворюються при перестановці цифр заданого числа.

- 3) Введіть три цілих числа, записаних через кому в одному рядку та підрахуйте їх добуток якщо всі ці числа гарантовано по модулю менші а) 2^{10} ; б) 2^{21} .
- 4) Напишіть функцію, що гарантовано приймає у якості аргументів 8-бітні натуральні числа та обчислює їх добуток як гарантовано 16-бітне натуральне число.
- 5) Визначити більше та менше з двох чисел, введених з клавіатури.
- 6) Дано три дійсних числа. Скласти програму для знаходження числа: найбільшого за модулем та найменшого за модулем.
- 7) Визначити, скільки розв'язків має рівняння та розв'язати його:

а) $ax^2 + bx + c = 0$;

б) $ax^4 + bx^2 + c = 0$.

Завдання для самостійної роботи:

- 8) Введіть два натуральних 32-бітних числа та виведіть їх суму як 32-бітне число, якщо немає переповнення типу. В противному випадку виведіть про це повідомлення. Аналогічно підрахуйте добуток двох цілих 32-бітних чисел.
- 9) Дано три дійсних числа x , y і z . Скласти програму для обчислення:
- а) $\max(x + y + z, xy - xz + yz, xyz)$;
- б) $\max(xy, xz, yz)$.
- 10) Дано три дійсних числа x , y і z . Визначити кількість:
- а) різних серед них;
- б) однакових серед них;
- в) чисел, що є більшими за їхнє середнє арифметичне значення;
- г) чисел, що є більшими за введене з клавіатури число a .
- 11) Обчислити значення функцій:
- а) $f(x) = |x|$;
- б) $f(x) = ||x| - 1| - 1$;
- в) $f(x) = \text{sign}(x)$
- г) $f(x) = \sin(|x|)$;

12) Перевірити, чи існує трикутник із заданими сторонами a, b, c . Якщо так, то визначити, який він: гострокутний, прямокутний чи тупокутний.

13) Визначити, скільки розв'язків має система рівнянь і розв'язати її:

$$\text{а) } \begin{cases} a_1x + b_1y + c_1 = 0 \\ a_2x + b_2y + c_2 = 0; \end{cases} \quad \text{б) } \begin{cases} |x + y| = 1 \\ ax + by + c = 0 \end{cases}$$

14) Знайти число точок перетину кола $x^2 + y^2 = r^2$ з відрізком $x = a, b \leq y \leq b + c^2$.

15) Скласти програму, яка по колу $(x - v)^2 + (y - u)^2 = r^2$ та прямій $ax + by + c = 0$ встановлює, який випадок має місце:

- дві точки перетину;
- одна точка дотику;
- жодної спільної точки.

16) З'ясувати, чи перетинаються два кола на площині.

17) Задано два квадрати, сторони яких паралельні координатним осям. З'ясувати, чи перетинаються вони. Якщо так, то знайти координати лівого нижнього та правого верхнього кутів прямокутника, що є їхнім перетином.

18) Дано два прямокутники, сторони яких паралельні координатним осям. Відомо координати лівого нижнього та правого верхнього кутів кожного з прямокутників. Знайти координати лівого нижнього та правого верхнього кутів мінімального прямокутника, що містить задані прямокутники.

19) Записати функції, що повертають значення 1 тоді й тільки тоді, коли:

- а) натуральне число n – непарне;
- б) остання цифра числа $n - 5$;
- в) ціле число n кратно натуральному числу m ;
- г) натуральні числа n і k одночасно кратні натуральному числу m
- д) сума першої і другої цифри двозначного натурального числа - двозначне число;
- е) число x більше за число y не менше, ніж на 7;

- є) принаймні одне з чисел x , y або z більше за 99;
- ж) тільки одне з чисел x , y або z менше за 1001.
- 20) Реалізувати функцію, яка перевіряє, чи належить дана своїми координатами точка трикутнику, що заданий координатами своїх вершин.
- 21) Точка простору задана декартовими координатами (x, y, z) . Перевірити, чи належить вона кулі з радіусом R і центром у початку координат.
- 22) Точка простору задана декартовими координатами (x, y, z) . Перевірити, чи належить вона циліндру, вісь якого збігається з віссю Oz . Висота дорівнює h , а нижня основа лежить у площині Oxy та має радіус r .
- 23) Реалізуйте функції та напишіть відповідну до кожної з них функцію, що буде рахувати їх похідні (за нескінченість прийміть найбільше можливе число типу double):

$$a) \text{onestep}(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

$$б) \text{ReLU}(x) = \max(0, x)$$

$$в) \text{leakyReLU}(x, a) = \begin{cases} ax, & x < 0 \\ 0, & x \geq 0 \end{cases}$$

$$г) \text{eReLU}(a, x) = \begin{cases} a(e^x - 1), & x < 0 \\ 0, & x \geq 0 \end{cases}$$

$$д) \text{sReLU}(tl, tr, al, ar, x) = \begin{cases} tl + al(x - tl), & x \leq tl \\ 0, & tl < x < tr \\ tr + ar(x - tr), & x \geq tr \end{cases}$$

$$е) \text{isReLU}(a, x) = \begin{cases} \frac{x}{\sqrt{1+ax^2}}, & x < 0 \\ x, & x \geq 0 \end{cases}$$

$$є) \text{softExponential}(a, x) = \begin{cases} -\frac{\ln(1-a(x+a))}{a}, & a < 0 \\ x, & a = 0 \\ \frac{e^{ax}-1}{a} + a, & a > 0 \end{cases}$$

$$ж) \text{sinc}(x) = \begin{cases} 1, & x = 0 \\ \frac{\sin x}{x}, & x \neq 0 \end{cases}$$

Розділ 4

Цикли

Контрольні запитання:

- Які типи циклів на C/C++? Напишіть цикл для введення n цілих чисел за допомогою трьох різних типів циклів.
- Напишіть цикл для введення дійсних чисел доки не введемо 0. Обчисліть максимум з цих чисел.
- Які інструкції та команди дозволяють завершити (перервати цикл)?
- Як можна уникнути виконання однієї (чи декількох) ітерацій циклу?
- Як обчислити факторіал за допомогою арифметичного циклу на C?

Завдання для аудиторної роботи:

- 1) Скласти функцію обчислення за даним дійсним x та натуральним n число $y = \sin(\sin(\dots \sin(x) \dots))$ (n разів).

- 2) Вивести на екран такий рядок:

$$n! = 1*2*3*4*5*...*n,$$

де n – введене з клавіатури натуральне число, використовуючи

- цикл по діапазону із зростанням;
- цикл по діапазону зі спаданням.

- 3) Скласти функції для обчислення значень многочленів і виконати їх при заданих значеннях аргументів:

а) $y = x^n + x^{n-1} + \dots + x^2 + x + 1, \quad n = 3, x = 2;$

б) $y = x^{2^n} y^n + x^{2^{n-1}} y^{n-1} + \dots + x^2 y + 1, \quad n = 4, x = 1, y = 2;.$

- 4) Дано натуральне число n . Написати програми обчислення значення виразу при заданому значенні x :

$$x + 2x^2 + \dots + (n-1)x^{n-1} + nx^n.$$

- 5) Скласти функцію обчислення подвійного факторіала натурального числа n : $y = n!!$.

Вказівка. За означенням

$$n!! = \begin{cases} 1 \cdot 3 \cdot 5 \cdot \dots \cdot n, & \text{при } n \text{ не парному,} \\ 2 \cdot 4 \cdot 6 \cdot \dots \cdot n, & \text{при } n \text{ парному} \end{cases}$$

6) Скласти програму обчислення

а) $\sqrt{2 + \sqrt{2 + \dots + \sqrt{2}}} \text{ (} n \text{ коренів),}$

б) $\sqrt{3 + \sqrt{6 + \dots + \sqrt{3(n-1) + \sqrt{3n}}}}$

7) Скласти програми обчислення значень многочлену для $x \in \mathbf{R}$, що по модулю менше за одиницю та $n \geq 0$:

$$y = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}.$$

8) Для довільного цілого числа $m \geq 1$ знайти найбільше ціле k , при якому $4^k \leq m$.

9) Для заданого натурального числа n одержати найменше число вигляду 2^r , яке перевищує n .

10) Знайдіть машинний нуль для дійсного типу float вашого компілятора, тобто таке число типу float $a > 0$, що $1 + a = 1$ буде істиною. Як перевірити це значення, користуючись стандартною бібліотекою?

Вказівка: в циклі ділити значення a на 2 доки не виконується вказана вище рівність.

11) Ввести послідовність наступним чином: користувачу виводиться напис "a[**]=", де замість ** стоїть номер числа, що вводиться. Тобто там виводяться написи "a[0]=", і після знаку рівності користувач вводить число, "a[1]=", і після знаку рівності користувач вводить число і так далі доки користувач не введе число 0. Після цього потрібно вивести суму введених чисел (масив чисел заводити необов'язково). Введіть послідовність цілих ненульових чисел (тобто введення закінчується коли ми вводимо 0) та виведіть середнє арифметичне введених чисел та середнє геометричне.

Завдання для самостійної роботи:

12) Скласти функції для обчислення значень многочленів і виконати їх при заданих значеннях аргументів:

а) $y = x^{2^n} + x^{2^{n-1}} + \dots + x^4 + x^2 + 1;$

$$\text{б) } y = x^{3^n} + x^{3^{n-1}} + \dots + x^9 + x^3 + 1;$$

$$\text{в) } y = x^{1^2} + x^{2^2} + \dots + x^{n^2}.$$

- 13) Дано натуральне число n . Написати програми обчислення значень виразів при заданому значенні x :

$$\text{а) } 1 + (x - 1) + (x - 1)^2 + \dots + (x - 1)^n;$$

$$\text{б) } 1 + \frac{1}{x^2+1} + \frac{1}{(x^2+1)^2} + \dots + \frac{1}{(x^2+1)^n};$$

$$\text{в) } 1 + \sin x + \sin^2 x + \dots + \sin^n x;$$

$$\text{г) } y = nx^{n-1} + (n-1)x^{n-2} + \dots + 2x + 1;$$

$$\text{д) } y = \sum_{k=0}^n kx^k(1-x)^{n-k}, \text{ за умови } 0 < x < 1, n \geq 0).$$

- 14) Введіть послідовність цілих ненульових чисел (тобто введення закінчується коли ми вводимо 0). Визначити кількість змін знаку в цій послідовності. Наприклад, у послідовності 1, -34, 8, 14, -5, 0 знак змінюється три рази.

- 15) Введіть послідовність натуральних ненульових чисел (тобто введення закінчується коли ми вводимо 0). Визначити порядковий номер найменшого з них.

- 16) Введіть послідовність дійсних ненульових чисел (тобто введення завершується коли ми вводимо 0). Визначити величину найбільшого серед від'ємних членів цієї послідовності. Якщо від'ємних чисел немає, то вивести найменший серед додатних членів.

- 17) Банк пропонує річну ставку по депозиту А та В гривень по вкладу додаються до основної суми депозиту кожен рік. Ви кладете в цей банк D гривень. Скільки років потрібно чекати, щоб сума вкладу зросла до очікуваної суми Р?

- 18) Скласти програми для обчислення елементів послідовностей. Бібліотеку `<math.h>` не використовувати.

$$\text{а) } x_k = \frac{x^k}{k} \quad (k \geq 1)$$

$$\text{д) } x_k = \frac{x^k}{k!} \quad (k \geq 0);$$

$$\text{б) } x_k = \frac{x^{2k}}{(2k)!} \quad (k \geq 0);$$

$$\text{е) } x_k = \frac{(-1)^k x^{2k}}{(2k)!} \quad (k \geq 0);$$

$$\text{в) } x_k = \frac{(-1)^k x^k}{k} \quad (k \geq 1);$$

$$\text{є) } x_k = \frac{(-1)^k x^k}{k!} \quad (k \geq 0);$$

$$\text{г) } x_k = \frac{x^{2k+1}}{(2k+1)!} \quad (k \geq 0);$$

$$\text{ж) } x_k = \frac{(-1)^k x^{2k+1}}{(2k+1)!} \quad (k \geq 0).$$

19) Задане натуральне число n . Скласти програми обчислення добутків:

а) $p = \left(1 + \frac{1}{1^2}\right) \left(1 + \frac{1}{2^2}\right) \dots \left(1 + \frac{1}{n^2}\right), \quad n > 2;$

б) $p = \left(1 - \frac{1}{2^2}\right) \left(1 - \frac{1}{3^2}\right) \dots \left(1 + \frac{1}{n^2}\right), \quad n > 2.$

20) Скласти програму друку таблиці значень функції $y = e^{-dx^2}$ для d на відрізьку $[-3, 3]$ з заданим кроком $h > 0$.

21) Скласти програму визначення кількості тризначних натуральних чисел, сума цифр яких дорівнює $n \geq 1$. Операцію ділення не використовувати.

22) Дано n цілих чисел. Скласти програму, що визначає, скільки з них більші за своїх "сусідів" тобто попереднього та наступного чисел.

23) Задані натуральне число n , дійсні числа y_1, \dots, y_n . Скласти програму визначення:

а) $\max(|z_1|, \dots, |z_n|)$, де $z_i = \begin{cases} y_i, & \text{при } |y_i| \leq 2, \\ 0.5, & \text{у інших випадках} \end{cases};$

б) $\min(|z_1|, \dots, |z_n|)$, де $z_i = \begin{cases} y_i, & \text{при } |y_i| \geq 1, \\ 2, & \text{у інших випадках} \end{cases};$

в) $z_1 + z_2 + \dots + z_n$, де $z_i = \begin{cases} y_i, & \text{при } y_i < 10, \\ 1, & \text{у інших випадках} \end{cases}$

24) Дано натуральне число n . Викинути із десяткового запису числа n цифри 0 і 5, залишивши порядок інших цифр. Наприклад, з числа 59025509 повинно вийти 929.

25) Знайти період десяткового дробу для відношення n/m для заданих натуральних чисел n та m .

26) Скоротити дріб n/m для заданих цілого числа n та натурального числа m .

27) Задана точність $\varepsilon > 0$. Обчислити число π з цією точністю за допомогою наступних формул:

а) $\frac{\pi^2}{6} = \sum_{k=0}^{\infty} \frac{1}{k^2};$

б) $\frac{\pi}{2} = \frac{2}{1} \cdot \frac{2}{2} \cdot \frac{4}{3} \cdot \frac{4}{5} \cdot \frac{6}{5} \cdot \frac{6}{7} \dots;$

в) $\frac{2}{\pi} = \frac{\sqrt{2}}{2} \cdot \frac{\sqrt{2+\sqrt{2}}}{2} \cdot \frac{\sqrt{2+\sqrt{2+\sqrt{2}}}}{2} \dots;$

$$\text{г)} \pi = \sum_{k=0}^{\infty} \frac{(-1)^k}{4^k} \left(\frac{2}{4k+1} + \frac{2}{4k+2} + \frac{1}{4k+2} \right);$$

$$\text{д)} \frac{\pi^4}{90} = \sum_{k=0}^{\infty} \frac{1}{k^4};$$

$$\text{е)} \pi = 2\sqrt{3} \sum_{k=0}^{\infty} \frac{(-1)^k}{3^k(2k+1)};$$

$$\text{є)} 2\pi = \sum_{k=0}^{\infty} \frac{(-1)^k}{16^k} \left(\frac{8}{8k+2} + \frac{4}{8k+3} + \frac{4}{8k+4} - \frac{1}{8k+7} \right);$$

$$\text{ж)} \pi = 8 \sum_{k=1}^{\infty} \sum_{m=1}^{\infty} \frac{1}{(4m-2)^k};$$

$$\text{з)} \frac{1}{\pi} = \frac{2\sqrt{2}}{9801} \sum_{k=0}^{\infty} \frac{(4k)!(1103+26390k)}{(k!)^4 396^{4k}}.$$

Додаткові задачі:

- 27) Ввести натуральні числа a і b та натуральне число n . Чи можна представити число n у вигляді $n = k * a + m * b$, де k та m – натуральні числа? Якщо можна – то знайдіть такі числа k та m , що мають найменшу суму модулів.
- 28) Представити дане натуральне число як суму двох квадратів натуральних чисел. Якщо це неможливо представити як суму трьох квадратів. Якщо і це неможливо, представити у вигляді суми чотирьох квадратів натуральних чисел.
- 29) Знайти всі цілі корені кубічного рівняння $ax^3 + bx^2 + cx + d = 0$ (a, b, c, d – задані цілі числа). *Вказівка:* цілі корені повинні бути від'ємними або додатними дільниками вільного члену d .
- 30) Напишіть функцію, яка розраховує для даного натурального числа n значення функції Ойлера — тобто кількість чисел від 1 до n , взаємно простих з n .
- 31) Ввести натуральне число $d > 1$ та натуральне число m . Знайдіть мінімальну кількість натуральних чисел вигляду x^d (d -степенів натуральних чисел) сума яких дорівнює m .

Розділ 5

Цикли. Рекурентні співвідношення. Рекурсія

Контрольні запитання:

- Яким чином обчислити числа Фібоначчі на C за допомогою циклів?
- Який загальний метод обчислення рекурентних послідовностей для C?
- Що таке рекурсія та як її застосувати для обчислення, наприклад, факторіалу? Чисел Фібоначчі?
- Що таке бінарний пошук та як його застосувати?

Завдання для аудиторної роботи:

1) Маємо дійсне число a . Скласти програми обчислення:

- а) серед чисел $1, 1 + \frac{1}{2}, 1 + \frac{1}{2} + \frac{1}{3}, \dots$ першого, більшого за задане число a ;
- б) такого найменшого $n > 0$, що $1 + \frac{1}{2} + \dots + \frac{1}{n} > a$.

2) Числами Фібоначчі називається числова послідовність $\{F_n\}$, задана рекурентним співвідношенням другого порядку $F_0 = 0, F_1 = 1, F_k = F_{k-1} + F_{k-2}, k = 2, 3, \dots$

Скласти функції:

- а) для обчислення F_n за номером члену;
 - б) номера найбільшого числа Фібоначчі, яке не перевищує задане число a ;
 - в) номера найменшого числа Фібоначчі, яке більше заданого a ;
 - г) суми всіх чисел Фібоначчі, які не перевищують 1000.
- 3) Введіть натуральне число n . Далі утворити рекурентну послідовність $\{a_i\}$ за наступним правилом: $a_0 = n$. Якщо a_k парне, то $a_{k+1} = a_k/2$, якщо a_k – непарне, то $a_{k+1} = 4a_k + 1$. Доведіть що для $n < 1000$ ця послідовність буде збігатись до одиниці. Знайдіть серед цих n число, якому потрібна максимальна кількість кроків для досягнення одиниці.
- 4) Скласти програми для обчислення добутків:

$$\text{а) } P_n = \prod_{i=1}^n \left(1 + \frac{1}{i!}\right);$$

$$\text{б) } P_n = \prod_{i=1}^n \left(1 + \frac{(-1)^k}{i^2}\right);$$

Вказівка. Добуток P_n обчислити за допомогою рекурентного співвідношення $P_0 = 1, P_k = P_{k-1} * a_k, k = 1, 2, \dots, n, (k = 1, 2, \dots, n)$ де a_k - k -тий множник.

- 5) Скласти програми для обчислення найменшого додатного члена числових послідовностей, які задаються рекурентними співвідношеннями, та його номера: $x_n = x_{n-1} + x_{n-3} + 100, x_1 = x_2 = x_3 = -99, n = 3, 4, \dots;$

- 6) Скласти програми для обчислення ланцюгових дробів

$$\text{а) } b_n = b + \frac{1}{b + \frac{1}{b + \frac{1}{b}}};$$

$$\text{б) } \lambda_n = 2 + \frac{1}{6 + \frac{1}{10 + \frac{1}{4n+2}}};$$

$$\text{в) } x_{2n} = 1 + \frac{1}{2 + \frac{1}{1 + \frac{1}{2 + \frac{1}{1 + \frac{1}{2}}}}};$$

Вказівка. Використати рекурентні співвідношення

$$\text{а) } b_0 = b, b_k = b + \frac{1}{b_{k-1}}, k = 1, 2, \dots, n;$$

$$\text{б) } b_0 = 4n + 2, b_k = 4(n - k) + 2 + \frac{1}{b_{k-1}}, k = 1, 2, \dots, n.$$

- 7) Скласти програми для обчислення суми:

$$S_n = \sum_{k=1}^n \frac{2^k}{a_k + b_k},$$

$$\text{де } \begin{cases} a_1 = 0, a_2 = 1, \\ a_k = \frac{a_{k-1}}{k} + a_{k-2}b_k, \end{cases} \begin{cases} b_1 = 1, b_2 = 0, \\ b_k = b_{k-1} + a_{k-1}, \end{cases} k = 3, 4, \dots;$$

Завдання для самостійної роботи:

- 8) Скласти програми обчислення довільного елемента послідовностей, заданих рекурентними співвідношеннями

$$\text{а) } v_0 = 1, v_1 = 0.3, v_i = (i + 2)v_{i-2}, i = 2, 3, \dots$$

$$\text{б) } v_0 = v_1 = v_2 = 1, v_i = (i+4)(v_{i-1} - 1) + (i+5)v_{i-3}, i = 3, 4, \dots$$

$$\text{в) } v_0 = v_1 = 0, v_2 = \frac{3}{2}, v_i = \frac{i-2}{(i-3)^2+1}v_{i-1} - v_{i-2}v_{i-3} + 1, i = 2, 3, \dots$$

- 9) Скласти програму обчислення довільного елемента послідовності v_n , визначеної системою співвідношень

$$v_0 = v_1 = 1, v_i = \frac{u_{i-1} - v_{i-1}}{|u_{i-2} + v_{i-1}| + 2}, i = 2, 3, \dots;$$

$$\text{де } u_0 = u_1 = 0, u_i = \frac{u_{i-1} - u_{i-2}v_{i-1} - v_{i-2}}{1 + u_{i-1}^2 + v_{i-1}^2}, i = 2, 3, \dots;$$

- 10) Скласти програми для обчислення сум:

$$\text{а) } S_n = \sum_{k=1}^n 2^k a_k, \text{ де } a_1 = 0, a_2 = 1, a_k = a_{k-1} + k * a_{k-2}, k = 3, 4, \dots;$$

$$\text{б) } S_n = \sum_{k=1}^n \frac{3^k}{a_k}, \text{ де } a_1 = a_2 = 1, a_k = \frac{a_{k-1}}{k} + a_{k-2}, k = 3, 4, \dots;$$

$$\text{в) } S_n = \sum_{k=1}^n \frac{k!}{a_k}, \text{ де } a_1 = a_2 = 1, a_k = a_{k-1} + \frac{a_{k-1}}{2^k}, k = 3, 4, \dots;$$

$$\text{г) } S_n = \sum_{k=1}^n k! a_k, \text{ де } a_1 = 0, a_2 = 1, a_k = a_{k-1} + \frac{a_{k-2}}{(k-1)!}, k = 3, 4, \dots;$$

$$\text{д) } S_n = \sum_{k=1}^n \frac{a_k}{2^k}, \text{ де } a_1 = a_2 = a_3 = 1, a_k = a_{k-1} + a_{k-3}, k = 4, 5, \dots;$$

$$\text{е) } S_n = \sum_{k=1}^n \frac{2^k}{k!} a_k, \text{ де } a_0 = 1, a_k = k a_{k-1} + \frac{1}{k}, k = 1, 2, \dots$$

- 11) Скласти програми для обчислення сум:

$$\text{а) } S_n = \sum_{k=1}^n \frac{3^{2k+1}}{a_k * b_{k+1}},$$

$$\text{де } \begin{cases} a_1 = 2, a_2 = 1, \\ a_k = \frac{a_k}{k+1} + a_{k-2} + b_k, \end{cases} \begin{cases} b_1 = 1, b_2 = 0, \\ b_k = 2b_{k-1} + a_{k-1}, \end{cases} k = 3, 4, \dots;$$

$$\text{б) } S_n = \sum_{k=1}^n \frac{a_k b_k}{(k+1)!},$$

$$\text{де } \begin{cases} a_1 = u, \\ a_k = 2b_{k-1} + a_{k-1}, \end{cases} \begin{cases} b_1 = v, \\ b_k = 2a_{k-1}^2 + b_{k-1}, \end{cases} k = 2, 3, \dots;$$

u, v – задані дійсні числа;

$$в) S_n = \sum_{k=1}^n \frac{2^k}{(1+a_k+b_k)k!}$$

$$\text{де } \begin{cases} a_1 = 1, \\ a_k = 3b_{k-1} + 2a_{k-1}, \end{cases} \quad \begin{cases} b_1 = 1, \\ b_k = 2a_{k-1} + b_{k-1}, \end{cases} \quad k = 2, 3, \dots;$$

$$г) S_n = \sum_{k=1}^n \left(\frac{a_k}{b_k} \right)^k,$$

$$\text{де } \begin{cases} a_0 = 1, a_1 = 2, \\ a_k = b_{k-2} + \frac{b_k}{2}, \end{cases} \quad \begin{cases} a_0 = 5, b_1 = 5, \\ b_k = b_{k-2}^2 - a_{k-1}, \end{cases} \quad k = 2, 3, \dots;$$

$$д) S_n = \sum_{k=1}^n \frac{a_k}{1+b_k},$$

$$\text{де } \begin{cases} a_0 = 1, \\ a_k = b_{k-1}a_{k-1}, \end{cases} \quad \begin{cases} b_0 = 1, \\ b_k = b_{k-1} + a_{k-1}, \end{cases} \quad k = 1, 2, \dots$$

12) Скласти програми для обчислення добутків

$$а) P_n = \prod_{k=0}^n \frac{a_k}{3^k}, \text{ де } \begin{cases} a_0 = a_1 = 1, a_2 = 3, \\ a_k = a_{k-3} + \frac{a_{k-2}}{2^{k-1}}, \end{cases} \quad k = 3, 4, \dots;$$

$$б) P_n = \prod_{k=1}^n a_k b_k,$$

$$\text{де } \begin{cases} a_1 = 1, \\ a_k = (\sqrt{b_{k-1}} + a_{k-1}) / 5, \end{cases} \quad \begin{cases} b_1 = 1, \\ b_k = 2b_{k-1} + 5a_{k-1}^2, \end{cases} \quad k = 2, 3, \dots$$

13) Реалізувати функцію яка з'ясовує, чи входить задана цифра до запису заданого натурального числа.

14) Реалізувати функцію "інверсії"(запису числа в зворотньому порядку цифр) заданого натурального числа.

Вказівка. Для побудови числа використати рекурентне співвідношення $y_0 = 0, y_i = y_{i-1} * 10 + a_i$, де a_i – наступна цифра числа n при розгляді цифр справа наліво.

15) Скласти програми наближеного обчислення суми всіх доданків, абсолютна величина яких не менше $\varepsilon > 0$:

$$а) y = \sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots;$$

- б) $y = \cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots;$
- в) $y = \sinh(x) = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \dots;$
- г) $y = \cosh(x) = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \dots;$
- д) $y = e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots;$
- е) $y = \ln(1+x) = x - \frac{x^2}{2!} + \frac{x^3}{3!} - \dots, (|x| < 1);$
- є) $y = \frac{1}{1+x} = 1 - x + x^2 - x^3 + \dots, (|x| < 1);$
- ж) $y = \ln \frac{1+x}{1-x} = 2 * \frac{x}{1} + \frac{x^3}{3} + \frac{x^5}{5} + \dots, (|x| < 1);$
- з) $y = \frac{1}{(1+x)^2} = 1 - 2 * x + 3 * x^2 - \dots, (|x| < 1);$
- і) $y = \frac{1}{(1+x)^3} = 1 - \frac{2*3}{2}x + \frac{3*4}{2}x^2 - \frac{4*5}{2}x^3 + \dots, (|x| < 1);$
- й) $y = \frac{1}{1+x^2} = 1 - x^2 + x^4 - x^6 + \dots, (|x| < 1);$
- к) $y = \sqrt{1+x} = 1 + \frac{1}{2}x - \frac{1}{2*4}x^2 + \frac{1*3}{2*4*6}x^3 - \dots, (|x| < 1);$
- л) $y = \frac{1}{\sqrt{1+x}} = 1 - \frac{1}{2}x + \frac{1*3}{2*4}x^2 - \frac{1*3*5}{2*4*6}x^3 - \dots, (|x| < 1);$
- м) $y = \arcsin(x) = x + \frac{1}{2} \frac{x^3}{3!} + \frac{1*3}{2*4} \frac{x^5}{5!} + \dots, (|x| < 1).$

Вказівка. Суму y обчислювати за допомогою рекурентного співвідношення $S_0 = 0$, $S_k = S_{k-1} + a_k$, $k = 1, 2, \dots$, де a_k – k -тий доданок, для обчислення якого також складається рекурентне співвідношення. В якості умови повторення циклу розглядається умова $|a_k| \geq \varepsilon$.

- 16) Ввести дійсні числа $x, \varepsilon (x \neq 0, \varepsilon > 0)$. Обчислити з точністю ε нескінченну суму і вказати кількість врахованих доданків.

а) $\sum_{k=0}^{\infty} \frac{x^{2k}}{2^k k!};$	в) $\sum_{k=0}^{\infty} \frac{x^{2k}}{2^k k!};$
б) $\sum_{k=0}^{\infty} \frac{(-1)^k x^k}{(k+1)^2};$	г) $\sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{k!(2k+1)!}.$

Додаткові задачі:

- 17) Дано натуральне число k . Скласти програму одержання k -тої цифри послідовності

а) 110100100010000 ..., в якій виписані підряд ступені числа 10;

- б) 123456789101112 ..., в якій виписані підряд всі натуральні числа;
- в) 149162536 ..., в якій виписані підряд квадрати всіх натуральних чисел;
- г) 01123581321 ..., в якій виписані підряд всі числа Фібоначчі.
- 18) Скласти програму знаходження кореня рівняння $tgx = x$ на відрізку $[0,001;1,5]$ із заданою точністю ε , використовуючи метод ділення відрізка навпіл.
- 19) Знайти корінь рівняння $x^3 + 4x^2 + x - 6 = 0$, який міститься на відрізку $[0,2]$, з заданою точністю ε .

Вказівка. Одним з методів розв'язування рівняння є метод хорд, який полягає в обчисленні елементів послідовності

$$u_0 = \frac{a+b}{2}, \quad u_n = u_{n-1} - \frac{y(u_{n-1})(u_{n-1}-u_0)}{y(u_{n-1})-y(u_0)}$$

до виконання умови $|u_n - u_{n-1}| < \varepsilon$. В умовах нашої задачі $a = 0, b = 2, y(x) = x^3 + 4x^2 + x - 6$.

- 20) а) Скласти програму, яка визначає потрібний спосіб розміну будь-якої суми грошей до 99 коп. за допомогою монет вартістю 1, 2, 5, 10, 25, 50 коп. Нехай в нашому гаманці відома кількість монет кожної вартості. Чи можемо ми розмінати дану суму?
- б) Розв'яжіть попередній пункт задачі для будь-якого натурального числа m ($1 < m < 10^8$) копійок так щоб загальна кількість монет при цьому була найменша.
- 21) Вирахуйте число π за допомогою наближення $\frac{(a_k+b_k)^2}{4t_k}$, що задаються послідовностями a_k, b_k, t_k, p_k , де $a_0 = 1, b_0 = \frac{1}{\sqrt{2}}, t_0 = \frac{1}{4}, p_0 = 1$ та $a_{k+1} = \frac{a_k+b_k}{2}, b_{k+1} = \sqrt{a_k \cdot b_k}, t_{k+1} = t_k - p_k(a_k - a_{k+1}), p_{k+1} = 2p_k$.

Розділ 6

Бітові операції

Контрольні запитання:

- Що таке та які бітові операції існують? Який пріоритет цих операцій?
- Чому дорівнюють наступні вирази: $3 \ll 2$, $5 \gg 2$, $5 \& 3$, $n \& 1$, $n | 1$, $n \wedge n$, ~ 0 .
- Як знайти значення самого лівого (найстаршого) біту? Самого правого (наймолодшого)? Третього зліва? Як встановити 5-й біт зліва в 1? В нуль?
- Для яких типів C краще не застосовувати бітові операції?
- Який нюанс для найстаршого біту числа є при використанні бітових операцій для цілого типу C/C++?

Завдання для аудиторної роботи:

- 1) Ввести натуральне 8-бітне число $n < 64$ і вивести 2^n , використовуючи бітові операції.
- 2) Ввести ціле число n та натуральне k і вивести ціле число, яке у якого k -тий біт встановлений в 1, а всі інші біти збігаються з бітами числа n на тих же позиціях. Наприклад, якщо введені 9 і 1, відповіддю буде 11.
- 3) Ввести натуральне 64-бітне число m . Встановіть j -ий біт з номером j рівним значенню нуль та виведіть отримане число в десятковому та шістнадцятковому вигляді.
- 4) Поміняйте місцями перші 8 бітів та останні 8 бітів натурального числа, виведіть отримане число в десятковому та шістнадцятковому вигляді.
- 5) Підрахуйте найбільшу кількість одиниць серед бітів даного числа, що йдуть підряд.
- 6) Описати словами результат наступних виразів: $x \& (x-1)$, $x \& (-x)$. Як можна програмно перевірити ці твердження?
- 7) Напишіть функцію що визначає до якої архітектури (big(high), little, middle endian) належить даний комп'ютер.

Завдання для самостійної роботи:

- 8) Ввести натуральне(32-бітне) число m . Встановіть її j -тий рівним нулеві та виведіть отримане число в десятковому та шістнадцятковому вигляді.
- 9) Визначить номер першого значущого зліва та справа біта натурального числа m .
- 10) Поміняйте місцями перші 8 біт та останні 8 біт натурального числа (розмір в бітах якого вважаємо невідомим до вводу) та виведіть отримане число в десятковому та шістнадцятковому вигляді.
- 11) Ввести натуральне 64-бітне число m . Встановіть її ліві k біт рівним нулеві та виведіть отримане число. Встановіть її праві k біт рівним нулеві та виведіть отримане число в десятковому та вісімковому вигляді. Розв'яжіть задачу для типу m unsigned та long long unsigned.
- 12) Ввести натуральне число m . Поміняйте місцями біти її двійкового запису з номерами i та j (що теж вводяться) та виведіть отримане число в десятковому та шістнадцятковому форматі.
- 13) Знайдіть кількість значущих (не рівних 0) бітів натурального 32-бітного числа.
- 14) За допомогою лише бітових операцій та операції декременту з'ясуйте чи є дане натуральне число ступенем двійки. Спробуйте з циклом та без циклу. (Підказка: подумайте, як виглядає бітове представлення декременту ступеню двійки, та використайте далі кон'юнкцію).
- 15) Ввести натуральні 32-бітні числа m та n та визначить скільки в них спільних одиничок бітового представлення. Визначить скільки в цих числах взагалі співпадає бітів.
- 16) Виведіть бітове (двійкове) представлення натурального числа, виводячи замість цифри 1 символ X, а замість цифри 0 – символ Y.
- 17) Інвертуйте (тобто прочитайте зліва направо) бітове представлення даного числа та виведіть двійкове представлення та десяткове для цієї інверсії.
- 18) Ввести натуральне однобайтове число n і вивести число, отримане в результаті циклічного зсуву числа n на один розряд вліво, тобто старший біт зсунутий в позицію молодшого, а всі інші біти зсуваються на один розряд вліво. Наприклад, якщо введено 130, відповіддю буде 5.

- 19) Визначити, скільки разів зустрічається 11 в двійковому поданні цілого додатного числа (в двійковому поданні 11110111 воно зустрічається 5 разів).
- 20) Викреслити i -тий біт з двійкового представлення натурального числа (молодші i бітів залишаються на місці, старші зсуваються на один розряд вправо). Наприклад, якщо введені 11 і 2, відповіддю буде 7.
- 21) Написати функцію, результатом якого є дане значення n , у якого молодший нульовий біт та найстарший нульовий біт встановлені в 1.
- 22) Написати функцію, результатом якого є дане значення n , у якого все біти встановлені в 1, крім молодших k бітів, які залишаються такі самі.
- 23) Підрахуйте кількість нулів серед бітів даного числа.
- 24) Знайдіть номер найстаршого значущого біта в даному 32-бітному числі.
- 25) Напишіть функцію, що визначає чи два натуральних числа не мають одиничних бітів на однакових позиціях.
- 26) Напишіть функцію, що визначає чи два натуральних числа не мають нульових бітів на однакових позиціях.
- 27) Напишіть функцію, що визначає чи два натуральних числа не мають однакових бітів на однакових позиціях.

Розділ 7

Статичні масиви. Лінійні масиви

Контрольні запитання:

- Які варіанти декларації масивів на С. На C++?
- Які варіанти ініціалізації масивів на С. На C++?
- Як найкраще передавати масив у аргументи функції?
- Чи можна повернути масив фіксованого розміру як результат функції?
- Чому масив як аргумент краще передавати через вказівник чи посилення?

Завдання для аудиторної роботи:

- 1) Ініціалізуйте масив 5 цілих чисел в програмі довільним чином. Введіть дійсне число та знайдіть кількість чисел у вашому масиві, що менше зі це число.
- 2) Масив заповнений таким чином: 5, 112, 4, 3. Вивести його елементи навпаки: 3,4,112,5. При цьому використання циклу є обов'язковим.
- 3) Заповнити масив типу double з 10 елементів з клавіатури (по черзі в циклі вводяться всі елементи) і знайти суму всіх елементів більших за число Ейлера e .
- 4) Масив типу int з 5 елементів заповнюється з клавіатури. Знайти і вивести на екран максимальне значення у вашому масиві.
- 5) Знайти суму всіх парних і непарних елементів масиву натуральних чисел. Масив заповнюється з клавіатури, 5 елементів.
- 6) Написати функції, в яких якщо потрібно повернути результат – масив, то це робиться за допомогою змінного аргументу функції:
 - функція вводу n -вимірному вектору дійсних чисел ($n < 20$);
 - функція виводу n -вимірному вектору дійсних чисел;
 - функція рахує суму двох векторів однакового розміру;
 - функція рахує скалярний добуток двох векторів однакового розміру.

Протестувати роботи цих функцій: ввести в головній програмі розмірність векторів, два вектори цієї розмірності та підрахувати їх суму та скалярний добуток і вивести результати.

Завдання для самостійної роботи:

7) Написати функцію, що вводить послідовність ненульових цілих чисел, введення завершується при вводі нуля. Кількість елементів масиву обмежена числом 20. Визначити середнє геометричне та середнє гармонічне цієї послідовності.

8) Вводиться масив натуральних чисел заданого розміру N :

- а) визначити скільки серед цих чисел повних квадратів простих чисел;
- б) визначити скільки серед цих чисел парних повних кубів;
- в) визначити скільки серед цих чисел n -тих ступенів цілих чисел (для будь-якого $n > 1$);
- г) визначити скільки серед них цілих ступенів двійки;
- д) визначити скільки серед них чисел, що кратні 3 та є кубами цілих чисел;
- е) визначити скільки серед них простих чисел;
- є) визначити скільки серед них чисел Фібоначчі;
- ж) визначити скільки серед них чисел, у яких 5-й, 6-й та 8-й біт двійкового запису дорівнюють 1;
- з) визначити скільки серед них чисел, які містять рівно 5 бітів в двійковому записі, що дорівнюють 1;
- і) визначити скільки серед них чисел, у яких сума цифр в десятковому запису ділиться на 7.

9) Задані натуральне число n та дійсні числа a_1, a_2, \dots, a_n . Скласти програму для знаходження:

- | | |
|-----------------------------------|--|
| а) $\max(a_1, a_2, \dots, a_n)$; | д) $\min(a_2, a_4, \dots) + \max(a_1, a_3, \dots)$; |
| б) $\min(a_1, a_2, \dots, a_n)$; | е) $\max(a_1 , \dots, a_n)$; |
| в) $\max(a_2, a_4, \dots)$; | є) $\max(-a_1, a_2, -a_3, \dots, (-1)^n a_n)$; |
| г) $\min(a_1, a_3, \dots)$; | ж) $\frac{(\min(a_1, \dots, a_n))^2}{\min(a_1^2, \dots, a_n^2)}$. |

10) Дано натуральне число n , цілі числа a_1, a_2, \dots, a_n . Скласти програму знаходження

- | | |
|--|--|
| а) $\min(a_1, 2a_2, \dots, na_n)$; | д) кількості повних квадратів серед $a_1a_n, a_1a_{n-1}, \dots, a_ka_{n-k}, k = [n/2]$; |
| б) $\min(a_1 + a_2, \dots, a_{n-1} + a_n)$; | |
| в) $\max(a_1, a_1a_2, \dots, a_1a_2 \dots a_n)$; | |
| г) кількості парних серед $a_1^2, a_2^2, \dots, a_k^2, k = [\sqrt{n}]$; | е) кількості квадратів непарних чисел серед a_1, a_2, \dots, a_n . |

11) Скласти функції для обчислення

- а) Значення поліному Чебишова заданого степеня n в точці x

$$T_0(x) = 1, T_1(x) = x,$$

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), n = 2, 3, \dots;$$

та функцію, що виводить коефіцієнти поліному Чебишова степеня $n < 256$.

- б) Значення поліному Ерміта заданого степеня n в точці x

$$H_0(x) = 1, H_1(x) = 2x,$$

$$H_n(x) = 2xH_{n-1}(x) - 2(n-1)H_{n-2}(x), n = 2, 3, \dots$$

та функцію, що виводить коефіцієнти поліному Ерміта для степеня $n < 256$.

12) В цілочисельному масиві $A[N]$ знайдіть моду, тобто елемент, що зустрічається найбільшу кількість разів. Якщо таких елементів декілька, то виведіть всі такі елементи.

13) В цілочисельному масиві $A[N]$ знайдіть елемент, що є найближчим до середнього арифметичного найбільшого та найменшого елементу масиву.

14) Напишіть функцію, яка в дійсному масиві $A[N]$ знаходить середнє відхилення (варіацію) масиву

15) Знайдіть в даному цілому числі цифру десяткового запису, яка зустрічається найбільшу кількість разів. Якщо їх декілька, виведіть найбільшу цифру.

16) Напишіть функцію, яка за заданим масивом значень $\{x_i\}_{i=1}^d$ обчислює:

$$f(x) = \sum_{i=1}^d (100x_{i+1} - x_i)^2 + (x_i - 1)^2.$$

- 17) В деяких видах спортивних змагань виступ кожного спортсмена незалежно оцінюється деякими суддями, потім з усієї сукупності оцінок видаляються найбільша та найнижча, а для решти оцінок обчислюється середнє арифметичне, яке і йде в залік спортсмену. Якщо найбільшу оцінку виставило декілька суддів, то з сукупності оцінок видаляється лише одна така оцінка; аналогічно з найнижчими оцінками. Дано натуральне число n , дійсні числа a_1, a_2, \dots, a_n . Вважаючи, що a_1, a_2, \dots, a_n – оцінки, виставлені суддями одному з учасників змагань, визначити оцінку, яка піде в залік цього спортсмену.

Додаткові задачі:

- 18) По заданих значеннях коефіцієнтів поліномів $P(x)$ та $Q(x)$ знайдіть значення коефіцієнтів поліному $P(Q(x))$.
- 19) Обчислити коефіцієнти багаточлена з заданими дійсними коренями $x[0], x[1], \dots, x[n]$. Кількість коефіцієнтів обмежена числом 100.
- 20) Побудувати N -розрядний код Грея. Кодом Грея зветься така послідовність N -розрядних двійкових чисел, в яких кожен два сусідніх, а також перше й останнє числа відрізняються лише одним розрядом. Так, для $N=2$ код Грея наступний: 00,01,11,10. Для $N=3$: 000,001,011,010,110,111,101. Переведіть всі числа з цього двійкового коду до десяткової системи числення.
- 21) В цілочисельному масиві $A[N]$ (не обов'язково впорядкованому) знайдіть медіану, тобто величину, що ділить ряд навпіл: по обидві сторони від неї знаходиться однакова кількість одиниць сукупності. Тобто, якщо кількість чисел непарна, обирається елемент, що є середнім за зростанням. Наприклад, для впорядкованого набору чисел 1, 3, 3, 6, 7, 8, 9 медіаною є четверте із них, число 6. Якщо кількість елементів парна, тоді медіану зазвичай визначають як середнє значення між двома числами по середині впорядкованого масиву. Наприклад, для наступного набору 1, 2, 3, 4, 5, 6, 8, 9 – медіана є середнім значенням для двох чисел по середині: вона дорівнюватиме $(4 + 5)/2 = 4.5$.

Розділ 8

Статичні масиви. Багатовимірні масиви

Контрольні запитання:

- Як коректно задекларувати багатовимірний масив? Що насправді означає цей запис?
- Як коректно ініціалізувати багатовимірний масив?
- Як ввести розміри та вміст двовимірної дійсної матриці?
- Як красиво рядок за рядком вивести матрицю?
- Як повернути коректно дані з масиву (в тому числі багатовимірного) з функції?
- Які є шляхи поміняти місцями два елементи масиву?

Завдання для аудиторної роботи:

- 1) Двовимірна матриця 3×3 ініціалізована цілими числами $\{\{1,2,3\}, \{4,5,6\}, \{7,8,9\}\}$. Введіть натуральні числа N і M та замініть елемент, що рівний числу M (якщо він є в матриці) на число N . Виведіть отриману матрицю рядок за рядком.
- 2) Двовимірна дійсна матриця 3×3 ініціалізована $\{\{1.0, 2.3\}, \{4.5, 6\}, \{7.8, 9\}\}$. Введіть натуральні числа i та j та дійсне число a замініть елемент з індексами i, j на число a (простежити при цьому коректність індексів). Виведіть отриману матрицю рядок за рядком.
- 3) Напишіть функцію для вводу двовимірної дійсної матриці довільного розміру $m \times n$, яка вводить з підказкою для користувача (вказуючи індекси елементів) кожен елемент в одному рядку. Величини m, n вводяться з консолі та їх значення менші за 20.
- 4) Напишіть функцію для вводу двовимірної цілої (дійсної) матриці довільного розміру $m \times n$, яка вводить з підказкою для користувача (номер рядку) матрицю рядок за рядком (числа в рядку розділяються одним пробілом). Величини m, n вводяться з консолі та їх значення менші за 25.
- 5) Напишіть функцію яка транспонує дану квадратну матрицю та перевірте її роботу.

- 6) В двовимірному масиві $A[N,M]$ знайдіть суму елементів $A[i,j]$, що $i - j = k$. Ціле число k може бути від'ємним, якщо таких елементів немає, то вивести нуль. Величини M, N вводяться з консолі та їх значення менші за 100.

Завдання для самостійної роботи:

- 7) Ввести натуральні числа $n, m < 100$ та матрицю розміру $n \times m$. Поміняти місцями її стовпці так, щоб їх максимальні елементи в кожному стовпці були на діагоналі.
- 8) Знайдіть квадратну матрицю, обернену до даної введеної з розміром $n \times n$.
- 9) Введіть квадратну матрицю заданого порядку $2n$. Повернути її на 180 градусів в за годинникову стрілкою.
- 10) Заповнити двовимірний квадратний масив $n \times n$ цілими числами від 1 до n^2 по спіралі, як показано на наступному малюнку:

```

37-36-35-34-33-32-31
|
38 17-16-15-14-13 30
|
39 18 5-4-3 12 29
|
40 19 6 1-2 11 28
|
41 20 7-8-9-10 27
|
42 21-22-23-24-25-26
|
43-44-45-46-47-48-49...

```

- 11) Дана матриця розміру $n \times m$. Поміняти місцями стовпці, що містять мінімальний і максимальний елементи матриці.
- 12) Дано дві матриці $n \times m$ і $m \times k$. Отримайте їх добуток.
- 13) Дана матриця розміру $n \times m$. Поміняти місцями її рядки так, щоб їх максимальні елементи стовпців були на останньому рядку.
- 14) У даній дійсній квадратній матриці порядку n знайти найбільший по модулю елемент.
- 15) Отримати квадратну матрицю порядку $n - 1$ шляхом викидання з вихідної матриці будь-якого рядка і стовпця, на перетині яких розта-

шований елемент зі знайденим значенням. Виконуйте до тих пір, поки не залишиться останній елемент.

- 16) Дана квадратна матриця порядку $2n + 1$. Дзеркально відобразити її елементи відносно побічної діагоналі матриці.
- 17) Дана дійсна квадратна матриця порядку $2n + 1$. Отримати нову матрицю, повернувши її блоки, обмежені діагоналями, на 90 градусів.
- 18) Дана матриця розміру $n \times m$. Поміняти місцями її перший і останній рядки, що містять тільки негативні елементи.
- 19) Дана цілочисельна матриця розміру $n \times m$. Знайти елемент, який є максимальним у своєму рядку і мінімальним в своєму стовпці. Якщо такий елемент відсутній, то вивести 0.
- 20) Складіть програму циклічної перестановки стовпців двовимірного масиву $n \times m$, при якій зсув зсувається вправо на k стовпців.
- 21) Дана матриця розміру $n \times m$. Поміняти місцями її стовпці так, щоб їх мінімальні елементи кожного стовпчика утворювали були у першому рядку.
- 22) Дана квадратна матриця порядку $2n + 1$. Дзеркально відобразити її елементи відносно вертикальної оС симетрії матриці.
- 23) Дана квадратна матриця порядку $2n$. Повернути її на 270 градусів в додатньому напрямку щодо її центру.
- 24) Дана матриця розміру $n \times m$. Поміняти місцями рядки, що містять мінімальний і максимальний елементи матриці.
- 25) У квадратній таблиці обміняйте місцями елементи рядка і стовпця, на перетині яких знаходиться мінімальний з позитивних елементів.
- 26) Дана квадратна матриця порядку $2n$. Повернути її на 90 градусів за годинниковою стрілкою відносно її центру.
- 27) Дана квадратна матриця порядку $2n + 1$. Дзеркально відобразити її елементи відносно головної діагоналі матриці.
- 28) Складіть програму циклічної перестановки рядків двовимірного масиву $n \times m$, при якій зсув відбувається вниз на k рядків.
- 29) Дана матриця розміру $n \times m$. Поміняти місцями її перший і останній стовпці, що містять лише додатні елементи.

- 30) Заповніть квадратну матрицю $n \times n$ за принципом латинського квадрата: в кожному рядку і кожному стовпці використовуються лише числа від 1 до n що не повторюються між собою.
- 31) Дана матриця дійсних коефіцієнтів. Впорядкувати її рядки по неспаданню перших елементів, суми значень рядків, величині найменших елементів рядків.

Додаткові задачі:

- 32) Дана матриця $n \times m$ з нулів та одиниць. Знайти найбільший за площиною прямокутник з одних одиниць. Зробіть цю задачу без вкладених 4-х циклів для $0 < n, m < 255$.
- 33) Нехай значення функції $f(n)$ – кількість літер у письмовому представленні числа n ($f(1) = 4$ ("один"), $f(3) = 3$ ("три"), $f(42) = 8$ ("сорок два"), $f(2001) = 13$ ("дві тисячі один")). Знайдіть всі числа до 10000, для яких $f(n) = n$. (Вказівка: Використовуйте масив для зберігання кількості літер у представленні цифри)

Розділ 9

Динамічні масиви. Робота з вказівниками

Контрольні запитання:

- Як можна створити лінійний динамічний масив та коректно завершити при цьому програму?
- Що таке вказівники? Які операції визначені на вказівниках? Як проітеруватись по даному масиву за допомогою вказівника?
- Як визначити динамічну матрицю за допомогою масиву вказівників та коректно її обробити?
- Які функції та з якої бібліотеки використовуються на C для виділення пам'яті? В чому їх різниця? Що відбудеться якщо потрібної пам'яті не було ними виділено?
- Які функції існують для очищення пам'яті? Що відбудеться, якщо їх не використовувати? Які ще проблеми виникають при некоректному очищенні чи його відсутності?

Завдання для аудиторної роботи:

- 1) Ввести натуральне число n . Створити та ввести масив з n дійсних чисел та підрахувати суму квадратів елементів цього масиву.
- 2) Написати функцію, що вводить масив цілих чисел доки не буде введений нуль через змінний аргумент та кількість елементів масиву повертається як результат роботи функції. Кількість елементів обмежена числом 100. Підрахувати кількість повних квадратів та кубів в цьому масиві.
- 3) Створити функцію, що вводить n -вимірний вектор (n задається як аргумент функції, результат – вказівник на результатний масив), виділяючи відповідну пам'ять та функцію, що відповідно очищує пам'ять. Напишіть програму, що вводить два вектори, підраховує та створює як окремий масив їх різницю якщо це можливо, та в будь-якому варіанті коректно завершує програму без витоків пам'яті.
- 4) Створити функції, що коректно ініціалізують нулями та вводять з консолі дійсну квадратну n -вимірну матрицю (n задається як аргумент функції), й функцію, що відповідно очищує пам'ять. Напишіть програму, що вводить дві матриці, підраховує та обчислює як окремий

масив їх добуток, якщо це можливо, та в будь-якому варіанті коректно завершує програму без витоків пам'яті. Зробіть дану задачу:

- представляючі матрицю у вигляді двовимірного масиву;
- представляючі матрицю у вигляді лінійного масиву розміру n^2 .

Завдання для самостійної роботи:

- 5) Створити функцію, що вводить матрицю цілих чисел довільних розмірностей, виділяючи відповідну пам'ять (розміри масивів) та функцію, що відповідно очищує пам'ять. Напишіть функцію, що підраховує детермінант та ранг матриці. Коректно протестуйте роботу цих функцій.
- 6) Створити функцію, що вводить матриці довільних розмірностей, виділяючи відповідну пам'ять та функцію, що відповідно очищує пам'ять. Напишіть програму, що вводить масив таких матриць, підраховує та створює як окрему матрицю суму всього масиву матриць, якщо всі матриці однакового розміру, та в будь-якому варіанті коректно завершує програму без витоків пам'яті.
- 7) Ввести натуральне число n . Створити та ввести масив з n натуральних довгих чисел та підрахувати кількість ступенів двійки та трійки в цьому масиві.
- 8) Користувачу надається можливість декілька разів вводити розмірність вектору дійсних чисел та самі ці значення. Після кожного вводу потрібно підрахувати середнє арифметичне та дисперсію всіх введених значень.
- 9) Вирішіть завдання виконуючи наступні вимоги:

Сформувати динамічний двовимірний дійсний масив $N \times M$, заповнити його випадковими числами або з консолі та вивести на екран. Виконати наступні дії, коректно оброблюючи всі можливі сценарії:

- а) додати рядок після заданого номера k ;
- б) додати стовпець після заданого номера k ;
- в) додати рядок в кінець матриці;
- г) додати стовпець в кінець матриці;
- д) додати рядок в початок матриці;
- е) додати стовпець в початок матриці;

- є) додати k рядків в кінець матриці;
- ж) додати k стовпців в кінець матриці;
- з) додати k рядків в початок матриці;
- і) додати k стовпців в початок матриці;
- й) видалити рядок з номером k ;
- к) видалити стовпець з номером k ;
- л) видалити рядки, починаючи з рядка $k1$ і до рядка $k2$;
- м) видалити стовпці, починаючи зі стовпця k і до стовпчика k ;
- н) видалити всі непарні рядки;
- о) видалити всі парні стовпці;
- п) видалити всі рядки, в яких є хоча б один нульовий елемент;
- р) видалити всі стовпці, в яких всі елементи менші за 1;
- с) видалити рядок, в якій знаходиться найменший за модулем елемент матриці (якщо їх декілька – видалити усі);
- т) додати рядок після кожного парного рядка матриці;
- у) додати стовпець після кожного парного стовпця матриці;
- ф) додати k рядків, починаючи з рядка за номером m ;
- х) додати k стовпців, починаючи зі стовпчика за номером m ;
- ц) додати рядок після рядка, що містить найбільший елемент;
- ч) додати стовпець після стовпця, що має найбільшу суму елементів;
- ш) додати рядок після рядка, що має найменше значення норми (суми квадратів)(якщо їх декілька – обираємо останній);
- ю) додати стовпець після стовпця, що містить найменший за модулем елемент (якщо їх декілька – обираємо перший);
- я) видалити рядок і стовпець, на перетині яких знаходиться найбільший елемент матриці.

Додаткові задачі:

- 10) Створити функцію, що вводить матриці довільних розмірностей, виділяючи відповідну пам'ять та функцію, що відповідно очищує пам'ять. Напишіть програму, що вводить масив таких матриць, підраховує та створює як окрему матрицю добуток всього масиву матриць, якщо це можливо, та в будь-якому варіанті коректно завершує програму без витоків пам'яті.
- 11) Петя та Вася кожен день на протязі N днів вимірюють декілька (від 0 до 1000) разів температуру повітря (хоча інколи хтось може забути це зробити). Створіть програму, що дозволить їм ввести ці результати за кожен день спостережень та підрахує середню температуру кожного з цих днів, де сумарна кількість вимірювань була більше одного. Програма повинна передбачити, що після вводу цих N днів вони можуть захотіти ввести наступні M днів таких спостережень. Передбачте можливість коректного завершення при нестачі ресурсів комп'ютера для зберігання та обробки даних.
- 12) В масиві натуральних чисел $A[N]$ всі числа є меншими ніж 16. Напишіть функцію, що зберігає дані цього масиву у масиві $N/2$ чисел типу `uint8_t` (тобто в кожному числі `uint8_t` зберігається два числа масиву $A[i]$).
- 13) В масиві натуральних чисел $A[N]$ всі числа є меншими ніж 64. Напишіть функцію, що зберігає дані цього масиву у масиві $[N * 4/3]$ чисел типу `uint8_t` (тобто в кожних трьох числах `uint8_t` зберігається чотири числа масиву $A[i]$).
- 14) В масиві натуральних чисел $A[N]$ всі числа є меншими 2^k . Знайдіть це число k та напишіть функцію, що зберігає цей масив в $N * k$ біт найбільш економічним чином(`int A[3], k=5 → uint8 B[2]` , тобто використовує 16 біт, або `int A[8], k=14 → uint16 B[7]` , тобто використовує 112 біт) та функцію що переводить навпаки числа з масиву B у масив A .

Розділ 10

Структури. Створення власного типу

Контрольні запитання:

- Що таке структура та як її створити на C?
- Як створити власний тип даних на C?
- Як визначити структуру, що має посилання на саму себе?
- Які варіанти ініціалізації структур? Як ввести структуру? Як отримати структуру як результат роботи функції? Через змінний аргумент?
- Нащо використовувати typedef при створенні власної структури?

Завдання для аудиторної роботи:

- 1) Визначити типи структури для зображення наступних понять та функції їх вводу-виводу:
 - а) дата (число, місяць, рік);
 - б) поле шахової дошки (напр., a5, b8);
 - в) прямокутник зі сторонами, паралельними осям координат, заданий через дві вершини. Вершина в свою чергу – теж структура яка містить дві дійсні координати;
 - г) поліном довільного ступеня (ступінь поліному та дійсні коефіцієнти — безрозмірний масив).
- 2) Використовуючи тип Поле шахової дошки описати булеву функцію, яка перевіряє, чи може за один хід ферзь перейти з одного заданого поля шахової дошки на інше задане поле.
- 3) Використовуючи опис типу Дата, визначити функції обчислення: дати завтрашнього дня та дня тижня за його датою в поточному році.
- 4) Визначимо тип Rational (Раціональне число) як:

```
typedef struct {  
    int numerator; // чисельник  
    unsigned int denominator; // знаменник  
} Rational;
```

Визначити функції для:

- обчислення суми двох раціональних чисел;
 - обчислення добутку двох раціональних чисел;
 - порівняння двох раціональних чисел;
 - зведення раціонального числа до нескоротного виду.
- 5) Задано масив розмірності N , компонентами якого є структури, що містять відомості про вершини гір. У відомостях про кожну вершину вказуються назва гори та її висота. Визначити функції введення/виведення гір та функції пошуку назви найвищої вершини та виведення висоти вершини з заданою назвою (якщо вершини з такою назвою немає в масиві – вивести відповідне повідомлення).

Завдання для самостійної роботи:

- 6) Визначити типи запису для зображення наступних понять та реалізуйте їх функції введення виведення:
- а) вартість (гривні, копійки);
 - б) час (година, хвилина, секунда);
 - в) повна дата (число, місяць, рік, година, хвилина);
 - г) адреса (місто, вулиця, будинок, квартира);
 - д) семінар (предмет, викладач, № групи, день тижня, години занять, аудиторія);
 - е) бланк вимоги на книгу (відомості про книгу: шифр, автор, назва; відомості про читача: № читачього квитка, прізвище; дата замовлення);
 - є) коло (радіус, координати центру);
 - ж) сфера в просторі;
 - з) прямокутний паралеліпіпед (сторони якого паралельні вісям координат);
 - і) вектор (розмір вектору – натуральне число та масив дійсних значень);
 - й) матриця (розміри матриці – два натуральних числа та масив дійсних значень);
 - к) багатокутник (розмір багатокутника та набір координат вершин).

- 7) В масиві структур записано вартість та вік кожної з N моделей легкових автомобілів. Визначити середню вартість автомобілів, вік яких більший за 5 років.
- 8) В масиві структур записано інформацію про ціну та наклад кожного з N журналів. Знайти середню вартість журналів, наклад яких менший за 10000 примірників.
- 9) В масиві структур записано дані про масу й об'єм N предметів, виготовлених із різних матеріалів. Знайти предмет, густина матеріалу якого найбільша.
- 10) В масиві структур записано дані про чисельність населення (у мільйонах жителів) та площі N держав. Знайти країну з мінімальною щільністю населення.
- 11) Задано масив P розмірності N , компонентами якого є відомості про мешканців деяких міст. Інформація про кожного мешканця містить його прізвище, назву міста, місцеву адресу у вигляді вулиці, будинку, квартири. Визначити функцію пошуку двох будь-яких жителів, що мешкають у різних містах за однаковою адресою.
- 12) Відомо дані про вартість кожного з N найменувань товарів: кількість гривень, кількість копійок. Скласти підпрограми пошуку:
 - а) найдешевшого товару в магазині;
 - б) найдорожчого товару в магазині;
 - в) товару, вартість якого відрізняється від середньої вартості товару в магазині не більш ніж на 10 гривень;
- 13) Задано масив P розмірності N , компонентами якого є структури, що містять анкети службовців деякого закладу. У кожній анкеті вказується прізвище та ім'я службовця, його стать, дата народження у вигляді числа, місяця, року. Визначити підпрограми пошуку:
 - а) посади, яку обіймає найбільша кількість співробітників;
 - б) співробітників з однаковими іменами;
 - в) співробітників, прізвища яких починаються із заданої літери;
 - г) найстаршого з чоловіків цього закладу;
 - д) співробітників, вік яких менший за середній по організації;

- е) пенсійного віку (врахувати, що пенсійний вік чоловіків і жінок – різний).
- 14) Задано масив імен L та масив P , компонентами P_i якого є записи, що містять дані про людину на ім'я з номером i з масиву L . Кожний запис складається з номером імені, статі людини та її зросту. Визначити підпрограми для:
- а) обчислення середнього зросту жінок;
 - б) пошуку імені найвищого чоловіка;
 - в) перевірки, чи є дві людини, однакові на зріст та виводу імені цієї людини.
- 15) Задано масив розмірності N , компоненти якого містять інформацію про студентів деякого вишу. Відомості про кожного студента містять дані про його прізвище, ім'я, по батькові, стать, вік, курс. Визначити процедуру пошуку:
- а) найпоширеніших чоловічих і жіночих імен;
 - б) прізвищ та ініціалів усіх студентів, вік яких є найпоширенішим.
- 16) Задано масив розмірності N , компонентами якого є відомості про складання іспитів студентами деякого вишу. Інформація про кожного студента задана в такому вигляді: прізвище, номер групи, оцінка_1, оцінка_2, оцінка_3. Визначити процедуру пошуку:
- а) студентів, що мають заборгованості принаймні з одного з предметів;
 - б) предмета, складеного найгірше;
 - в) студентів, що склали всі іспити вище за задану оцінку.

Додаткові задачі:

- 18) Визначити універсальний тип, який допускає зображення точки на площині у прямокутній або полярній системі координат (третє поле – тип координат). Побудувати функцію обчислення площі трикутника з вершинами A , B , C .

Розділ 11

Робота з бінарними файлами на C

Контрольні запитання:

- Цикл роботи з файлами на C/C++.
- Як створити та працювати з текстовим файлом на C? Як можна вводити та виводити файл посимвольно? Порядково?
- Чим відрізняється бінарний файл від текстового?
- Як створити бінарний файл? Як читати з бінарного файлу?
- Як записати та прочитати масив дійсних чисел в/з бінарного файлу?
- Як прочитати всі цілі числа з файлу, якщо на початку роботи невідомо, скільки їх там насправді?
- Які додаткові речі можна робити з бінарним файлом, що неможна робити з текстовим?
- Як записати масив структур у файл та прочитати k -тий запис у файлі?

Завдання для аудиторної роботи:

- 1) Введіть довжину масиву дійсних чисел та сам масив. Реалізуйте функцію, яка запише цей масив в файл з заданим ім'ям. Реалізуйте функцію, що виведе вміст файлу з дійсних чисел на консоль в одному рядку через коми.
- 2) Використовуючи файл F, компоненти якого є дійсними числами, побудувати файл G, що містить усі числа з файлу F, які менші по модулю за задане число $a > 0$. Послідовність чисел зберігається. Після цього видалити всі елементи з файлу F, які менші по модулю за число a .
- 3) Створіть файл F, компоненти якого є цілими числами. При цьому введення цілих чисел робиться в нескінченному циклі, доки користувач не введе 0. Побудувати файл G, який містив би всі компоненти файлу F:
 - а) що є парними числами;
 - б) що є ступенем 3;
 - в) що є точними квадратами;

- г) що мають лише 3 ділянки;
 - д) що є паліндромами;
 - е) що є числами Фібоначчі.
- 4) Створить файл, який містить відомості про прямокутники: вказано номер прямокутника у файлі, координати (дійсні числа) верхнього лівого кута та нижнього правого кута прямокутника. Скласти функцію пошуку номера прямокутника з найбільшою площею й визначення цієї площі. Скласти функції видалення прямокутника за даним номером та заміни запису про прямокутник за порядковим номером у файлі.
- 5) У файлі записано декілька поліномів у наступному вигляді: для кожного поліному записано натуральне число – кількість мономів m (ненульових доданків поліному) та далі m мономів, які представлені структурами вигляду $(coef, deg)$, де $coef$ – дійсний коефіцієнт, deg ступінь моному ($deg \geq 0$). Таким чином в файлі записано декілька поліномів (номер поліному встановлюється порядком запису поліному). Визначити підпрограми для виконання наступних дій над поліномом:
- а) введення полінома та запис (додавання) його в файл заданий своїм іменем;
 - б) друк поліному з файлу за номером на консоль у звичному вигляді поліному від x ;
 - в) обчислення похідної від полінома за файлом та номером;
 - г) додавання поліному у файл заданий іменем;
 - д) видалення поліному за даним номером;
 - е) заміна коефіцієнту заданого ступенем та номером поліному;

Завдання для самостійної роботи:

- б) Дано файл, компоненти якого є натуральними числами. Скласти підпрограми для обчислення:
- а) кількості парних чисел серед компонент;
 - б) кількості квадратів непарних чисел серед компонент;
 - в) різниці між найбільшим парним і найменшим непарним числами компонент;
 - г) кількості простих чисел серед компонент;

- д) кількості компонент у найдовшій зростаючій послідовності компонент файлу.
- 7) Дано файл, компоненти якого є дійсними числами. Скласти підпрограми для обчислення:
- а) суми компонент файлу;
 - б) кількості від'ємних компонент файлу;
 - в) останньої компоненти файлу;
 - г) передостанньої компоненти;
 - д) найбільшої по модулю компоненти файлу;
 - е) найменшої з компонент файлу з парними номерами;
 - є) суми найбільшої та найменшої з компонент;
 - ж) різниці першої й останньої компоненти файлу;
 - з) кількості компонент файлу, які менші за середнє арифметичне всіх його компонент.
- 8) Задано натуральне число n та файл F, компоненти якого є цілими числами. Побудувати файл G, записавши до нього найбільше значення перших n компонент файлу F, потім – наступних n компонент тощо. Розглянути два випадки:
- кількість компонент файлу ділиться на n ;
 - кількість компонент файлу не ділиться на n .
- Остання компонента файлу G має дорівнювати найбільшій із компонент файлу F, які утворюють останню (неповну) групу.
- 9) Дано файл F, компоненти якого є цілими числами. Файл містить рівну кількість додатних і від'ємних чисел – перевірте це і в протилежному випадку видайте відповідне повідомлення та не робить нічого. Використовуючи допоміжний файл H, переписати компоненти файлу F до файлу G так, щоб у файлі G:

- а) не було двох сусідніх чисел одного знаку;
- б) спочатку йшли додатні, потім – від'ємні числа;
- в) числа йшли таким чином: два додатних, два від'ємних тощо. Якщо це неможливо – то переписати поки можливо в такому вигляді, а останні два числа вивести на консоль.

- 10) Дано файл F, компонентами якого є записи (структури) вигляду

```
struct TPair {  
    unsigned key; // ключ  
    char data[10]; // дані  
};
```

Такий файл називатимемо впорядкованим за ключами, якщо записи в ньому розташовуються в порядку зростання (спадання) ключів. Скласти процедуру пошуку запису за ключем у впорядкованому файлі. Скласти процедуру вилучення запису із заданим ключем:

- з впорядкованого файлу;
- з неупорядкованого файлу.

- 11) Багаж пасажирів характеризується номером пасажирів, кількістю речей і їхньою загальною вагою. Дано файл пасажирів, який містить прізвища пасажирів, і файл, що містить інформацію про багаж декілька пасажирів (номер пасажирів – це номер запису у файлі пасажирів). Скласти процедури для:

- а) знаходження пасажирів, у багажі якого середня вага однієї речі відрізняється не більш ніж на a кг від загальної середньої ваги речей;
- б) визначення пасажирів, які мають понад двоє речей, і пасажирів кількість речей у яких більша за середню кількість речей;
- в) видачі відомостей про пасажирів, кількість речей у багажі якого не менша, ніж у будь-якому іншому багажі, а вага речей – не більша, ніж у будь-якому іншому багажі із цією самою кількістю речей;
- г) визначення, чи мають принаймні два пасажирів багажі, які не відрізняються за кількістю речей і відрізняються вагою не більш ніж на a кг (якщо такі пасажирів є, то показати їхні прізвища);
- д) визначення пасажирів, багаж якого складається з однієї речі вагою не менше m кг.

- 12) Дано файл, який містить відомості про іграшки: указано назву іграшки (напр., м'яч, лялька, конструктор тощо), її вартість у гривнях і вікові межі для дітей, яким іграшка призначається (напр., для дітей від двох до п'яти років). Скласти функції, що виводять наступні результати у бінарний файл та на консоль:

- а) пошуку назв іграшок, вартість яких не перевищує заданої кількості гривень, призначених дітям п'яти років;
- б) пошуку назв іграшок, призначені дітям i , n , і m років;
- в) пошуку назв найдорожчих іграшок (ціна яких відрізняється від ціни найдорожчої іграшки не більш ніж на a грн);
- г) визначення ціни найдорожчого товару з назвою *name*;
- д) визначення ціни всіх товарів з назвою *name*;
- е) пошуку двох іграшок, що призначені дітям x років, сумарна вартість яких не перевищує Y грн;
- є) пошуку конструктора ціною Y грн, призначеного дітям від x до 12 років. Якщо такої іграшки немає, то занести відомості про її відсутність до файлу.

Додаткові задачі:

- 13) У двох файлах міститься таблиця футбольного турніру, у першому – записано назви команд; у другому – результати матчів, що зберігаються у записах типу `T_Match`:

```
typedef struct {  
    unsigned int n1, n2;  
    unsigned int b1, b2;  
} T_Match;
```

Тут у структурі типу `T_Match` поля $n1$, $n2$ – номери першої і другої команд (тобто номери назв команд у файлі команд); $b1$, $b2$ – кількість м'ячів, забитих першою та другою командами, відповідно. Кожній команді за перемогу нараховується 3 очки, за нічию – 1, за поразку – 0. Із двох команд, які мають однакову кількість очок, першою вважається:

- та, що має кращу різницю забитих і пропущених м'ячів;
- за однакової різниці має більше забитих м'ячів;
- за всіма однаковими попередніми показниками визначається жеребкуванням (для жеребкування використати генератор випадкових чисел).

Знайти команду, яка є лідером.

Вказівка. Описати підпрограми створення файлів команд і матчів, додавання результату матчу, визначення лідера.

- 14) Файл бази даних з малюнками містить на початку ціле 32-бітне число 2051, потім ціле 32-бітне число K — кількість малюнків, а наступні два 32-бітних числа n, m — висота та ширина кожного малюнку у пікселях. При цьому ці числа задані в форматі big-indian. Наступний вміст файлу — беззнакові натуральні байти ($K * n * m$ байтів), кожен з яких — значення яскравостей пікселів (число від 0 до 255) кожного з цих малюнків, що проходяться у порядку зліва-направо та згоризонту.

Напишіть функцію, що перевіряє даний файл (заданий ім'ям) на відповідність даному формату та виводить масив яскравостей малюнка з заданим номером, якщо такий номер та сам файл коректно задані. В протилежному випадку вивести змістовне повідомлення про помилку.

Розділ 12

Робота з потоками введення/виведення C++.

Контрольні запитання:

- Як використовувати бібліотеки C на C++? Що потрібно для того, щоб код на C працював так само на C++?
- Яка різниця булевого типу та його використання на C та C++?
- Як вивести в C++ використовуючи потоки виведення дійсне число з заданою точністю? В науковому представленні? З заданою шириною?
- Що таке перевантаження функцій та навіщо воно може бути потрібно?
- Що таке new та new[]? Коли потрібно перше та коли друге?
- В чому різниця між new та malloc?
- Як очищувати пам'ять після new та new[]?

Завдання для аудиторної роботи:

- 1) Ввести в двох різних рядках послідовно два дійсних числа x та y та обчислити значення x в ступені y . Результат вивести в десятковому та науковому представленні.
- 2) На терміналі вводяться $10 * n$ цифр. Перші 10 цифр – це перше натуральне число, наступні 10 – друге і так далі. Введіть всі ці числа в масив розміру n та обчисліть і виведіть їх суму (вважайте що сума влезить в точність unsigned long long).
- 3) Ввести натуральне n та вивести на екран таблицю для коренів всіх чисел від 1 до n (вважаючи що n достатньо мале, щоб вміститись в консоль виводу) слідкуючи, щоб виведення було рівним (корінь під відповідним числом) та кількість цифр після коми була або 0 або 2:

```
+++++
  1    2    3    4    5
+++++
  1  1.41  1.73    2  2.24
+++++
```

- 4) Ввести з консолі натуральне число n та динамічний масив з n цілих чисел $\{m_i\}_{i=1}^n$ та введіть з текстового файлу n дійсних чисел $\{x_i\}_{i=1}^n$. Перевірте як наявність, так і коректність цього файлу. Обчисліть та виведіть у текстовий файл числа $\{x_i^{m_i}\}_{i=1}^n$.
- 5) Вхідний потік заданий текстовим файлом містить набір цілих чисел $A_i (0 \leq A_i \leq 10^{18})$, відділений один від іншого довільною кількістю пробілів і переводів рядків. Розмір вхідного потоку не перевищує 128 КБ. Для кожного числа A_i , починаючи з останнього та завершуючи першим, в окремому рядку вивести його квадратний корінь з трьома знаками після десяткової крапки.

Приклад:

Вхід:

1427 0

876652098643267843

5276538

Вихід:

2297.072

936297014.116

0.000

37.776

Завдання для самостійної роботи:

- 6) Ввести декілька (невідомо заздалегідь скільки, але менше 1000) дійсних чисел записаних через коми та обчислити значення функції $\log()$ для кожного з них. Якщо значення виходить за межі області вивести слово "None", для інших значень результат вивести в науковому та десятковому представленні шириною 5 символів.
- 7) В текстовому файлі записані дійсні числа розділені крапками з комами. Прочитайте ці дані в динамічний масив та для кожного елементу масиву обчисліть значення функції $\arcsin()$ для кожного з них. Якщо значення виходить за межі області визначення, то ігноруйте його. Виведіть результат в науковому представленні в інший текстовий файл.

- 8) В текстовому файлі записані цілі числа розділені новим рядком. Прочитайте ці дані в динамічний масив та для кожного елементу масиву обчисліть значення експоненти для кожного з них. Виведіть результат в десятковому представленні в той самий текстовий файл, додавши до кожного рядка цього файлу значення відповідної експоненти шириною 6 символів.
- 9) В текстовому файлі записані натуральні числа розділені пробілами. Перепишіть файл замінивши всі його компоненти які є більшими за наступне число на їх квадрати, а всі інші числа залишивши на їх позиціях.
- 10) В текстовому файлі записані цілі числа розділені комами та довільною кількістю пробілів. Перепишіть файл, видаливши всі його недодатні компоненти та помінявши роздільник на єдиний пробіл.
- 11) В текстовому файлі записані дійсні числа розділені крапками з комами. Перепишіть файл замінивши всі його від'ємні компоненти на відповідні модулі з роздільником кінця рядку та шириною кожного числа рівній максимальної точності серед даних дійсних чисел в початковому файлі.
- 12) Три додатних дійсних числа вводяться як рядок вигляду
 $A=xxx.xxx, B=xxExxx, C=xxx.xxxx$
При цьому рядок зобов'язаний містити символи як представлено в рядку. Обчисліть середнє гармонічне даних чисел, якщо це можливо та виведіть їх у науковому та звичайному форматі.
- 13) Ввести дійсне число від -10000 до 10000 та вивести його k -ту ступінь ($|k| < 10$) з точністю до 20 знаків до десяткової коми та 4 знаками після десяткової коми (нуль залишається нулем завжди).
- 14) Розв'яжіть задачу 5.3, так щоб вона працювала для довільного n , тобто забезпечте перенесення таблиці на нові рядки при перевищенні у виводі довжини рядку (в рядок нехай вміщується 8 стовпців таблиці) і вирівнювання було тепер по лівому краю.
- 15) У текстовому файлі записано непорожню послідовність дійсних чисел, які розділяються пробілами, табуляціями або кінцями рядків. Визначити функцію обчислення середнього геометричного з цих чисел.
- 16) У текстовому файлі F записано послідовність цілих чисел, які розділяються пропусками. Визначити процедуру запису до текстового файлу G усіх додатних чисел разом їх коренями з точністю до 5 знаку після

крапки із F. Числа в файлі F розділені табуляцією, а числа та їх корені розділені комою.

- 17) У текстовому файлі кожний рядок містить кілька натуральних чисел, які розділяються пропусками. Числа визначають вигляд геометричної фігури (номер) та її розміри. Прийнято такі домовленості:

- відрізок прямої задається координатами своїх кінців і має номер 1;
- прямокутник задається координатами верхнього лівого й нижнього правого кутів і має номер 2;
- коло задається координатами центра й радіусом і має номер 3.

Визначити процедури обчислення:

- відрізка з найбільшою довжиною;
- прямокутника з найбільшим периметром;
- кола з найменшою площею.

- 18) У файлі записані координати точок на площині задані парою цілих чисел. Точки записуються в форматі : (x_1 , x_2) (x_1 , x_2) , ... – саме так через коми та дужки. Створити файл, в якому будуть записані координати всіх відрізків з точок цього файлу, при цьому ці відрізки відсортовані за зростанням довжини.

- 19) У файлі записані координати Точок в просторі задані трійкою цілих чисел. Точки записуються в форматі : x_1 , x_2 , x_3 ; x_1 , x_2 , x_3 ; ... Знайти відрізок з точок цього файлу, що має найбільшу довжину.

- 20) У файлі записані координати матеріальних точок на площині задані парою цілих чисел та масою(дійсне число). Точки записуються в форматі : [x_1 , y_1 , m_1] , [x_2 , y_2 , m_2] , ... – саме так через коми та дужки. Знайдіть дві точки з найбільшим важелем сили ($m \cdot (x + y)$).

- 21) У файлі записані дати , що задані трійкою цілих чисел у форматі (чч1./мм1/рр1),(чч2./мм2/рр2), ... – саме в такому форматі. Створити файл, в якому будуть записано найстарша та найсвіжіша дати (вважайте, що роки дат з 1951 по 2049).

Додаткові задачі:

- 22) Розглянемо послідовність чисел a_i , $i = 0, 1, 2, \dots$, що задовольняють умовам:

$a_0 = 0, a_1 = 1$; та $a_{2i} = a_i$ і $a_{2i+1} = 2a_i + 1$ для кожного $i = 1, 2, 3, \dots$

Напишіть програму, яка для заданого значення n знаходить максимальне серед чисел a_0, a_1, \dots, a_n . Вхідні дані складаються з декількох тестів (не більше 10). Кожен тест - рядок, в якому записано ціле число n ($1 \leq n \leq 99999$). В останньому рядку вхідних даних записано число 0. Для кожного n у виводі запишіть максимальне значення.

- 23) Створити текстовий (.txt) файл з 100,000,000 рядків з числами в діапазоні від 0 до 999,999,999. Формат чисел - 9 знаків ($1 = 000000001$, $65535 = 000065535$), всі числа розташовані в випадковому порядку без повторів (кожен рядок - унікальне число).

Приклад.

000603453

914645283

700500041

035827127.

Розділ 13

Робота з класом рядок на C++

Контрольні запитання:

- Які конструктори для класу рядок? Які для копії-конструкторів? Скільки та які оператори є перевантаженими для класу рядок?
- Як видалити підрядок використовуючи методи класу string?
- Як можна проітеруватись по символах рядку C++?
- Як узнати довжину рядка?
- Як знайти перше входження даного підрядка в рядку? Останнє?
- Як вивести всі слова в реченні, що розділено пробілами? Комами?

Завдання для аудиторної роботи:

В даній групі задач потрібно реалізувати функції та в тих функціях де потрібно виводити рядок зробіть 2 варіанти:

- результат записати в новий рядок;
 - результат замінює рядок, що є аргументом функції.
- 1) В даному рядку знайти підрядок який знаходиться між першою двокрапкою та наступною за нею комою. Якщо немає двокрапки рядок залишати без змін, якщо після двокрапки немає коми - залишити від двокрапки до кінця рядку.
 - 2) В даному рядку видалити підрядок який знаходиться між першою та останньою крапкою. Якщо крапок немає, то видалити в рядку всі пробіли на початку тексту, якщо лише одна крапка, то видалити все до неї.
 - 3) Ввести рядок, що складається з символів латинського алфавіту, слова в якому відокремлені пробілами (одним або декількома). Визначити кількість слів, які починаються і закінчуються однією і тією ж літерою.
 - 4) Ввести рядок, що складається з символів латинського алфавіту, слова в якому відокремлені пробілами (одним або декількома). Перетворити кожне слово в рядку, видаливши з нього останню літеру цього слова (кількість пробілів між словами не змінювати).

- 5) Даний рядок – речення з символів латинського алфавіту. Вивести найкоротше слово в реченні. Якщо таких слів декілька, то: а) вивести перше з них; б) останнє з них; в) всі такі слова.
- 6) Напишіть функцію часткового сплітінгу рядку, тобто функцію, що приймає рядок та повертає перші n слів з рядка (роздільник задається як аргумент функції).

Завдання для самостійної роботи:

- 7) Даний рядок, що складається з символів латинського алфавіту, розділених пробілами (одним або декількома). Вивести рядок, що містить ці ж слова, але розділені одним символом ' ' (кома). В кінці поставити крапку.
- 8) Даний рядок, що складається з символів латинського алфавіту, розділених пробілами (одним або декількома). Перетворити кожне слово в рядку, видаливши з нього всі входження останньої літери цього слова (кількість пробілів між словами не змінювати).
- 9) Речення складається з слів, розділених одним або декількома пропусками або комами. Написати програму, що друкує все слова, що закінчуються на заданий символ.
- 10) Даний рядок, що складається з символів латинського алфавіту, розділених пробілами (одним або декількома). Перетворити кожне слово в рядку видаливши з нього всі входження заданого символу (кількість пропусків між словами не змінювати).
- 11) Даний рядок-речення з символів латинського алфавіту. Перетворити рядок так, щоб кожне слово починалося з великої літери.
- 12) Даний рядок-речення з символів латинського алфавіту. Отримайте найдовше слово в реченні (якщо таких слів декілька, то отримайте масив відповідних слів).
- 13) Визначити, скільки разів в рядку зустрічається задане слово та скільки разів зустрічається заданий підрядок.
- 14) Даний рядок, що складається з символів латинського алфавіту, розділених пробілами (одним або декількома). Визначити кількість слів, які містять введений символ.
- 15) Речення складається з слів, розділених одним або декількома пробілами. Написати програму, що отримає масив всіх слів, що закінчуються на введений символ.

- 16) У англійському реченні слова розділені одним пробілом. У всіх словах, що слідують за словами-артиклями 'a', 'an' та 'the' в даному рядку першу літеру замінити на маленьку.
- 17) Написати програму, що визначає, який відсоток слів в англійському тексті містить подвоєну приголосну.
- 18) У мові використовується латинський алфавіт, причастя завжди закінчується суфіксом "ings". Заданий рядок слів, в якому слова відокремлюються одним або декількома пропусками. Надрукувати всі причастя з даного рядку.
- 19) Даний рядок з малих символів латинського алфавіту. Замініть кожну літеру на наступну за ним за алфавітом, символ 'z' замінити на 'a'. Для великого регістру так само.
- 20) Даний рядок із символів латинського алфавіту. Замініть всі входження рядків "one", "two", "three", ..., "nine" на символи '1', '2', '3', ..., '9'.
- 21) Ввести рядок, що складається з символів латинського алфавіту, слова в якому відокремлені пробілами (одним або декількома). Перетворити кожне слово в рядку, видаливши з нього всі входження передостанньої літери цього слова якщо довжина слова більше ніж 2 (кількість пробілів між словами не змінювати).
- 22) Даний рядок із символів латинського алфавіту. Замініть всі словосполучення вигляду "X's Y" на "Y of X", де X, Y – слова в рядку.
- 23) Визначте, який відсоток символи кожного слова складають з символів даного речення. Рядок – англійський текст.
- 24) Даний текст, що складається з символів латинського алфавіту, пробілів і знаків пунктуації. Знайдіть найпоширенішу голосну букву (інший регістр не утворює нову літеру).

Додаткові задачі:

- 25) Даний рядок в якому зустрічаються слова, які складаються з восьми цифрових символів. Переведіть всі їх у формат дати "dd-mm-yyuu" і перевірте коректність такої дати.

Розділ 14

Робота з текстовими файлами

Контрольні запитання:

- Які типи файлових змінних є в C++?
- Які режими відкриття файлів є в C++? Як змінювати ці режими?
- Як перевірити та проконтролювати коректність відкриття файлів C++?
- Як записати у текстовий файл масив цілих чисел через кому у якості роздільника та прочитати потім цей масив?
- Як з'ясувати кількість рядків в текстовому файлі? Як проітеруватись по всіх словах текстового файлу?

Завдання для аудиторної роботи:

- 1) У текстовому файлі, що складається зі слів, відокремлених одним пропуском, замінити першу літеру у словах, що йдуть за словами *die*, *der*, *das*, на відповідну літеру верхнього регістру.
- 2) У текстовому файлі слова зашифровані – кожне з них записано навпаки. Розшифрувати повідомлення та вивести його в інший файл. Слова розділяються пробілами (довільною кількістю) та знаками коми, крапки, знаками оклику та питання.
- 3) Перевірте у текстовому файлі правильність розстановки тегів `<td>`: кожному відкритого тегу повинен відповідати закритий `</td>`. Теги можуть бути окремими словами, а можуть бути не розділеними від іншого тексту.
- 4) Напишіть функцію, що приймає рядок та повертає масив заданої довжини n (як змінний аргумент) перші n дійсних чисел, що містяться в рядку. Роздільник задається як аргумент функції.

Завдання для самостійної роботи:

- 5) Написати функцію, що створює файл, що складається з N рядків. В кожному рядку записано інформацію про студента - прізвище, курс, група, та деяка кількість цілих чисел через коми - оцінки за іспити. Знайти прізвище студента з найгіршою середньою оцінкою.

- 6) Визначити функцію, що визначає кількість рядків текстового файлу, які:
- починаються із заданого символу;
 - закінчуються заданим символом;
 - починаються й закінчуються одним і тим самим символом;
 - що складаються з однакових символів.
- 7) Написати функцію, що створює файл, що складається з N рядків. В кожному рядку записано інформацію про студента – прізвище, ім'я, група, та деяка кількість дійсних чисел через пробіли – оцінки за іспити. Знайти середню оцінку для всіх студентів даної групи.
- 8) Тег – це слово, яке починається з символу '<', а завершується символом '>'. Обернений тег містить другий символ '/'. Знайдіть всі теги в даному файлі та перевірте чи всі ці теги мають відповідний обернений.
- 9) Реалізуйте наступну гру: користувач вводить своє ім'я та відповідає на деяку кількість запитань вигляду $i \times j = ?$, де i та j – випадкові числа від 1 до 20. Запитання та результат для кожного з гравців запам'ятовуються в файлі у вигляді рядків:
- Vasya 2x2=5 3x4=12 ...
Petya 3x5=15 4x1=4 ...
- І після того, як m гравців зіграли, за допомогою функції `itogi(im'ya файлу)` підраховується й виводиться відсоток правильних відповідей для кожного гравця.

Даний блок задач вимагає організувати роботу з текстовим файлом. Вхідний файл потрібно змінити згідно з вказаними умовами, тобто вхідний та вихідні файли збігаються.

- 10) Дано число N і текстовий файл. Видалити з файлу рядки з номерами, кратними N . Порожні рядки не враховувати і не видаляти. Якщо рядки з необхідними номерами відсутня, то залишити файл без змін.
- 11) Дан текстовий файл, що містить текст, вирівняний по лівому краю (довжина кожного рядка не перевищує 50 символів). Вирівняти його по центру, додавши в початок кожного непорожнього рядку необхідну кількість пробілів. Рядки непарної довжини перед centruванням доповнювати зліва пробілом.

- 12) Напишіть функцію, що створює текстовий файл, який складається з N рядків. Перетворити файл, видаливши в кожному його рядку зайві пробіли (немає пробілів на краях рядків та не більше одного підряд).
- 13) Дан файл з текстом із символів латинського алфавіту. Зашифрувати файл, виконавши циклічний зсув кожної літери вперед на n позицій в алфавіті. Розділові знаки і пропуски не змінювати.
- 14) Дано два текстові файли з іменами Name1 і Name2. Додати в кінець кожного рядка файлу Name1 відповідний рядок файлу Name2. Якщо файл Name2 має менше рядків ніж файлу Name1, то виконайте перехід до початку файлу Name2.
- 15) Знайдіть в текстовому файлі всі цілі числа (послідовність цифр з можливими знаками '+', '-') та замініть їх на відповідні записи у двійковій системі числення. Всі інші слова в файлі залишити без зміни.
- 16) Знайдіть в текстовому файлі всі дійсні числа та замініть всі додатні числа на відповідні їм натуральні логарифми з кількістю символів 10, а від'ємні на значення експоненти довжиною 7 символів. Всі інші слова в файлі залишити без змін.
- 17) Замінити в файлі всі однакові символи, що йдуть підряд на відповідний символ та кількість повторень. *Приклад.* Так, "aaabcc" перетвориться на "a3bc2".

Даний блок задач вимагає організувати роботу з текстовим файлом. Вихідні файли не передбачають зміни. Змінені дані зберігаються в іншому файлі.

- 18) Організувати текстовий файл, що складається з N рядків. Визначити максимальний і мінімальний розмір рядків у файлі та вивести всі мінімальні та максимальні рядки в інший файл.
- 19) Вводиться прописна латинська літера c і текстовий файл. Створити текстовий файл, який містить всі слова з вхідного файлу, що починаються цією літерою (як великою, так і малою). Розділові знаки, розташовані на початках і в кінцях слів, не враховувати. Якщо вхідний файл не містить відповідних слів, залишити вихідний файл порожнім.
- 20) Дано числа N_1 , N_2 і текстовий файл. Видалити з файлу рядки з номерами між N_1 , N_2 , не включаючи меж. Зміни вивести в другий файл. Якщо виконати видалення неможливо, видайте про це повідомлення на екран і в вихідний файл.

- 21) Даний файл з текстом із символів латинського алфавіту, цифр та знаків. Замініть всі цифри їх назвами англійською мовою.
- 22) Створити текстовий файл F, що складається з N рядків. Після цього створити файли H і G. У файл H записати рядки файлу F непарної довжини, а в файл G парної довжини.
- 23) Визначити функцію, яка:
 - підраховує кількість порожніх рядків;
 - обчислює максимальну довжину рядків текстового файлу.
- 24) Визначити процедуру виведення:
 - усіх рядків текстового файлу;
 - рядків, які містять понад 60 символів.
- 25) В даному текстовому файлі знаходиться англійський текст. Вирівняйте його по лівій та правій границі так, щоб розподіл слів у рядках був найбільш рівномірним та запишіть результат в інший файл.
- 26) Визначити процедуру, яка переписує до текстового файлу G усі рядки текстового файлу F:
 - із заміною в них символу '0' на '1', і навпаки;
 - кожне слово в інвертованому вигляді.
- 27) Визначити процедуру, яка переписує компоненти текстового файлу F до файлу G, вставляючи до початку кожного рядка один символ пропуску. Порядок компонент не має змінюватися.

Додаткові задачі:

- 28) В текстовому файлі записані в кожному рядку значення поліномів за допомогою знаків +, -, *, ** (ступінь) та цифр і літери x . Введіть значення x з консолі та для всіх коректних записів поліномів обчисліть їх значення для даного x та виведіть в новий текстовий файл.

Розділ 15

Створення власних класів. Інкапсуляція.

Контрольні запитання:

- Що таке класи і які шляхи визначення класів в C++?
- Яким чином можна визначити методи класу?
- Приватний та публічний доступ до членів та методів. Яка різниця?
- Які методи в класі визначені за замовченням? Як і коли потрібно ці методи визначати самостійно?
- Шляхи визначення конструктору класу. Як викликати конструктор в головній функції?
- Статичні члени та методи класу. Як визначити і коли вони потрібні?
- Дружні класи та методи. Як вони використовуються?

Завдання для аудиторної роботи:

- 1) Визначити клас раціональне число з членами: *nominator* – ціле число, *denominator* – натуральне число. Визначить наступне:
 - методи введення та виведення з терміналу;
 - методи додавання та множення раціонального числа;
 - зробіть члени класу приватними та визначить методи ініціалізації окремо чисельника і знаменника (при цьому не дайте користувачу можливість ініціалізувати знаменник нулем);
 - створіть приватний метод класу для скорочення раціонального числа через НСД;
 - визначить конструктор класу який ініціалізує за замовченням раціональне число одиницями та конструктор, що ініціалізує його двома довільними числами;
 - також у класі перевантажте основні арифметичні оператори, оператори порівняння та інші оператори, що необхідні для роботи з раціональними числами.

Використовуючи цей клас, розв'яжіть такі задачі:

- знайдіть найменше раціональне число в масиві раціональних чисел;
- підрахуйте суму ряду за формулою з заданою точністю 0.01 та перевірте, чи дійсно цей ряд сходиться до відповідного числа:

$$\frac{\pi^2}{12} = 1 - \frac{1}{2^2} + \frac{1}{3^2} - \frac{1}{4^2} + \dots$$

- 2) Створіть клас **Точка**, який містить статичний член - лічильник екземплярів класу та напишіть програму, що дозволяє вводити багатокутник з будь-якої кількості вершин вводячи точки доки користувач не відповість на запитання «Ввести вершину?» - «Ні». Після цього використовуючи статичний метод `counter()` виведіть інформацію про кількість вершин у багатокутнику та виведіть його периметр.
- 3) Визначить клас **Поліном**, що ініціалізується кількістю елементів масиву `N` та виділяє при цьому пам'ять під `N` дійсних чисел. Створіть методи для заповнення членів цього масиву (через конструктор та окремим методом) та конкретного коефіцієнта за номером, а також метод виведення поліному. Визначить деструктор та копі-конструктор. Який оператор також бажано перевантажити?

Визначить дружні функції для цього класу для введення та виведення його з консолі у бінарний файл.

Завдання для самостійної роботи:

Описати класи розділивши інтерфейс та реалізацію та заборонивши введення некоректних даних, з методами введення/виведення та іншими:

- 4) Описати клас **Точка** на площині. Реалізуйте методи введення, виведення. Описати клас **Відрізок** на площині, що складається з 2-х точок та містить крім введення/виведення методи підрахунку середини відрізка, довжини відрізка. За допомогою визначення порожньої Точки реалізуйте метод перетину двох відрізків, що повертає Точку (у випадку, якщо цих точок декілька виведіть будь-яку з них, а якщо жодної – порожній відрізок). Описати клас **Трикутник** з методами введення/виведення, периметру та площі.
- 5) Описати клас **Коло** на площині, що задається координатами центру та радіусом. Описати методи отримання довжини діаметру, площі та периметру кола, перетину двох кіл (повертає відповідно 0,1 або 2 точки як масив через змінний аргумент). Введіть в програмі декілька екземплярів класу та зробіть можливість в будь-який момент вводу

нового кола чи знищення попереднього вираховувати центр мас цих кіл.

- 6) Описати клас **Прямокутник**. Сторони прямокутника паралельні осям координат. Для прямокутника задані координати лівого верхнього кута та довжини сторін. Описати методи отримання довжини кожної зі сторін, площі та периметру, перетину двох прямокутників (якщо перетин порожній – поверніть Прямокутник вигляду $(-1,-1,-1,-1)$).
- 7) Описати клас **Трикутник**. Основа трикутника паралельна вісі x координат. Для трикутника задані верхній кут та довжини бічних сторін. Описати методи отримання довжини кожної зі сторін, кутів, площі та периметру.
- 8) Описати наступні класи з методами визначення різниці між сутностями одного класу:
 - а) **Час** (години, хвилини, секунди);
 - б) **Дата**(рік, місяць, день). Клас **Дата** створіть так, щоб в програмі він міг бути визначеним лише один раз.
- 9) Описати клас ігрова **Дошка**(визначається розміром та назвою гри: шашки (міжнародні, російські та турецькі, шахи, нарди) та **Фігура** (назва, гра, масив можливих ходів – ходи описуються в термінах зрозумілих класу Дошка).
- 10) Описати класи з методами введення/виведення та додавання і різниці при однаковій назві:
 - а) **Валюта** (назва валюти, значення, центи (копійки));
 - б) **Товар** (назва товару, вартість, валюта в який вимірюється вартість, одиниця в який вимірюються товар).

Реалізуйте для обох класів дружні функції обміну валюти за даним курсом.

- 11) Створити клас **Book** (Книжка – назва, автор, кількість сторінок, рік видання) та реалізувати програму пошуку книжки за авторами та назвою в каталозі (каталог – масив книжок, що зберігається у файлі).
- 12) Визначить клас **Вектор**, що ініціалізується кількістю елементів масиву N та виділяє при цьому пам'ять під N дійсних чисел. Створіть методи для заповнення членів цього масиву (через конструктор та окремим методом) та конкретного елементу вектору за номером.

Визначити деструктор та копі-конструктор. Із використанням динамічних масивів розв'язати задачу: у двох масивах містяться коефіцієнти векторів степеню m і n відповідно. Написати методи для введення/виведення таких векторів з файлу, скалярного та векторного добутку (за можливості) для цих векторів, або змістовного повідомлення, чому така операція неможлива.

- 13) Опишіть класи **Matrix3** та **Vector3**, що є відповідно матрицею розмірності 3×3 та тривимірним вектором. Перевантажте математичні оператори для цих класів та спеціальні методи (множення матриці на вектор у тому числі). Функцію `abs()` визначте для матриці та вектору як визначення норми. Для матриці опишіть метод `det()`, що повертає визначник цієї матриці.
- 14) Створіть клас для реалізації гри «Хрестики-нолики», який має наступні методи:

- малювання початкового стану за допомогою символів '|' та '_';
- малювання символу в даному полі за допомогою символів пробілу, 'O' та 'X';
- приймання ходу гравця з клавіатури (з перевіркою коректності вводу, унеможливленням введення гравцем некоректного ходу та можливістю виходу з гри);
- перевірка на те що гра закінчилася та визначення результату гри.

В головній програмі розіграйте партію для перевірки даних методів.

- 15) Опишіть два класи:

- **Дата**, що містить ціле число, яке представляє будь-яку дату (наприклад, як різниця від дати до 1 січня цього року).
- **Гість**, що містить всю необхідну інформацію про жителя деякого готелю: Прізвище, дату заселення та виписки, номер в готелі тощо.
- **Готель**, що містить масив номерів готелю, вартість кожного з них і т.п.

Використовуючи вищенаведені класи розв'язати задачу:

- відомість про кількість вільних кімнат у готелі в дану дату;
- пошуку вільної кімнати у зазначений період;

- вартості проживання даного жильця у зазначений період;
 - виведення номера кімнати гостя у готелі (у заданий період).
- 16) Визначити клас **Квадратне рівняння**. Реалізувати методи для пошуку коренів, екстремумів, а також інтервалів убування / зростання. Створити масив об'єктів і визначити найбільші і найменші значення коренів.
- 17) Визначити клас **Інтервал** з урахуванням включення/невключення країв. Створити методи по знаходженню перетину і об'єднанню інтервалів, причому інтервали, що не мають спільних точок, перетинатися /об'єднуватися не можуть. Створить масив з n інтервалів і визначить відстань між найбільш віддаленими кінцями.
- 18) Визначити клас **Точка** на площині в часі (координати точки, швидкість та прискорення по кожній з координат). Створити методи по встановленню та виведенні швидкостей та прискорення точки в момент часу t . Перевірити для двох точок можливість перетину траєкторій в проміжку часу $[t_1, t_2]$. Визначити відстань між двома точками в заданий момент часу. Ввести масив точок та підрахувати кількість всіх перетинів траєкторій за даний проміжок часу.

Додаткові задачі:

- 19) Доповніть задачу 3) методами ініціалізації через рядок та текстові й бінарні файли. Реалізувати методи: введення поліному, виведення поліному, обчислення значення поліному у точці x , взяття похідної поліному, суми, різниці та добутку поліномів. Використати цей клас для розв'язання задачі: ввести два поліноми P_1 , P_2 та рядок, який містить вираз, що залежить від двох поліномів (наприклад, $P_1 - P_2 * (P_1 + P_2)$). Обчислити поліном, який буде значенням цього виразу.

Розділ 16

Робота з класами. Наслідування та поліморфізм.

Контрольні запитання:

- Що таке перевантаження методів? Чому воно зручно в мовах зі строгою типізацією?
- Чим перевантаження операторів відрізняється від перевантаження інших методів?
- Які оператори не можна перевантажувати? Коли перевантаження операторів може бути небезпечним?
- Чому при перевантаженні операторів вводу-виводу нам потрібно ключове слово `friend`?
- Які типи наслідування є на C++ та яка між ними різниця?
- Поясніть на прикладі, що таке раннє та пізнє зв'язування
- Що таке чисто віртуальний клас та чисто віртуальний метод? Коли вони потрібні?
- Що таке віртуальний деструктор, та чому він потрібний?
- Як реалізувати множинне наслідування на C++?
- Що робити та які шляхи правильного множинного наслідування якщо й класи батьки й клас-нащадок мають метод з однаковою назвою? Що зміниться, якщо це не метод, а перевантажений оператор?

Завдання для аудиторної роботи:

- 1) Клас `Person` описано таким чином:

```
class Person{ //Клас Особа
    string name; //прізвище
    unsigned byear; //рік народження
public:
    int input(); //ввести особу
    void show(); //вивести особу
};
```

Реалізуйте запропоновані методи (можете додати ще власних) та зробіть для класу переважання стандартних операторів вводу-виведення.

Описати клас Знайомий на базі класу Person. У цьому класі повинно бути як мінімум одне додаткове поле «номер телефону», а також методи введення та виведення інформації про знайомого. Використати цей клас для побудови класу телефонного довідника (кількість знайомих обмежена числом 100). Передбачити дії: створення довідника, додавання запису про знайомого, пошуку номера телефону за прізвищем та заміни номера телефону. Телефонний довідник зберігає дані про знайомих у файлі. Вказівка: телефонний довідник представити у вигляді класу, що зчитує дані з текстового файлу.

- 2) На базі класу Точка на площині створіть клас Точка3Д (точка в просторі. Реалізуйте методи введення, виведення. Реалізуйте методи за допомогою даних класів клас Відрізок, що дозволяє працювати з точками розмірностей 2 та 3. введення та виведення, визначення довжини відрізка. Вказівка: визначіть додатковий член "розмірність" в класах Точка, Точка3Д. Додаткова задача – визначення чи перетинаються два відрізки.

Завдання для самостійної роботи:

- 3) Описати клас Пасажир на базі класу Person. Клас містить дані про місце відправлення та місце слідування, а також місце пасажира. Створіть клас Каса, який дозволяє додавати та виводити інформацію про Пасажирів, містить методи пошуку по прізвищу, місцям відправлення, прибуття та місцю. Також серед заданого масиву місць у потягу знайдіть місце яке не зайняте (у випадку якщо таких місць декілька – виведіть найменше за значенням, якщо їх немає відповідне повідомлення).

Вказівка: інформацію про пасажирів представити у вигляді бінарного файлу.

- 4) Описати клас Студент на базі класу Person. У класі Студент повинна бути інформація про оцінки отримані ним протягом сесії (за 5-ти бальною та 100 бальною шкалами). Скласти програму для обчислення нарахованої студентам стипендії в залежності від результатів сесії:

- За старим підходом нарахування стипендії (середній бал за всі іспити має бути не меншим ніж 4 за 5-ти бальною шкалою).

- З новим підходом нарахування стипендії (стипендію отримують 40% від загального числа студентів, які є найкращими по рейтингу)

Вказівка: інформацію про студентів представити у вигляді масиву. Дані зчитувати з клавіатури.

- 5) Реалізувати клас **Слово**, який має члени типу Рядок: Приставка, Приставка2, Корінь, Суфікс, Закінчення (клас повинен мати гетери та сетери).

Створіть наслідники цього класу: **Дієслово**, **Іменник**, **Прикметник**.

Реалізуйте для них методи: Род, Число, Лице, Відмінок – які будуть відповідним чином змінювати (якщо це можливо) дане слово.

```
Nominativ n("house"); n.plural();// "houses";
```

```
Verb v("go"); v.third();// "goes"
```

```
v.plural(); v.third();// "go"
```

Створіть декілька слів, що є екземплярами Дієслово, Іменник, Прикметник та виконайте відповідні методи для них, щоб можна було побачити результат.

- 6) Реалізувати наступні класи:

Створити клас **Фігура**, який є базовим.

- Описати клас **Прямокутник**. Сторони прямокутника паралельні осям координат. Для прямокутника задані лівий верхній кут та довжини сторін. Описати методи отримання довжини кожної з сторін, площі прямокутника, периметру, чи перетинаються 2 прямокутники, координати центру мас.
- Описати клас **Трикутник**. Основа трикутника паралельна осі x координат. Для трикутника задані ліва нижня координата, довжина основи та 2 кути спільні з основою. Описати методи отримання довжини кожної зі сторін. Описати методи отримання площі, периметру, координати центру мас.
- Описати клас **Еліпс**. Для нього є заданими координати фокусів та радіуси. Описати методи отримання геометричних характеристик. Описати методи отримання довжини радіусів, площі, периметру, координати центру мас.

Скласти програму створення заданої кількості фігур та знаходження їх спільного центру мас.

7) Створити клас **Фігура**, який є базовим. Опишіть класи для таких геометричних фігур та реалізуйте зазначені методи:

- Клас **Трапеція**. Основи трапеції паралельні осі Ox . У цьому класі реалізуйте операції знаходження периметра і площі, методи переміщення та повороту.
- Клас **Паралелограм**. Основи паралелограму паралельні осі Ox . У цьому класі реалізуйте операції знаходження периметра і площі, методи переміщення та повороту.
- Клас **Круг**. Реалізуйте методи відшукування площі круга, довжини кола, методи переміщення та повороту.

Скласти програму створення заданої кількості фігур, їх переміщення так щоб в них не було перетинів та знаходження їх сумарної площі та периметру. Знайдіть фігуру з найбільшою площею.

8) Створити клас **Фігура**, який є базовим. Опишіть класи для таких геометричних фігур та реалізуйте зазначені методи:

- Клас **Прямокутник**. Для прямокутника задані лівий верхній кут та правий нижній кут. Описати методи отримання довжини кожної з сторін, площі прямокутника, периметру.
- Клас **Трикутник**, що містить масив з трьох вершин. Описати методи отримання довжини кожної з сторін, площі прямокутника, периметру.
- Клас **П'ятикутник**, що містить масив вершин. Реалізуйте метод перевірки чи є цей п'ятикутник опуклим.
- Клас **Багатокутник**. Реалізуйте метод перевірки чи є цей багатокутник опуклим.

Дано масив фігур вищенаведених класів. Знайдіть всі опуклі багатокутники. Знайдіть в цьому масиві фігуру, що має найменший периметр.

9) Створити клас **Фігура3D**, який є базовим. Опишіть класи для таких геометричних фігур та реалізуйте зазначені методи:

- Клас **Паралелепіпед**. Реалізуйте методи пошуку площі бічної поверхні і об'єму.
- Клас **Піраміда3(трикутна)**. Реалізуйте методи пошуку площі бічної поверхні і об'єму.

- Клас **Піраміда4**(прямокутна). Реалізуйте методи пошуку площі бічної поверхні і об'єму.

Введіть масив фігур та підрахуйте їх сумарний об'єм та сумарну площу всіх граней та загальну кількість вершин.

- 10) Створити клас **Лінійне рівняння** для лінійного рівняння з методом пошуку дійсного розв'язку. Створити клас **Квадратне рівняння** для квадратного рівняння — нащадок першого класу, з методом пошуку дійсних розв'язків. Створити клас **Біквадратне рівняння** для біквадратного рівняння — нащадок другого класу, з методом пошуку дійсних розв'язків. В усіх класах передбачені методи введення/виведення та завдання відповідно двох та трьох дійсних коефіцієнтів. Введіть масив рівнянь з текстового файлу та знайдіть:

- всі рівняння, що мають нескінчену кількість розв'язків;
- кількість рівнянь, що не мають дійсних розв'язків;
- найменший за модулем розв'язок;
- суму квадратів всіх дійсних розв'язків.

- 11) Опишіть клас **Машина**, що має метод `go(distance)`, який змінює пройдений кілометраж автомобілем та залишок пального. Метод `go(...)` залежить від віртуального методу `fuelPerKm()`, який визначає скільки потрібно пального автомобілю для проїзду одного кілометра. Нехай `Personal` (легковий автомобіль) і `Truck` (вантажівка) — класи, що наслідують клас **Машина** і перевизначають метод `fuelPerKm()`. При цьому потрібно врахувати, що цей метод залежить від кількості пасажирів (+10% на кожного пасажирів) для авто класу `Personal` або ваги вантажу для `Truck` (+25% на кожну тонну вантажу). Визначити чи зможе задане авто проїхати задану відстань.

- 12) Визначити клас **Рівняння** для однієї змінної. Клас дозволяє задавати інтервал, де шукається корінь та має метод для знаходження кореня. Створити нащадки цього класу: лінійне рівняння, кубічне рівняння, синус, експоненціальне рівняння, які дозволяють ввести параметри та коефіцієнти таких типів рівнянь. Реалізувати метод визначення коренів методом бісекції або іншими в різних класах. Реалізуйте відповідні методи відображення таких рівнянь. Введіть масив рівнянь та:

- виведіть всі рівняння, що не мають дійсних розв'язків;
- найбільший розв'язок;

- чи є інтервал, на якому у всіх рівнянь є хоча б один дійсний розв'язок;
 - суму всіх дійсних розв'язків.
- 13) Визначить базовий клас **Товар** (назва, артикул, одиниця виміру, вартість, дата постачання товару) та відповідні нащадки: **Іграшки**(вікові обмеження), **Їжа**(час придатності), **Техніка**(наявність гарантії, час гарантії). Створіть бінарний файл з товарами та методи:
- пошуку даного товару(по назві та по типу): виводити чи є даний товар, та якщо є – список всіх товарів, що було знайдено;
 - оформлення замовлення (вибір декількох товарів, підрахунок їх сумарної вартості та видалення заказаних товарів з файлу);
 - зниження вартості товарів, термін придатності чи часу гарантії на них менше ніж 5 днів на 20%.
- 14) Створіть клас **Адреса**, що містить рядкові поля Місто, Вулиця, та числові номер дома та квартири. Створять від нього нащадку **Міжнародна адреса**, що додає також до класу рядкові поля країна та поштовий код. Введіть масив адрес та знайдіть найпопулярніше місто в даних адресах для якого також було введено як **Міжнародна адреса**. Запишіть у текстовий файл всі адреси з цим містом доповнивши всі адреси що були введені без міжнародних даних за допомогою відомостей, що дало введення міжнародної адреси для цього міста.
- 15) За допомогою класу **Адреса**, що містить рядкові поля Місто, Вулиця, та числові номер дома та квартири створіть клас-нащадок класу **Person**, що містить ці дані. Окрема створіть клас **ЕАдрес**, що містить електронну пошту, адресу сторінки (може бути порожньою) та телефон. Зробіть можливим використання нового класу як з першим варіантом, так і з другим. Створіть бінарний файл з екземплярами цього класу. Знайдіть всіх людей, що живуть в одному місті та мають однаковий домен електронної пошти.
- 16) Створіть абстрактний клас **Число** з методами введення/виведення, додавання, множення, ділення. Створіть класи **Раціональне число** та **Комплексне число** як нащадки цього класу. За допомогою даних класів створіть функцію введення поліному від таких чисел та обчисліть їх значення в даній числовій точці.

Розділ 17

Перетворення типів та робота з виключеннями

Контрольні запитання:

- Які варіанти перетворень стандартних типів один між іншим можливі в C++?
- Яким перетворенням краще скористатись для перетворень між цілими типами? Яким при перетворенні цілих до дійсного та навпаки?
- Чим відрізняються перетворення вгору та вниз? Яке перетворення типу краще для перетворення вгору, а яке вниз?
- Чому не можна відловити виключення при діленні на нуль в C++ зі стандартними типами?
- Як створити власне виключення в C++? Як його коректно обробити?
- Яке стандартне виключення дозволяє коректно обробити `static_cast`?
- Як складнощі виникають якщо виключення виникає в деструкторі класу?
- Як коректно працювати з виключенням, що виникає в конструкторі класу?

Завдання для аудиторної роботи:

- 1) В класі Раціональній дріб з попередніх лекцій перезавантажте методи введення, виведення (`cin>>`, `cout<<`) та оператори віднімання, ділення як перевантажені оператори. Тобто з типом Раціональній дріб можна тепер працювати як зі стандартним типом. Чому краще перевантажити два оператори віднімання? Перепишіть методи введення (`cin>>`) та конструктор і сеттери, щоб вони викликали виключення при ініціалізації знаменника нулем. Коректно обробить в коді це виключення. Напишіть дружню функцію запису Раціонального дробу в файл, яка буде викидати виключення при некоректному відкритті файлу та обробить його в тілі програми.
- 2) Створіть клас Людина (члени: ПІБ, стать, вік) та його нащадки Студент (додано: курс, група, ВУЗ), Викладач (додано: ВУЗ, посада,

з.п.). Методи введення, виведення, конструктори для різної кількості вхідних даних. Створіть клас Аспірант, що є нащадком і студента і викладача. Коректно визначте член ВУЗ для нього.

Створіть програму що буде вводити масив Людей, серед яких є Студенти, Викладачі, Аспіранти. Без створення нових членів класу виведіть коректно ВУЗ для кожного екземпляру масиву. Забезпечте обробку помилок для коректного вводу людей.

Завдання для самостійної роботи:

- 3) Скласти функцію для обчислення значення натурального числа за заданим рядком символів, який є записом цього числа у системі числення за основою b ($2 \leq b \leq 16$). Використати функцію, яка за заданим символом повертає відповідну цифру у системі числення за основою b . Використати у цій функції твердження про стан програми `assert` для перевірки того, що відповідний символ є цифрою у системі числення за основою b . Обробити помилку неправильного символу рядка та показати змістовне повідомлення про помилку створивши власне виключення.

- 4) Скласти власний клас для комплексного типу з методами введення/виведення та арифметичними операціями. Напишіть функцію для обчислення суми всіх доданків, модуль яких не менше $\varepsilon \geq 0$, у комплексній точці z :

$$\arctg(z) = z - \frac{z^3}{3} + \frac{z^5}{5} - \dots + (-1)^n \frac{z^{2n+1}}{2n+1} + \dots, \quad (|z| < 1).$$

Використати у цій функції твердження про стан програми для перевірки того, що параметр z відповідає заданій умові та зробить обробку всіх можливих виключень – включаючи некоректне введення та виділення пам'яті під масиви. Обробити у програмі помилку неправильного значення z та показати змістовне повідомлення про помилку.

- 5) Описати клас **Трьохбайтне ціле число** для роботи з цілими числами, представленими трьома байтами. Інтервал представлення при цьому від -2^{23} до $2^{23} - 1$. Зробіть методи та конструктор вводу, що обробляють введене ціле число та кидають виключення при некоректному вводі та перезавантажте арифметичні дії. Арифметичні дії не повинні дозволяти переповнення інтервалу представлення, тобто $2^{23} - 1 + 1$ – це помилка, і якщо результат операції виводить за межі інтервалу представлення, повинно ініціюватися відповідне виключення. Перевизначити у цьому класі операції $+$, $-$, $*$, $/$ (цілочисельне). Описати також три класи обробки помилок для трьохбайтних цілих

чисел: загальний клас обробки помилок та два його підкласи для обробки помилки переповнення та помилки ділення на 0.

Використати цей клас для розв'язання задач:

- обчислення $\sum_{k=1}^m x_k$, де x_k – масив **Трьохбайтних цілих чисел**, n – натуральне;
- обчислення x^n , де x – **Трьохбайтне ціле число**, n – натуральне.

Забезпечити обробку помилок при виконанні обчислень.

- 6) Створіть клас для роботи з бінарними файлами, в яких записані цілі числа. В класи визначені члени: ім'я файлу, кількість чисел у файлі. Реалізуйте методи, введення чисел з консолі в файл, створення файлу з масиву чисел, виведення змісту файлу на консоль, повернути число за даним номером, додавання до файлу масиву чисел в кінець, видалення числа за даним номером. Забезпечити обробку помилок при роботі з файлами. Створіть відповідні виключення для проблем при створенні файлу, проблем при читанні з файлу, некоректних номерах чи кількості чисел.
- 7) Створіть клас для роботи з текстовими файлами, в яких записані дійсні числа які розділяються пропусками в одному рядку та можуть бути розташовані у різних рядках. В класи визначені члени: ім'я файлу, кількість чисел у файлі, кількість рядків файлу. Реалізуйте методи:
- введення чисел з консолі в файл рядок за рядком;
 - створення файлу з двовимірною масиву чисел;
 - виведення змісту файлу на консоль, повернути число за даним номером;
 - додавання до файлу масиву чисел в кінець новим рядком;
 - видалення числа за даним номером рядку та місцем в ньому.

Створіть відповідні виключення для обробки проблем при створенні файлу, проблем при читанні з файлу, некоректних номерах чи кількості чисел. Забезпечити обробку помилок, якщо у файлі, що читаються, зустрічаються не дійсні числа.

- 8) Описати клас Поліном, що заданий ступенем та масивом дійсних коефіцієнтів та реалізувати методи: введення поліному з консолі та рядку, виведення поліному, обчислення значення поліному у точці x , взяття похідної поліному, суми, різниці та добутку поліномів.

Описати також клас обробки помилок при неправильному введенні поліному (ступінь – не невід’ємне ціле число, коефіцієнт – не дійсне число) та забезпечити ініціювання помилки при викликах всіх методів з некоректними даними. Забезпечити обробку помилок неправильно-го введення поліному в основній програмі.

- 9) Створіть клас роботи з рядком, який має наступну властивість: користувач задає власноруч допустиму множину символів, з яких може складатись цей рядок у вигляді масиву символів. Члени класу: масив допустимих символів та його довжина, масив введених символів та його довжина. Методи класу:

- перезавантажте методи введення/виведення в/з консолі та в/з текстового файлу;
- методи зміни(додавання/видалення) допустимих символів;
- довжина рядку;
- конкатенація рядків (при цьому допустимі символи — це перетин множин допустимих символів, тобто після конкатенації в нас може зменшитися результатний рядок);
- хеш рядку (ваш будь-який розумний варіант хешу).

Забезпечити ініціювання помилки при неправильному введенні та роботі з рядками та роботі з файлами.

- 10) Реалізуйте клас Вектор, що ініціалізується кількістю елементів масиву n та виділяє при цьому пам'ять під n дійсних чисел. Створіть методи для заповнення членів цього масиву (через конструктор та окремим методом) та конкретного елементу вектору за номером. Написати методи для введення/виведення таких векторів з файлу, скалярного та векторного добутку (за можливості) для цих векторів та обробіть за допомогою виключень проблеми з введенням та арифметичними операціями та методами доступу над векторами. Також спробуйте врахувати можливі проблеми з пам'яттю.

Розділ 18

Створення шаблонів функцій та шаблонів класів

Контрольні запитання:

- Як створити функцію-шаблон? В яких ситуаціях вона корисна?
- Як створити клас-шаблон? Що потрібно зробити якщо шаблоном є лише єдиний метод класу?
- Навіщо потрібні простори імен та що таке стандартний простір імен? Як його підключити та що робити коли не можна його підключати на весь файл програми?
- Як створити власний простір імен що містить власні математичні функції \sin , \cos , \exp . Як їх коректно використати разом зі стандартними функціями?
- Створіть вкладені простори імен та функції з однаковими ідентифікаторами в них та функцію з таким самим ідентифікатором глобально. Як правильно використати ці функції використовуючи ключове слово `using`? Як без нього?

Завдання для аудиторної роботи:

В завданнях цього циклу створіть власний простір імен та в цьому просторі потрібні функції та класи.

- 1) Перепишіть функцію шаблон для пошуку максимуму двох чисел, так щоб вона працювала для всіх стандартних числових типів. Чи працює вона для рядків? Що потрібно зробити, щоб вона працювала і для типу Рационального дробу з попередніх лекцій? (Вказівка: щось потрібно визначити для класу Рациональний дріб)
- 2) Написати функцію, що вводять масив цілих чисел доки не буде введений нуль та повертає результат через змінний аргумент та кількість елементів масиву повертається як результат роботи функції. Для невідомої заздалегідь кількості елементів потрібно робити реалізацію стеку. Створіть власну реалізацію класу шаблону `Stack` для будь-якого типу. Перевірте її роботу за допомогою стандартного класу `Stack` з STL для даної задачі та іншого типу чисел що вводяться.

Завдання для самостійної роботи:

- 3) Створити клас-шаблон `BlackBox`, який містить конструктор (порожній та від масиву (вказівника) будь-якого типу), метод `push()`, що дозволяє додати елемент певного типу, та метод `pop()`, що видає та видаляє випадковий елемент, що вже міститься в класі та виключення, якщо `BlackBox` порожній, метод `хрор()`, що просто повертає випадковий елемент цього класу. Кількість елементів обмежена 100.
- 4) Створити клас-шаблон `Mediana`, який містить конструктор (порожній та від масиву (вказівника) будь-якого типу), що містить операції порівняння, метод `push()` який дозволяє додати елемент будь-якого типу, що містить операції порівняння, метод `pop(int n)`, що видає та видаляє елемент за номером n за порядком, або виключення якщо n більше розміру всіх елементів та метод `mediana()`, що повертає медіану елементів цього класу. Кількість елементів обмежена 100.
- 5) Визначити клас `Масив`, який містить розмір масиву та відповідний масив даних довільного типу.

Реалізувати в ньому методи сортування як для самого масиву та як статичні методи (`inplace`):

- а) обмінне сортування (метод бульбашки);
 - б) обмінне сортування «Шейкер-сортування»;
 - в) сортування за допомогою вибору (метод простого вибору);
 - г) сортування вставками;
 - д) сортування методом хешування (сортування з обчисленням адреси);
 - е) сортування вставками (метод простих вставок);
 - є) сортування бінарним злиттям;
 - ж) сортування Шелла (сортування зі спадним кроком);
 - з) швидке сортування;
 - і) сортування купою.
- 6) Створіть клас раціональне число на базі шаблону пари для довільних типів знаменника та чисельника. Перевантажте методи для всіх арифметичних операцій та порівнянь (зокрема, остача від ділення – це ділення після якого видаляється ціла частина). Зробіть наступну спеці-

алізацію, якщо знаменник або чисельник – рядок: створюється рядок вигляду "чисельник /знаменник" з виключеннями на всі арифметичні операції, крім додавання (для нього – це конкатенація), але коректною роботою з порівнянням/введенням/виведенням/доступом.

7) Створіть клас рядок, що приймає у якості символу будь-який тип (зокрема інший рядок) та роздільник(того самого типу) - що відокремлює в запису ці символи. Методи класу:

- переавантажте методи введення/виведення в/з консолі та в/з текстового файлу;
- введення та заміна роздільника;
- метод конкатенації (з додаванням між рядками роздільника);
- довжина рядку;
- злиття символів – тобто перетворення масиву символів на єдиний символ типу рядок;
- доступ до даного символу за квадратними дужками;
- видалення даного символу.

Забезпечити ініціювання помилки при неправильному введенні та роботі з рядками та роботі з файлами та спеціалізацію як звичайний рядок при символі типу char.

8) Визначити клас Інтервал с урахуванням включення/невключення країв та нескінченості на інтервалах, на базі шаблону пара. Якщо тип на одному з країв – рядок, то вважається що це відповідна нескінченість. Створити методи по знаходженню перетину і об'єднанню інтервалів, причому інтервали, що не мають спільних точок, перетинатися /об'єднуватися не можуть. Створіть масив з n інтервалів та знайдіть їх спільний перетин.

9) Реалізуйте функцію `sum(T* x, size_t n)`, яка рахує суму будь-якого масиву, що передається їй як аргумент. При цьому тип `char` сумується як символ, а тип вказівник вважається масивом розміру 1 та сумується утворюючи масив розміру n (нульові вказівники просто ігноруються в додаванні):

```
int v1[] = { 1, 2, 3 }; // sum(v1,3) =6
double v2[] = { 1, 2, 3 }; //sum(v2,3) =6.0
string v3[] = { "a", "bc", "def" }; // sum(v3,3) ="abcdef"
char v4[] = { 'a', 'b', 'c' }; // sum(v4,3) ="abc"
```

```
int* v5[] = { {1,4}, {2}, {3} }; // sum(v5,3) = {1,2,3}
```

- 10) Визначить клас Функція. Клас дозволяє задавати інтервал де шукається корінь та створювати функцію від ступнів дійсних чисел та від функцій косинус, корінь та логарифм. Створити методи для обчислення значення за формулою лівих прямокутників, за формулою правих прямокутників, формулою середніх прямокутників, по формулі трапецій, по формулі Сімпсона (параболічних трапецій).

Створіть метод для семплювання функції – задаються межі інтервалу та кількість семплів на інтервалі, обчислюються дискретні значення в даних точках і будується й виводиться таблиця, що містить пари точки - значення.

Розділ 19

Стандартна бібліотека C++. Послідовні контейнери та контейнери-адаптори.

Контрольні запитання:

- Створіть власний клас-шаблон `vector<T>` з методом `Норма()`. Порівняйте його дію з стандартним шаблоном `vector` в головній програмі.
- З яких частин складається бібліотека шаблонів C++?
- Для чого потрібні контейнери-адаптори? Які контейнери-адаптори визначені в C++?
- Які контейнери прямого доступу визначені в C++?
- Яка різниця між контейнерами `list`, `forward_list`, `vector`, `array`?
- Основні методи контейнеру вектор (доступ до елементів, заміна елементів, розміри)?
- Які переваги `array` або `vector` перед стандартним масивом чи вказівником?
- Як додавати елемент в вектор, стек, список?
- Як видаляти елементи в `list`, `forward_list`, `vector`, `array`?
- Які варіанти проітеруватись по елементах послідовних контейнерів?
- Як визначити кількість елементів будь-якого контейнеру?
- Які коректні шляхи ітерації по вектору? Мультивідображенню? Будь-якому контейнеру?
- Як коректно пройти по всім елементам відображення?

Завдання для аудиторної роботи:

- 1) Біля прилавка в магазині вишикувалася черга з n покупців, кожен з яких ставав за попередником через 1 хвилину після як той ставав в чергу. Час обслуговування продавцем i -го покупця – t_i , $i = 1, \dots, n$ хвилин. Нехай дано натуральне n і дійсні числа t_1, t_2, \dots, t_n . Отримати c_1, c_2, \dots, c_n де з c_i – час перебування i -го покупця в черзі

- $i = 1, \dots, n$. Вказати номер покупця, для обслуговування якого продавцеві потрібно найменше часу, та покупця, що провів в черзі найбільше часу.
- 2) Реалізувати функції для введення d -вимірних векторів (d вводиться з клавіатури). Ввести n d -вимірних векторів та обчислити значення суми норм векторів.
 - 3) Створіть клас-шаблон Поліном, який приймає список чисел будь-якого типу на базі стандартного класу list коефіцієнтів поліному. Методи: введення-виведення, додавання, множення та обчислення значення. Перевірте, що клас працює коректно для дійсних, цілих чисел та для типу Раціональний дріб з попередніх завдань.
 - 4) Ввести натуральні числа $n, m < 100$ та матрицю розміру $n \times m$ як вектор списків. Поміняти місцями її рядки так, щоб їх максимальні елементи утворювали спадну послідовність.

Завдання для самостійної роботи:

- 5) Дана матриця $n \times m$ з цілих чисел. Знайти в ній найбільшу за площею 4-зв'язну область, що складається з однакових елементів, використавши клас stack для зберігання координат цієї області.
- 6) Реалізувати структуру «чорний ящик» на базі queue, що зберігає множину чисел і має внутрішній лічильник k спочатку рівний нулю. Структура повинна підтримувати операції додавання числа в множину і повернення k -го по мінімальності числа з множини.
- 7) На клітковому аркуші намальований круг. Вивести в файл опису всіх клітин, цілком лежать всередині кола в порядку зростання відстані від клітини до центру кола. Використовувати клас priority_queue.
- 8) На базі шаблону list реалізувати структуру зберігання чисел з підтримкою наступних операцій:
 - додавання / видалення числа;
 - пошук числа, найближчого до заданого (тобто модуль різниці мінімальний).
- 9) У вхідному файлі розташовані два набору додатних цілих чисел; між наборами – роздільник (будь-яке від'ємне число). Побудувати два списки S_1 і S_2 , елементи яких містять відповідно числа 1-го і 2-го набору таким чином, щоб усередині одного списку числа були впорядковані

за зростанням. Потім об'єднати списки C1 і C2 в один відсортований список.

- 10) Реалізуйте клас `Auto`, що містить члени: назва, модель, номер, ідентифікатор власника. Визначте для цього класу методи введення/виведення. Реалізуйте за допомогою стандартних шаблонів наступні задачі:
- а) в шаблоні `vector` даний масив даних про авто, потрібно вивести всіх власників даної марки;
 - б) в шаблоні `list` є дані про авто, відсортуйте їх по назві та виведіть всі їх номери в цьому порядку;
 - в) в шаблоні `deque` зберігаються дані по черзі з авто на заправці. Проделайте заповнення черги на заправці виводячи стан черги при кожному вибуванні чи прибуванні авто на заправку;
 - г) в шаблоні `stack` зберігаються авто на складі ринку. Проделайте роботу складу: виведіть стан складу при поповненні та видавленні змісту складу;
 - д) використайте шаблон `queue` для моделювання черги з авто на мойку;
 - е) використайте шаблон `priority_queue` для моделювання черги замовлень по ремонту в залежності від вартості ремонту (це додатковий член класу, що вводиться окремим методом).
- 11) Складіть клас `Employee` із двома членами даних: `hours` та `hourlyPay`. Працівник також повинен мати функцію `calcSalary()`, яка повертає заробітну плату за цього працівника. Генеруйте довільну погодинну оплату праці та години для довільної кількості працівників. Зберігайте вектор `Співробітник`. Дізнайтеся, скільки грошей компанія витратить за даний період оплати праці.
- 12) Створіть шаблон класу `Matrix`, який створений як `vector <vector <T>>`. Надайте його дружньому методу `ostream & operator << (ostream &, const Matrix &)` для відображення матриці. Створіть наступні бінарні операції, використовуючи об'єкти функції STL, де це можливо: оператор `+` (`const Matrix &, const Matrix &`) для додавання матриці, оператор `*` (`const Matrix &, const vector <int> &`) для множення матриці на вектор та оператор `*` (`const Matrix &, const Matrix &`) для множення матриць. Перевірте шаблон класу `Matrix`, використовуючи `int` і `float`.

- 13) Реалізувати функцію, що виконує додавання чисел, заданих вектором `unsigned char` в різних системах числення:

```
vector<int> addition (const vector<UCHAR> & A, int baseA, const
vector<UCHAR> & B, int baseB, int baseResult);
```

Функція повинна перевіряти вхідні дані про коректність і повертати порожній вектор у разі виявлення помилки. В текстових файлах записані перше та друге число та останній аргумент – основа числення. Основа системи числення результату вводиться з консолі та результат записується в третій файл. Для даної функції повинен бути створений набір тестів, що перевіряють функції на наборі прикладів, та коректність введення в разі некоректних даних.

Додатково реалізуйте також підтримку записів вхідних даних у нових рядках. Наприклад

```
16: "FD" або
10: "253"
2
```

Результат буде в новому рядку: "111111101"

- 14) Даний текстовий файл, що містить рядкові представлення цілих чисел. Заповнити вектор `V` числами з цього файлу та вивести їх у вихідному порядку. У випадку некоректних даних видайте змістовне повідомлення.
- 15) В консолі вводиться масив цілих чисел. Заповнить список `L` цими числами і вивести елементи списку `L` в початковому порядку у вихідному, а потім в оберненому порядку. Відсортуйте дані за зростанням у списку, але виведіть навпаки за спаданням.
- 16) Даний вектор цілих чисел з парною кількістю елементів. Заповнить дек `D` даними числами так, щоб перша половина чисел збігалася з порядком заповнення вектору, а друга була в зворотному порядку.
- 17) Ввести список цілих чисел з консолі. Вставити перед кожним ненульовим елементом вихідного списку число `-1`, а після кожного рівного `2` – нуль.
- 18) Ввести з консолі список `L` натуральних чисел. Вставити після кожного непарного елементу з першої половини вихідного списку число `-1`, а перед кожним парним елементом другої половини `-2`.
- 19) Ввести з текстового файлу дек довільного типу `D`. Видалить середній

елемент дека, якщо кількість елементів непарна або 2 середні елементи – якщо парна.

- 20) Ввести з текстового файлу дійсний вектор V з непарною кількістю елементів N ($N \geq 5$). Якщо там парна кількість елементів – додати до вектору 3 дійсні числа з консолі. Видалити три середні елементи (тобто середній елемент та два його сусіди) вектора за один виклик `erase`.
- 21) Ввести список L з консолі та вектор V з бінарного файлу (тип – рядок). Перемістити елемент списку L з даним номером в кінець списку V .
- 22) Ввести список L з елементами $A_1, A_2, \dots, A_{N-1}, A_N$ (N – парне, якщо ні, то додайте нуль до списку). Змінити порядок елементів у списку на наступний: $A_1, A_N, A_2, A_{N-1}, A_3, A_{N-2}, \dots, A_{N/2}, A_{N/2-1}$.
- 23) Ввести два списки $L1$ і $L2$ з однаковою кількістю елементів – N . Якщо це не так, то видалить з кінця більшого списку потрібну кількість елементів. Отримати в списку $L2$ комбінований набір елементів – список вигляду $B_1, A_1, B_2, A_2, \dots, B_N, A_N$, де A_I – елементи вихідного списку $L1$, а B_I – елементи списку $L2$. *Вказівка.* Використайте метод `splice` для $L2$ з інкрементами у другому та третьому аргументах.

Розділ 20

Стандартна бібліотека C++. Асоціативні контейнери.

Контрольні запитання:

- Які типи ітераторів існують в STL? Які з них визначені для вектору? Які до списку? Які для множини?
- Які асоціативні контейнери існують в C++? Що додає приставка `multi` до назви контейнера?
- Для яких стандартних класів-шаблонів не визначений метод `push_back()`? Чому? Як в ці класи додаються елементи?
- Для яких стандартних класів-шаблонів не визначений метод `push_back()`? Чому? Як в ці класи додаються елементи?
- Як визначити кількість елементів будь-якого контейнеру?
- Які коректні шляхи ітерації по вектору? Мультивідображенню? Будь-якому контейнеру?
- Як коректно пройти по всім елементам відображення?

Завдання для аудиторної роботи:

- 1) Заданий файл з текстом англійською мовою. Виділити все різні слова. Для кожного слова підрахувати частоту його входження. Слова, що відрізняються регістром літер, вважати різними. Використовувати Map.
- 2) З використанням Set виконати попарне підсумовування довільного кінцевого ряду чисел за такими правилами: на першому етапі підсумовуються попарно сусідні числа, на другому етапі підсумовуються результати першого етапу і т. д. до тих пір, поки не залишиться одне число.
- 3) У файлі записані координати точок на площині задані парою цілих чисел. Точки записані в форматі : (x_1 , x_2) (x_1 , x_2) , ... - саме так через коми та дужки. Створити файл, в якому будуть записані координати всіх відрізків з точок цього файлу, при цьому ці відрізки відсортовані за зростанням довжини.

Завдання для самостійної роботи:

- 5) На площині задано N точок. Вивести в файл описи всіх прямих, які проходять більш ніж через одну точку із заданих. Для кожної прямої вказати, через скільки точок вона проходить. Використовувати клас `multimap`.
- 6) На площині задано N відрізків. Знайти точку перетину двох відрізків, що має мінімальну абсцису. Використовувати клас `map`.
- 7) Відредагувати задане речення, видаляючи з нього ті слова, які зустрічаються в реченні задану кількість разів.
- 8) На клітковому аркуші паперу зафарбована частина клітин. Виділити все різні фігури, які утворилися при цьому. Фігурою вважається набір зафарбованих клітин, які сусідні один з одного при руху в чотирьох напрямках. Дві фігури є різними, якщо їх не можна сумістити поворотом на кут, кратний 90 градусам, і паралельним зсувом. Використовуйте клас `multiset`.
- 9) Напишіть програму, яка знаходить усі спільні слова між двома вхідними файлами, використовуючи `set_intersection()`. Змініть його, щоб показати слова, які не є спільними, за допомогою `set_symmetric_difference()`.
- 10) З бінарного файлу зчитати вектор V_0 , ціле число $N > 0$ і набір векторів V_1, \dots, V_N . Знайти кількість векторів V_i , $i = 1, \dots, N$, в яких містяться всі елементи вектору V_0 (з урахуванням повторень). Використати алгоритм `include`.
- 11) З консолі введений рядок та вектор символів V . Знайдіть усі різні числа, які одночасно входять і в першу, і в другу половину вектору, і записати їх у текстовому файлі з заданим ім'ям у порядку заданому вектором, додаючи після кожного числа символ пробілу (використати `set_intersection`).
- 12) Даний рядок `name` і вектор V з парною кількістю цілих елементів. Знайдіть усі різні числа, які входять у другу половину вектору та при цьому відсутні в першій половині. Записати ці числа в текстовому файлі з ім'ям `name` за зростанням, виводиться кожне число на новому рядку. Використовувати алгоритм `set_difference`.
- 13) Ввести з консолі вектори рядків $V1$ і $V2$. Знайти всі числа (з урахуванням повторень), які входять хоча б в один із вихідних векторів, і вивести їх в порядку незростання; при цьому, якщо, наприклад, де-

який рядок входить в один із векторів 3 рази, а в іншій 5 разів, то його потрібно вивести 5 разів. Використовувати алгоритм `set_union`.

- 14) Ввести з текстового файлу вектор `V`, що містить не менше трьох різних чисел. Виведіть усі його різні елементи, крім максимального та мінімального, в порядку спадання за допомогою допоміжної множини та без неї.
- 15) Дан текстовий файл з ім'ям `NameT`. Підрахувати число повторень в ньому малих латинських літер ('a' - 'z') і створити файл з ім'ям `NameS`, рядки якого мають вигляд: "<літера> - <число повторень даної літери>". Літери, відсутні в тексті, в файл не включати. Рядки впорядкувати за спаданням кількості повторень літер, а при однаковій кількості повторень – по зростанню кодів літер.
- 16) Ввести вектор `V`, елементи якого є англійськими словами. Визначити сумарну довжину слів, що починається з однієї і тієї же літери (без урахування регістру), і вивести всі різні літери, з яких починаються елементи вектору `V`, разом із сумарною кількістю цих елементів (у алфавітному порядку букв); довжину виводити зразу після відповідної літери. Використовувати допоміжне відображення `M`, ключі якого є початковими літерами елементів вектору `V`, а значення – сумарна кількість цих елементів. При заповненні відображення `M` не використовувати умовні конструкції.
- 17) Ввести вектор цілих чисел `V`. Виконати групування елементів вектору `V`, використовуючи у якості ключу груп останню (тобто праву) цифру елементу: в одній групі повинні входити всі елементи `V`, що закінчуються однією цифрою (згруповані елементи повинні розташовуватися в тому ж порядку, в якому вони доступні у векторі). Представити результат групування у вигляді відображення `M`, ключі яких є ключами групування, а значення – вектори, що містять згруповані елементи (таким чином, відображення `M` повинно мати тип `map<int, vector<int>>`). Вивести отримане відображення (для кожного елементу відображення `M` спочатку вивести ключ, двокрапку, а потім елементи пов'язаного з ним вектору через кому).
- 18) Ввести вектор `V`, елементи якого є англійськими словами. Виконати групування елементів вектору `V`, використовуючи як ключ групувань другу літеру елементів (або першу – якщо слова містять лише одну літеру): В групі повинні входити всі елементи `V`. Представити результат групування у вигляді відображення `M`, ключі якого є ключі групування (таким чином, відображення `M` повинно мати тип `multi-`

map <char, string>). Вивести отримане відображення (для кожного елемента відображення M вивести ключ, а потім пов'язаний з ним елемент вектору V, ключі можуть повторюватися).

- 19) Ввести вектор дійсних чисел V. У кожній групі його елементів, що мають однакову початкову цифру, знайдіть суму значень цих елементів, за виключенням початкового елемента групи (передбачається, що елементи групи розміщуються в тому ж порядку, що і у вихідному векторі). Якщо група складається з єдиного елемента, то сума повинна дорівнювати 0. Для кожної групи виводьте відповідну їй цифру і знайдену суму, впорядкуйте пари по зростанню цифр, а потім суми.
- 20) Послідовність даних в текстовому файлі містить відомості про клієнтів клубу. Кожний елемент послідовності включає наступні цілі поля: <Код клієнту> <Рік> <Номер місяця> <Кількість занять (у годинах)> Знайти елемент послідовності з мінімальною тривалістю занять. Виведіть цю інформацію, а також відповідний рік і номер місяця (у вказаному порядку в тому ж рядку). Якщо таких елементів декілька — виведіть усі в довільному порядку.
- 21) Послідовність даних у бінарному файлі містить відомості про оцінки учнів з трьох предметів: алгебра, геометрія та інформатика. Кожен елемент послідовності містить дані про одну оцінку та має наступні поля: <Прізвище> <Ініціали> <Клас> <Назва предмета> <Оцінка>. Повних тезок (з однаковим прізвищем та ініціалами) серед учнів немає. Клас задається цілим числом, оцінка - цілим числом в діапазоні 1–12. Назва предмету вказується з заголовної літери. Для кожного учня визначити середню оцінку по кожному з предметів і вивести її з двома знаками після крапки (якщо не має жодної оцінки, то вивести для цього предмету 0.0). Відомості про кожного учня виводити на окремому рядку, вказавши прізвище, ініціали та середні оцінки по алгебрі, геометрії та інформатиці. Дані розмістити в алфавітному порядку прізвищ та ініціалів.
- 22) Реалізувати красиве виведення для всіх асоціативних контейнерів STL (map, set, multimap, multiset) за допомогою стандартного виведення, тобто потрібно перевантажити функцію std::operator << для відповідних типів. Приклад виклику:

```
...
std::set<int> testSet{1,3};
std::map<int, std::string> testMap;
testMap[1] = "123";
```

```
testMap[2] = "456";  
std::cout << testMap;  
std::cout << testSet;  
...
```

Вихідні дані (текстовий файл/виведення в консоль):

```
1=123; 2=456  
1,3
```

Додаткові задачі:

- 23) Необхідно реалізувати Proxy (заступник) для контейнера map, що керує доступом. Реалізувати надбудову над стандартним контейнером map, яка дозволяла б управляти доступом до певних ключів (з розмежуванням читання, запису та видалення), а також підмінити при необхідності результати.

У цій системі є три класи:

- `std :: map` – базовий контейнер;
- `Auditor` – базовий клас «аудитора», який реалізує перевірку прав і заміну значень;
- `ProxiedMap` – надбудова над контейнером, яка все операції з ним передує викликом об'єкта `Auditor`.

Реалізувати декілька підкласів для `Auditor` (той що все дозволяє, що дозволяє читання та підмінює значення по заданих ключами константних значеннях і т.д.)

Розробити набір тестів, що перевіряє чи дійсно блокуються спроби несанкціонованого доступу (для різних об'єктів класу `Auditor`). Можливе ускладнення: Реалізувати декілька видів ітераторів (наприклад: ітератор, що обходить всі доступні ключі; ітератор, що обходить всі ключі, але сигналізує про помилку доступу при спробі звернення і т.д.)

Розділ 21

Стандартна бібліотека C++. Алгоритми та функтори.

Контрольні запитання:

- Які існують бібліотеки алгоритмів в STL?
- Як скористатись алгоритмами сортування? Акумуляованої суми? Пошуку?
- Як знайти потрібний елемент в будь-якому контейнері за допомогою стандартних алгоритмів?
- Що таке предикат та функтор? Як ними скористатись?
- Як відсортувати вектор по незростанню? Неспаданню?
- Коли і як можна скористатись бінарним пошуком в колекції?
- Як за допомогою алгоритмів знайти суму масиву одним викликом функції? А як норму вектору двома викликами за допомогою функтору?

Завдання для аудиторної роботи:

- 1) Побудуйте клас БулевийВектор (BoolVector) на базі `vector<bool>`. Реалізувати методи для виконання порозрядних кон'юнкції, диз'юнкції і заперечення векторів, а також підрахунку кількості одиниць і нулів у векторі.
- 2) Введіть з консолі список цілих чисел List і число X. Не використовуючи допоміжних об'єктів і не змінюючи розміру списку, переставити елементи списку так, щоб спочатку йшли числа, що не перевищують X, а потім числа, які є більшими за X.
- 3) Напишіть функцію, яка повертає а) суму найбільших k чисел даного вектору, б) масив з k найменших чисел даного вектору, якщо k не перевищує розмір масиву та: а) нуль, б) порожній масив в протилежному випадку.
- 4) Створіть генератор, який повертає поточне значення `clock()` (`y < ctime`). Створіть список `<clock_t>` і заповніть його своїм генератором за допомогою функції `create_n()`. Видаліть усі дублікати зі списку та роздрукуйте його на `cout` за допомогою `сору()`.

- 5) За допомогою функцій `transform()` і `toupper()` (з `<cctype>`) напишіть один виклик функції, який перетворить рядок на всі великі літери.
- 6) Створіть шаблон функціонального об'єкту `Sum`, який буде накопичувати всі суми значень в діапазоні при використанні з `for_each()`.
- 7) Напишіть генератор анаграм, який приймає слово як аргумент командного рядка і створює всі можливі перестановки літер.
- 8) Напишіть генератор анаграм речень, який приймає речення як аргумент командного рядка і створює всі можливі перестановки слів у реченні.

Завдання для самостійної роботи:

- 9) У файлі записані координати Точок в просторі задані трійкою цілих чисел. Точки записуються в форматі : `x1 , x2 , x3 ; x1 , x2, x3 ; ...`. Створити файл, в якому будуть записані відрізки з точок цього файлу, при цьому ці відрізки відсортовані за зростанням довжини.
- 10) Армія хоче набрати людей зі свого виборчого списку служб. Вони вирішили набрати тих, кому менше 35 років, починаючи від найстаршого і закінчуючи наймолодшим. Згенеруйте довільну кількість людей. Надайте їм такі дані, як вік та прізвища записані як вектор. Розділіть вектор так, щоб ті, кому менше 35 років, були упорядковані на початку списку, починаючи від наймолодшого до найстаршого, а решту частину списку залишали відсортувати за прізвищем.
- 11) Напишіть програму, яка знаходить усі спільні слова між двома вхідними файлами, використовуючи `set_intersection()`. Змініть його, щоб показати слова, які не є спільними, за допомогою `set_symmetric_difference()`.
- 12) У файлі записані координати Точок на площині задані парою цілих чисел та масою(дійсне число). Точки записуються в форматі : `(x1 , x2): m1 , (x1 , x2): m2 , ...`. Створити файл, в якому будуть записані відрізки з точок цього файлу, при цьому ці точки відсортовані за важелем сили (`m1*(x1 +x2)`).
- 13) У файлі записані дати , що трійкою цілих чисел у форматі: `чч/мм/рр, ...`. Створити файл, в якому будуть записані дати з цього файлу без повторень, при цьому ці дати відсортовані за спадання дати (врахуйте, що роки дат з 1951 по 2049).
- 14) У файлі записані дати , що двома цілими числами та рядком (англійські або числові назви місяця) у форматі: `чч1 місяць1 рік1, чч2`

місяць2 рік2... Вивести дати без повторень з цього файлу у форматі: рік1/місяць1/число1, рік2/місяць1/число2,... (місяць заданий назвою) при цьому ці дати відсортовані за зростанням дати

- 15) Створіть ієрархію класів з базовим класом `B` та похідним класом `D`. Помістіть функцію віртуального члена `void f ()` у `B` таким чином, щоб вона надрукувала повідомлення, що вказує, що було викликано `B sf ()`, і перевизначити цю функцію для `D` щоб надрукувати інше повідомлення. Створіть вектор `<B *>` і заповніть його об'єктами `B` і `D`. Використовуйте `for_each ()` для виклику `f ()` для кожного з об'єктів у вашому векторі.

Створіть програму, яка, отримуючи ціле число в командному рядку, створює таблицю факторіалів з усіх факторіалів, включаючи число в командному рядку. Для цього напишіть генератор для заповнення вектора `<int>`, а потім використовуйте `partial_sum()` зі стандартним об'єктом функтором.

- 16) Створіть алгоритм стилю STL `transform_if ()`, який схожий на першу форму `transform ()`, але виконує перетворення лише на об'єктах, які задовольняють одинарному предикату. Об'єкти, які не задовольняють предикату, опускаються з результату. Потрібно повернути новий кінцевий ітератор.
- 17) Створіть алгоритм стилю STL, який є перевантаженою версією `for_each ()`, але використовує другу форму `transform()` і займає два діапазони введення, щоб він міг передавати об'єкти другого діапазону введення а двійковій функції, яку він застосовує до кожного об'єкта першого діапазону.
- 18) Створіть клас «Місто» з даними про населення, висоту та погоду. Зробіть погоду за допомогою перерахування {ДОЩИТЬ, СНІЖНО, ХМАРНО, ЯСНО}. Створіть клас, який генерує об'єкти `Town`. Створіть назви міст (незалежно від того, мають вони сенс чи ні, це не має значення) або витягніть їх з Інтернету. Переконайтеся, що назва всього міста має маленькі регістри, а дублікатів назв немає. Для населення, висот та погодних полів створіть генератор, який випадковим чином генеруватиме погодні умови, популяції в межах [100-1000000) та висоти між [0, 8000) футами.

а) Заповніть вектор об'єктами міста. Перепишіть вектор у новий файл під назвою `Towns.txt`.

б) Відбувся бебі-бум, що призвело до збільшення населення на 10%

у кожному місті. Оновіть дані про місто за допомогою `transform()`, переписіть дані назад у файл.

- в) Знайдіть міста з найбільшим і найменшим населенням. Для цієї вправи застосуйте оператор `<` для вашого класу `Town`. Також спробуйте реалізувати функцію, яка повертає `true`, якщо її перший параметр менше, ніж другий. Використовуйте його як предикат для виклику використовованого вами алгоритму.
- г) Знайдіть усі міста на висоті 2500–3500 футів включно. За необхідності реалізуйте оператори рівності для класу `Town`.
- д) Нам потрібно розмістити аеропорт на певній висоті, але розташування не є проблемою. Впорядкуйте свій список міст так, щоб не було дублікатів (дублікат означає, що жодні дві висоти не знаходяться в одному діапазоні 100 футів. До таких класів належать [100, 199], [200, 199] і т.д. Відсортуйте цей список за зростанням принаймні двома різними способами, використовуючи об'єкти функції в `<functional>`. Зробіть те ж саме для порядку зменшення. За необхідності впроваджуйте реляційні оператори для міста.

19) Створіть довільну кількість випадкових чисел у масиві на основі стеку. Використовуйте `max_element()`, щоб знайти найбільше число в масиві. Поміняйте його номером у кінці масиву. Знайдіть наступне найбільше число та розташуйте його в масиві в позиції перед попереднім числом. Продовжуйте це робити, доки всі елементи не будуть переміщені. Коли алгоритм буде завершено, ви отримаєте відсортований масив. (Це сортування виділенням)

20) Напишіть програму, яка знімає телефонні номери з файлу (що також містить імена та іншу відповідну інформацію) та змінює номери, що починаються з 222 на 863. Обов'язково збережіть старі номери. Формат файлу такий:

```
222 8945
756 3920
222 8432
```

Напишіть програму, яка за прізвищем знайде кожного з цим прізвищем із відповідним номером телефону. Використовуйте алгоритми, які мають справу з діапазонами (`upper_bound`, `lower_bound`, `equal_range` тощо). Сортуйте за прізвищем, що діє як первинний ключ, а за іменем, що діє як вторинний ключ. Припустимо, що

ви прочитаєте імена та номери з файлу, формат якого буде таким. Обов'язково впорядкуйте їх так, щоб прізвища були впорядковані, а імена впорядковані в межах прізвищ:

Ivanov Ivan 365 9583

Ivanov Petro 379 2530

Petrov Petro 253 2619

Отримавши файл із даними, подібними до наведених нижче, витягніть із нього всі державні аббревіатури та помістіть їх в окремий файл. (Зверніть увагу, що ви не можете залежати від номера рядка для типу даних. Дані містяться на випадкових рядках.)

- 21) Порівняйте роботу функцій `sort()`, `partial_sort()` та `nth_element()` одну з іншою і з'ясуйте, чи дійсно варто використовувати одне із слабких сортувань, коли вони можуть спрацювати коректно.
- 22) Введіть відображення в якому ключ – це слово, а значення декілька слів, які визначають це слово. При цьому порочного кола немає. Пронумеруйте слова таким чином, щоб слова з більшим номером визначались лише словами з меншими номерами та виведіть це представлення у файл.

Бібліографія

- [1] Керниган Б., Ритчи Д. Мова програмування Сі. Пер. з англ., 3-є изд., испр. – СПб. : "Невський Діалект 2001. – 352 с.
- [2] Кренивич А. П. С у задачах і прикладах : навчальний посібник із дисципліни "Інформатика та програмування" / А.П. Кренивич, О.В. Обвінцев. – К. : Видавничо-поліграфічний центр "Київський університет 2011. – 208 с.
- [3] Кренивич А. П. С у задачах і прикладах : навч. посібник / А. П. Кренивич, О. В. Обвінцев. – Київ : ВПЦ "Київський університет", 2012. – 212 с.
- [4] Вакал Є.С., Личман В.В., Обвінцев О.В., Бублик В.В., Попов В.В. Задачі до курсу «Інформатика та програмування». [Електронний ресурс] –Режим доступу до ресурсу: <http://matfiz.univ.kiev.ua/informatics/problems/>
- [5] Збірник задач з дисципліни "Інформатика і програмування" Вакал Є.С., Личман В.В., Обвінцев О.В., Бублик В.В., Довгий Б.П., Попов В.В. -2-ге видання, виправлене та доповнене –К.:ВПЦ "Київський університет 2006.–94с
- [6] Кренивич А.П. Python у прикладах і задачах. Частина 1. Структурне програмування Навчальний посібник із дисципліни "Інформатика та програмування"–К.: ВПЦ "Київський Університет 2017. –206с.
- [7] Вінник В.Ю. Алгоритмічні мови та основи програмування: мова С / В.Ю. Вінник – Житомир: ЖДТУ, 2007. – 328 с.
- [8] Трофименко О.Г., Прокоп Ю.В., Швайко І.Г. та ін. С++. Основи програмування. Теорія та практика: Підручник / О.Г. Трофименко, Ю.В. Прокоп, І.Г. Швайко, Л.М. Буката, Л.А. Косирева, Ю.Г. Леонов, В.В. Ясинський; за ред. О.Г. Трофименко. – Одеса: Фенікс, 2010. — 544 с.
- [9] Белов Ю.А., Карнаух Т.О., Коваль Ю.В., Ставровський А.Б. Вступ до програмування мовою С++. Організація обчислень : навч. посіб. / Ю. А. Белов, Т. О. Карнаух, Ю. В. Коваль, А. Б. Ставровський. – К. : Видавничо-поліграфічний центр "Київський університет 2012. – 175 с

- [10] Прата С. Язык программирования С++. Лекции и упражнения. 6-е изд. – М.: Издательский дом «Вильямс», 2012. – 1248 с.
- [11] Alex Allain. Jumping into C++. [Электронный ресурс] –Режим доступа до ресурсу: www.cprogramming.com
- [12] Абрамян, М. Э. Введение в стандартную библиотеку шаблонов С++. Описание, примеры использования, учебные задачи : учебник / М. Э. Абрамян ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Изд-во ЮФУ, 2017. – 177 с.
- [13] Крячков, А.В. Программирование на С и С++. Практикум : учеб. пособие / А.В. Крячков, И.В. Сухина, В.К. Томшин. – 2-е изд., испр.– М.: Горячая линия–Телеком, 2000. – 344 с.: ил.
- [14] Бублик В.В. Методические указания и задания к лабораторным занятиям по курсу "Вычислительные машины и программирование" для студентов механико-математического факультета / В.В. Бублик – К. : КГУ, 1986. –60 с.
- [15] Пильщиков, В.Н. Сборник упражнений по языку Паскаль : учеб. пособие для вузов / В.Н. Пильщиков. – М. : Наука, 1989. – 160 с
- [16] С.А. Абрамов, Г.Г. Гнездилова, Е.Н. Капустина, М.И. Селюн. Задачи по программированию - Вологда, 2000г. - 595 с.
- [17] Электронний довідник С language [Електронний ресурс] –Режим доступу до ресурсу: <https://en.cppreference.com/w/c/language>
- [18] Відкритий підручник по С++ українською мовою [Електронний ресурс] –Режим доступу до ресурсу: <https://purecodecpp.com/uk/archives/920>
- [19] Bruce Eckel. Thinking on C++. [Електронний ресурс] –Режим доступу до ресурсу: <https://www.micc.unifi.it/bertini/download/programmazione/TICPP-2nd-ed-Vol-one-printed.pdf>
- [20] Блинов, И.Н., Романчик, В. С. Java. Методы программирования : уч.-мет. пособие / И. Н. Блинов, В. С. Романчик. — Минск : издательство «Четыре четверти», 2013. — 896 с. - ISBN 978-985-7058-30-3.