

Міністерство освіти і науки України  
Національний університет "Львівська політехніка"  
Інститут комп'ютерних наук та інформаційних технологій  
Кафедра програмного забезпечення



### **Звіт**

Про виконання лабораторної роботи №3  
з дисципліни «Вступ до інженерії програмного забезпечення»

**Лектор:**

доцент каф. ПЗ  
Левус Є.В.

**Виконав:**

ст. гр. ПЗ-11  
Ясногородський Н. В.

**Прийняла:**

доцент каф. ПЗ  
Левус Є.В.

« \_\_ » \_\_\_\_\_ 2022 р.

$\Sigma$  = \_\_\_\_\_ .

**Тема:** Зміст та роль тестування у життєвому циклі розробки програмного забезпечення.

**Мета:** Вивчити основні методи та принципи тестування на прикладі створення власної програми.

**Теоретичні відомості:**

Відповіді на контрольні запитання:

9. У чому полягає перевірка логічної схеми програми?

Перевірити кожний можливий логічний шлях на коректність. Дозволяє виявити крайні випадки та перевірити частини коду які дуже рідко викликаються

18. Сформууйте рекомендації стосовно тестування на основі власного досвіду програмування, про які не згадувалося вище?

Якщо потреба у коректності домінує всі інші бізнес потреби (зазвичай швидкість), то варто писати програмний код за метадологією TDD (Test-Driven-Development). У ній спочатку пишуть тести до програми і тільки потім реалізацію

31. Що таке альфа тестування?

Зазвичай закрите тестування всередині компанії розробників

**Завдання:**

1. Провести функціональне тестування програми, використавши такі тести:

1. димовий тест,
2. тест нормальних умов,
3. тест екстремальних(граничних) умов,
4. тест виняткових умов.

1.1. Результати тестів 1, 2, 3 та 4 оформити у вигляді таблиці (за зразок взяти таблицю 2). У таблиці інформація заноситься в зрозумілій формі, достатньо повно, щоб не вимагалось додаткових (усних) пояснень. Для представлення тестових даних та фактичних результатів бажано використовувати посилання на зображення екранів чи відповідні файли. В очікуваних результатах зафіксувати, яка функція була протестована та відповідно, що є очікуваним результатом. Якщо помилки не знайдено у Тип звіту ставимо знак -. Примітки використовуємо для додаткових пояснень.

1.2. Знайдені дефекти виправити. Провести повторне тестування, результати якого оформити окремою таблицею.

2. Окремо провести тестування логічної схеми програми (головна програма та дві функції-підпрограми). Для цього зобразити її у вигляді графів управління.

Вершини графів пронумерувати. Записати усі отримані маршрути роботи програми, використовуючи номери вершин.

2.1. Для кожного маршруту провести тестування і записати результати тестування у звітній таблиці.

2.2. Знайдені дефекти виправити. Провести повторне тестування, результати якого оформити окремою таблицею.

### Lab 10 Ver 1

№	Тестові дані	Фактичні результати	Очікувані результати	Ступінь критичності	Тип звіту	Примітки
Димовий тест						
1	Запуск програми	Запуск програми без аварійних завершень	Запуск програми без аварійних завершень	-	-	-
Тест нормальних умов						
1	Файл students.json	Коректне зчитування даних про студентів і вивід списку	Коректне зчитування даних про студентів і вивід списку	-	-	-
2	Ручне введення студентів з коректними даними	Зчитування всіх даних з клавіатури і виведення списку	Зчитування всіх даних з клавіатури і виведення списку	-	Пропозиція	Додати можливість редагування існуючих студентів
3	Файл students.json	Виведення студентів без оцінок 2 (за допомогою фільтра)	Виведення студентів без оцінок 2	-	-	-

4	Файл students.json	Виведення студентів тільки з оцінками 4 та 5 (за допомогою фільтра)	Виведення студентів тільки з оцінками 4 та 5			
4	Файл students.json	Виведення на екран студентів, відсортованих за віком групи.	Виведення на екран студентів, відсортованих за віком групи.	-	-	-
5	Файл students.json	Запис студентів до файлу «students.json»	Запис студентів до файлу «students.json»	-	-	-
Тест екстремальних умов						
1	Порожній файл	Пустий вивід	Пустий вивід	-	Пропозиція	Пропозиція : сповіщення користувача про те, що список порожній
2	Файл students.json (Граничні дані оцінок - 5)	Успішне зчитування і опрацювання даних.	Успішне зчитування і опрацювання даних.	-	-	-
3	Файл students_with_max_date.json (Дата народження сьогодні)	Успішне зчитування і опрацювання даних.	Успішне зчитування і опрацювання даних.	-	-	-

4	Файл students.json (Граничні дані оцінок - 1)	Успішне зчитування і опрацювання даних.	Успішне зчитування і опрацювання даних.	-	-	-
5	Файл students_without_2.json	Фільтр про відсутність оцінки 2 не впливає на вивід	Фільтр про відсутність оцінки 2 не впливає на вивід	-	-	-
6	Файл students_with_same_dates_and_4_5_grades.json	Файл має студентів з однаковим віком. Сортуювання виводить будь яких двох наймолодших	Сортуювання виводить будь яких двох наймолодших	-	-	-
7	Файл students_only_with_2_grades.json	Фільтр про відсутність оцінки 2 має показати пустий вивід	Фільтр про відсутність оцінки 2 має показати пустий вивід	-	-	-
Тест виняткових умов						
1	Файл students_wrong_name_type.json (імя не стрічка)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	Середній	Помилка проектування	Потрібно додати валідацію типів вхідних даних з файлу
2	Файл students_without_birthDate.json (без дати народження)	Коректна робота програми, сортування за датою нічого не робить	Коректна робота програми, сортування за датою нічого не робить	-	-	-

3	Файл students_too_big_date.json (Дата народження після сьогодні)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-
4	Введення дати в некоректному форматі (коректний формат: будь який iso)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-
5	Файл students_wrong_grades_value.json (оцінки більше 5)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-
6	Файл students_wrong_grades_value.json (оцінки менше 1)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-
7	students_wrong_grades_type.json (Оцінка не число)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	Низька	Помилка кодування	Додати перевірку на числа

### Lab 10 Ver 2

№	Тестові дані	Фактичні результати	Очікувані результати	Ступінь критичності	Тип звіту	Примітки
Димовий тест						

1	Запуск програми	Запуск програми без аварійних завершень	Запуск програми без аварійних завершень	-	-	-
Тест нормальних умов						
1	Файл students.json	Коректне зчитування даних про студентів і вивід списку	Коректне зчитування даних про студентів і вивід списку	-	-	-
2	Ручне введення студентів з коректними даними	Зчитування всіх даних з клавіатури і виведення списку	Зчитування всіх даних з клавіатури і виведення списку	-	Пропозиція	Пропозиція реалізована
3	Файл students.json	Виведення студентів без оцінок 2 (за допомогою фільтра)	Виведення студентів без оцінок 2	-	-	-
4	Файл students.json	Виведення студентів тільки з оцінками 4 та 5 (за допомогою фільтра)	Виведення студентів тільки з оцінками 4 та 5			
4	Файл students.json	Виведення на екран студентів, відсортованих за віком групи.	Виведення на екран студентів, відсортованих за віком групи.	-	-	-
5	Файл students.json	Запис студентів до файлу «students.json»	Запис студентів до файлу «students.json»	-	-	-

Тест екстремальних умов						
1	Порожній файл	Пустий вивід	Пустий вивід	-	Пропозиція	Пропозиція реалізована
2	Файл students.json (Граничні дані оцінок - 5)	Успішне зчитування і опрацювання даних.	Успішне зчитування і опрацювання даних.	-	-	-
3	Файл students_with_max_date.json (Дата народження сьогодні)	Успішне зчитування і опрацювання даних.	Успішне зчитування і опрацювання даних.	-	-	-
4	Файл students.json (Граничні дані оцінок - 1)	Успішне зчитування і опрацювання даних.	Успішне зчитування і опрацювання даних.	-	-	-
5	Файл students_without_2.json	Фільтр про відсутність оцінки 2 не впливає на вивід	Фільтр про відсутність оцінки 2 не впливає на вивід	-	-	-
6	Файл students_with_same_dates_and_4_5_grades.json	Файл має студентів з однаковим віком. Сортують виводить будь яких двох наймолодших	Сортують виводить будь яких двох наймолодших	-	-	-
7	Файл students_only_with_2_grades.json	Фільтр про відсутність оцінки 2 має показати пустий вивід	Фільтр про відсутність оцінки 2 має показати пустий вивід	-	-	-
Тест виняткових умов						



1	Файл students_wrong_name_type.json (ім'я не стрічка)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	Середній	Помилка проєктування	Помилку виправлено
2	Файл students_without_birthDate.json (без дати народження)	Коректна робота програми, сортування за датою нічого не робить	Коректна робота програми, сортування за датою нічого не робить	-	-	-
3	Файл students_too_big_date.json (Дата народження після сьогодні)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-
4	Введення дати в некоректному форматі (коректний формат: будь який iso)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-
5	Файл students_wrong_grades_value.json (оцінки більше 5)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-
6	Файл students_wrong_grades_value.json (оцінки менше 1)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	-	-	-

7	students_wrong_grades_type.json (Оцінка не число)	Виведення помилки про некоректність даних.	Виведення помилки про некоректність даних.	Низька	Помилка кодування	Помилку виправлено
---	--	--	--	--------	-------------------	--------------------

## Результати структурного тестування

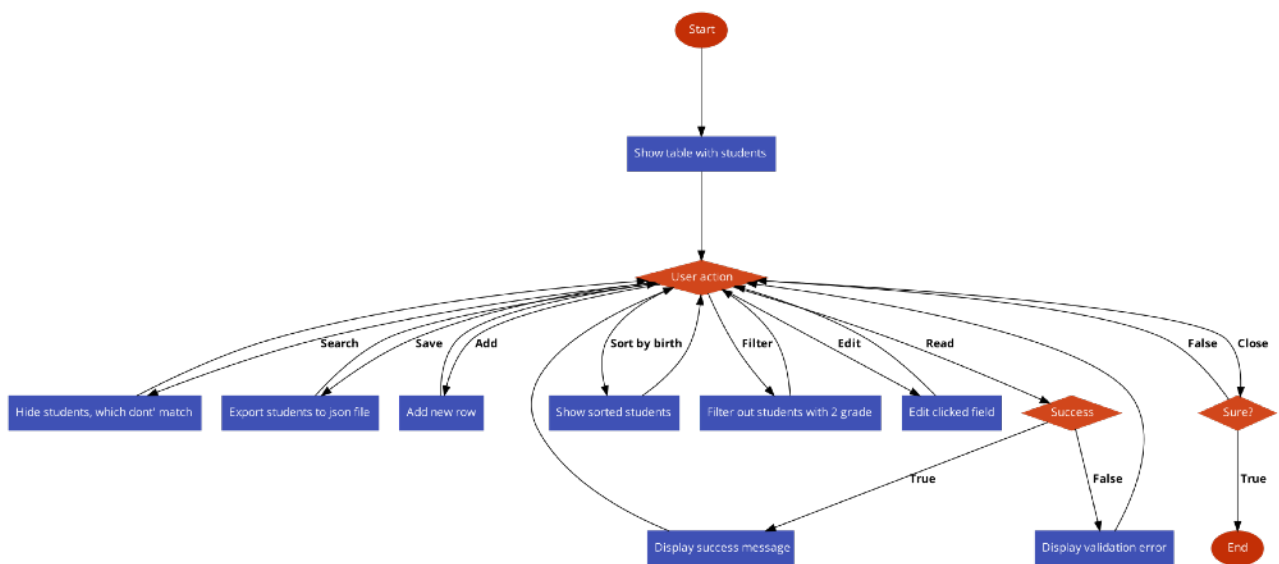


Рис. 1. Блок-схема головної функції програми

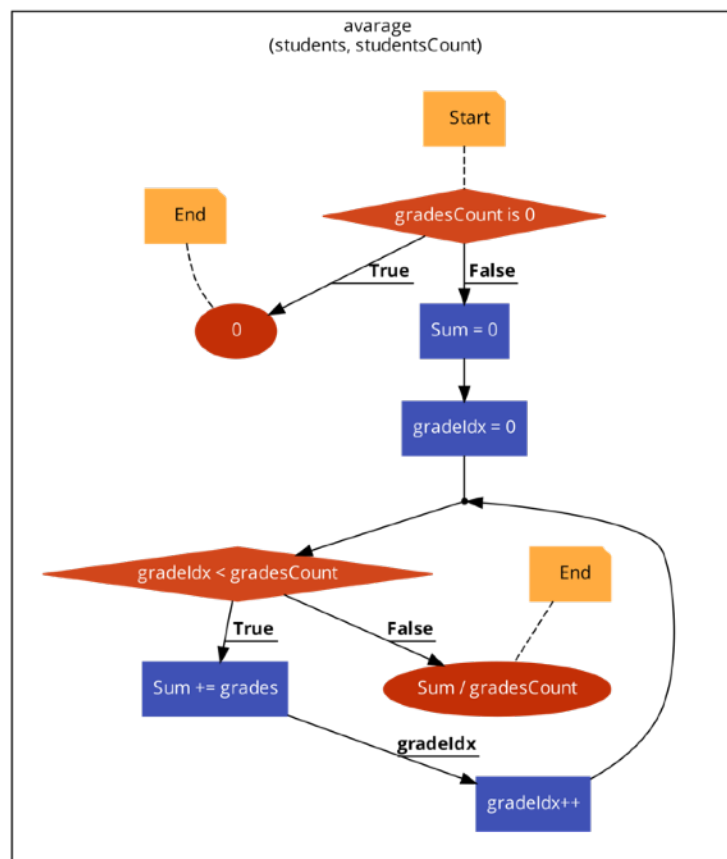


Рис. 2. Блок-схема знаходження середнього балу

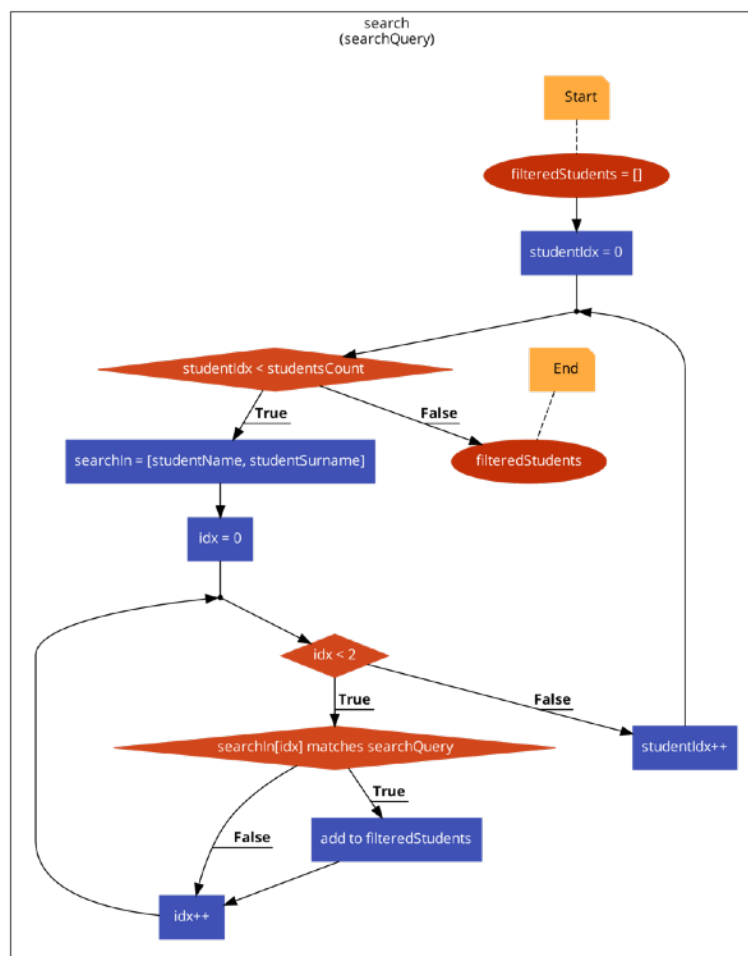


Рис.3. Блок-схема пошуку за іменем або прізвищем

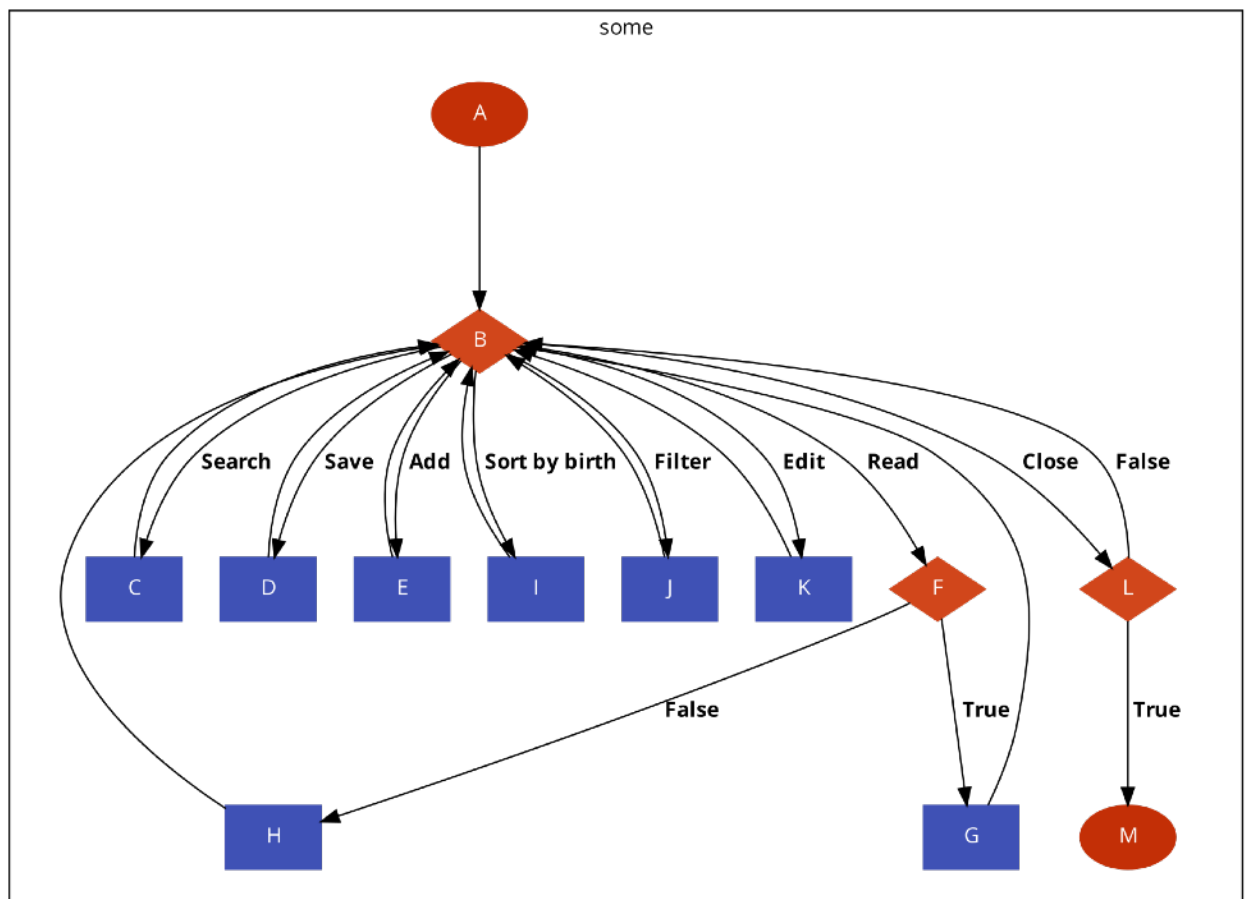


Рис. 4. Граф управління головною функцією

### Таблиця результатів тестування графу управління головною функцією

№ з/п	Маршрут	Результати
1	A-B-L-M	Успішний запуск програми. Файли не були відкриті. Завершення програми
2	A-B-C-B-L-M	Успішний запуск програми. Пошук пустого списку. Завершення
3	A-B-D-B-L-M	Успішний запуск програми. Повідомлення про неможливість збереження пустого списку. Завершення
4	A-B-E-B-L-M	Успішний запуск програми. Ввід даних про одного студента. Вивід інформації про правильність даних. Заборона зберегти невірні дані. Завершення

5	A-B-I-B-L-M	Успішний запуск програми. Сортуння пустого списку не викликає помилок. Завершення
6	A-B-J-B-L-M	Успішний запуск програми. Фільтрація пустого списку не викликає помилок. Завершення
7	A-B-K-B-L-M	Успішний запуск програми. Функція редагування для пустого списку не доступна Завершення
8	A-B-F-G-B-L-M	Успішний запуск програми. Завантаження студентів з файлу у правильному форматі. Вивід списку на екран. Завершення
9	A-B-F-H-B-L-M	Успішний запуск програми. Спроба завантаження студентів з файлу у неправильному форматі. Вивід детальної помилки. Завершення
10	A-B-F-G-B-C-B-L-M	Успішний запуск програми. Завантаження студентів з файлу у правильному форматі. Вивід списку на екран. Перевірка пошуку за імям та прізвищем. Завершення
11	A-B-F-G-B-D-B-L-M	Успішний запуск програми. Завантаження студентів з файлу у правильному форматі. Вивід списку на екран. Збереження даних у файл. Завершення
12	A-B-F-G-B-I-B-L-M	Успішний запуск програми. Завантаження студентів з файлу у правильному форматі. Вивід списку на екран. Перевірка сортування за датою народження. Завершення
13	A-B-F-G-B-J-B-L-M	Успішний запуск програми. Завантаження студентів з файлу у правильному форматі. Вивід списку на екран. Перевірка фільтрації за студентами без оцінки 2. Завершення

14	A-B-F-G-B-K-B-L-M	Успішний запуск програми. Завантаження студентів з файлу у правильному форматі. Вивід списку на екран. Перевірка редагування даних про студента. Збереження неправильних даних заборонено, вивід про помилки. Завершення
----	-------------------	--

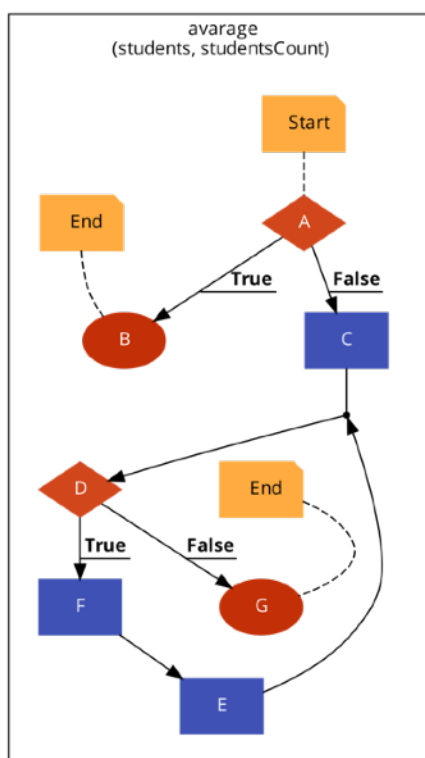


Рис. 5. Граф управління функцією знаходження середнього балу

### Таблиця результатів тестування графу управління функцією

№ з/п	Маршрут	Результати
1	A-B	Немає оцінок - повернення 0
2	A-C-D-G	Недосяжний шлях, оскільки має бути принаймі одна ітерація (тому що є принаймі одна оцінка)

3	A-C-D-F-E-C-D-G	Опрацювання будь якої кількості оцінок, вираховування сумми всіх оцінок. Повернення сумми поділеної на кількість оцінок
---	-----------------	---

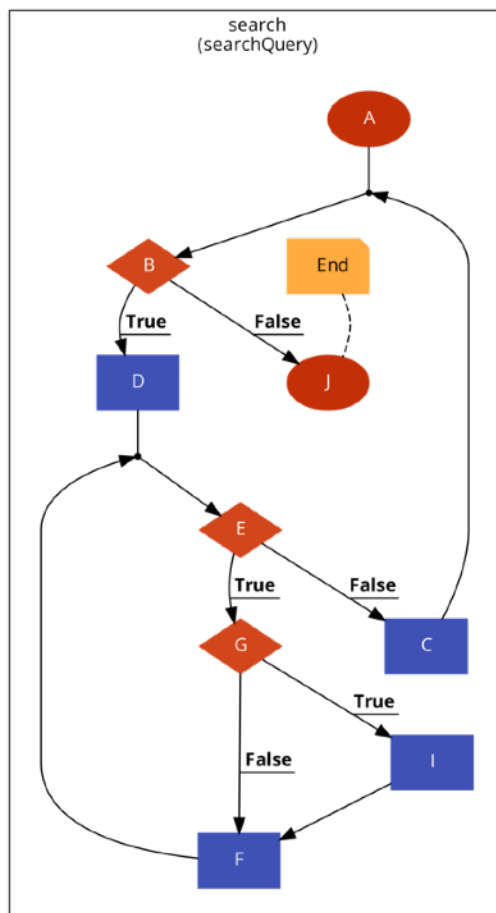


Рис. 6. Граф управління функцією пошуку за іменем або прізвищем

### Таблиця результатів тестування графу управління функцією

№ з/п	Маршрут	Результати
1	A-B-J	Немає студентів - повернення пустого списку
2	A-B-D-E-C-*	Шлях неможливий оскільки відбувається ітерація по масиву довжини 2 (неможливо зразу вийти з циклу)
3	A-B-D-E-G-F-*-J	Ітерування за списком студентів з перевіркою співпадіння частин стрічок імені та прізвища з запитом, не дає співпадінь - повернення пустого списку

4	A-B-D-E-G-I-F-E-C-B-J	Ітерування за списком студентів з перевіркою співпадіння частин стрічок імені та прізвища з запитом, є співпадіння - повернення нового списку зі студентами для яких виконалася умова.
---	-----------------------	--

**Висновок:** В ході виконання лабораторної роботи №3 я навчився правильно проводити та оформляти документацію етапу тестування програмного забезпечення та засвоїла ці навички. Виконав поставлене завдання, тестуючи власний код, виявив і виправив помилки в ньому.



## Додаткові матеріали

**Students Panel**

Search by name or surname

Name	Surname	Grades	Avarage	Birth
Ray	Baird	3,1,4,4,2,1	2.50	26-12-2003
Munoz	Rivers	3,5,3,3,5,2	3.50	19-12-2003
Elizabeth	Horne	2,3,2,4,4,5	3.33	25-11-2003
Aileen	Gill	4,2,2,1,3,2	2.33	07-10-2003
Tyler	Valencia	2,2,2,5,4,4	3.17	04-10-2003
Harvey	Garcia	3,2,3,1,2,4	2.50	17-09-2003
Rosemary	Sweet	4,3,1,4,4,5	3.50	12-08-2003
Kirk	Sutton	3,5,2,2,2,2	2.67	05-08-2003
Wallis	Maldonado	4,1,2,4,1,1	2.17	23-07-2003
Dixie	Mayo	3,5,5,1,3,5	3.67	11-07-2003

< 1 2 3 4 5 ... 10 > 10 / page

Рис. Запуск програми з даними

**Students Panel**

Search by name or surname


Name	Surname	Grades	Avarage	Birth
 No Data				

Рис. Запуск програми без даних

Students Panel

Load

Save

Add

Search by name or surname

Name	Surname	Grades	Avarage	Birth
Lyons	Bruce	<input type="text" value="4,5,2,5,1,4,0"/> <div>Grade must be in range of 1-5</div>	3.50	02-01-1998

Рис. Валідація при редагуванні оцінок

Grades	Avarage	Birth																																																	
4,5,2,5,1,4,4	3.57	<input type="text" value="02-01-1998"/> <div> <div>&lt;&lt; &lt; Jan 1998 &gt; &gt;&gt;</div> <table> <tr> <td>Su</td><td>Mo</td><td>Tu</td><td>We</td><td>Th</td><td>Fr</td><td>Sa</td> </tr> <tr> <td>28</td><td>29</td><td>30</td><td>31</td><td>1</td><td>2</td><td>3</td> </tr> <tr> <td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td> </tr> <tr> <td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td> </tr> <tr> <td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td> </tr> <tr> <td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td> </tr> <tr> <td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td> </tr> </table> <div>Today</div> </div>	Su	Mo	Tu	We	Th	Fr	Sa	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7
Su	Mo	Tu	We	Th	Fr	Sa																																													
28	29	30	31	1	2	3																																													
4	5	6	7	8	9	10																																													
11	12	13	14	15	16	17																																													
18	19	20	21	22	23	24																																													
25	26	27	28	29	30	31																																													
1	2	3	4	5	6	7																																													
3,2,5,4,1,3	3.00																																																		
1,2,2,3,5,5	3.00																																																		
4,4,1,5,1,2	2.83																																																		
1,2,4,5,3,4	3.17																																																		
5,1,5,5,3,3	3.67																																																		

Рис. Введення дати народження за допомогою віджета

Students Panel

Load

Save

Add

we

Name	Surname	Grades	Avarage	Birth
Garner	West	3,4,5,3,5,4	4.00	20-05-1998
Doreen	Bowen	1,3,3,2,1,4	2.33	24-06-2002
Carr	Caldwell	3,3,2,5,3,4	3.33	15-04-2003
Rosemary	Sweet	4,3,1,4,4,5	3.50	12-08-2003

<

1

>

Рис. Пошук студентів за іменем та прізвищем

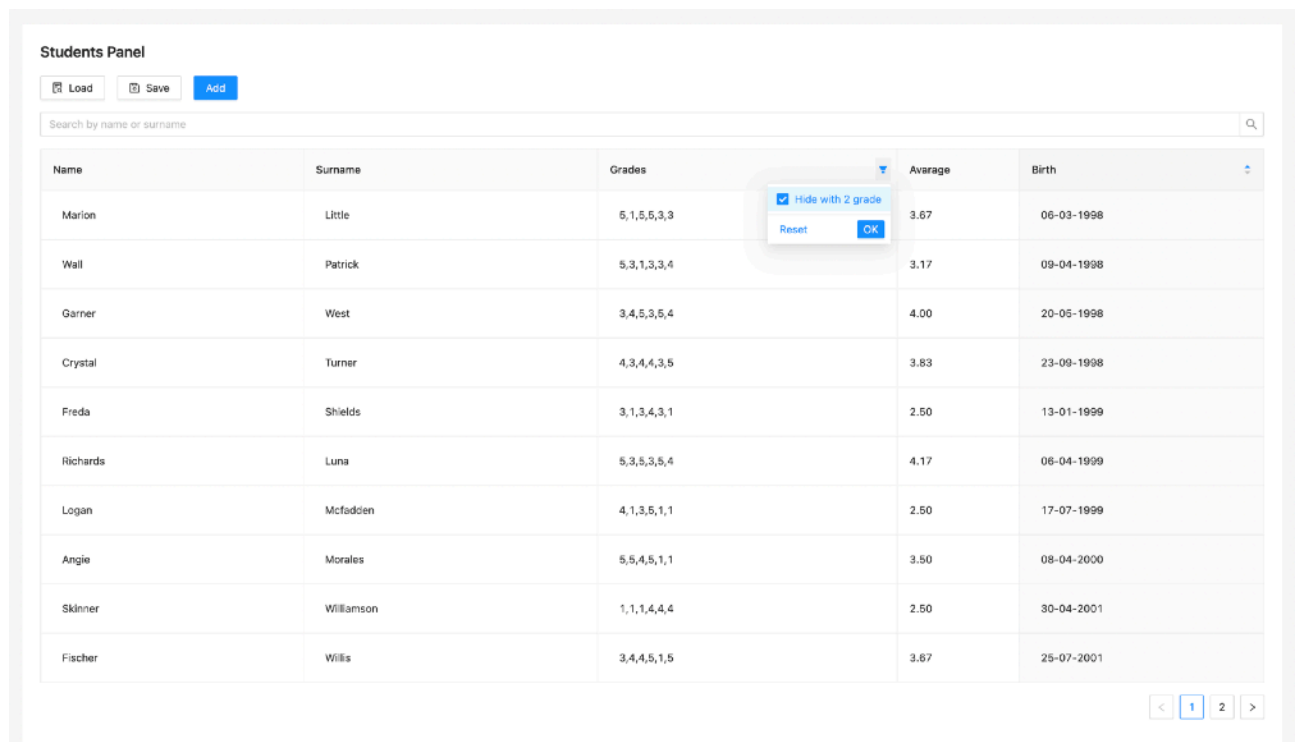


Рис. Фільтрація студентів з оцінкою 2

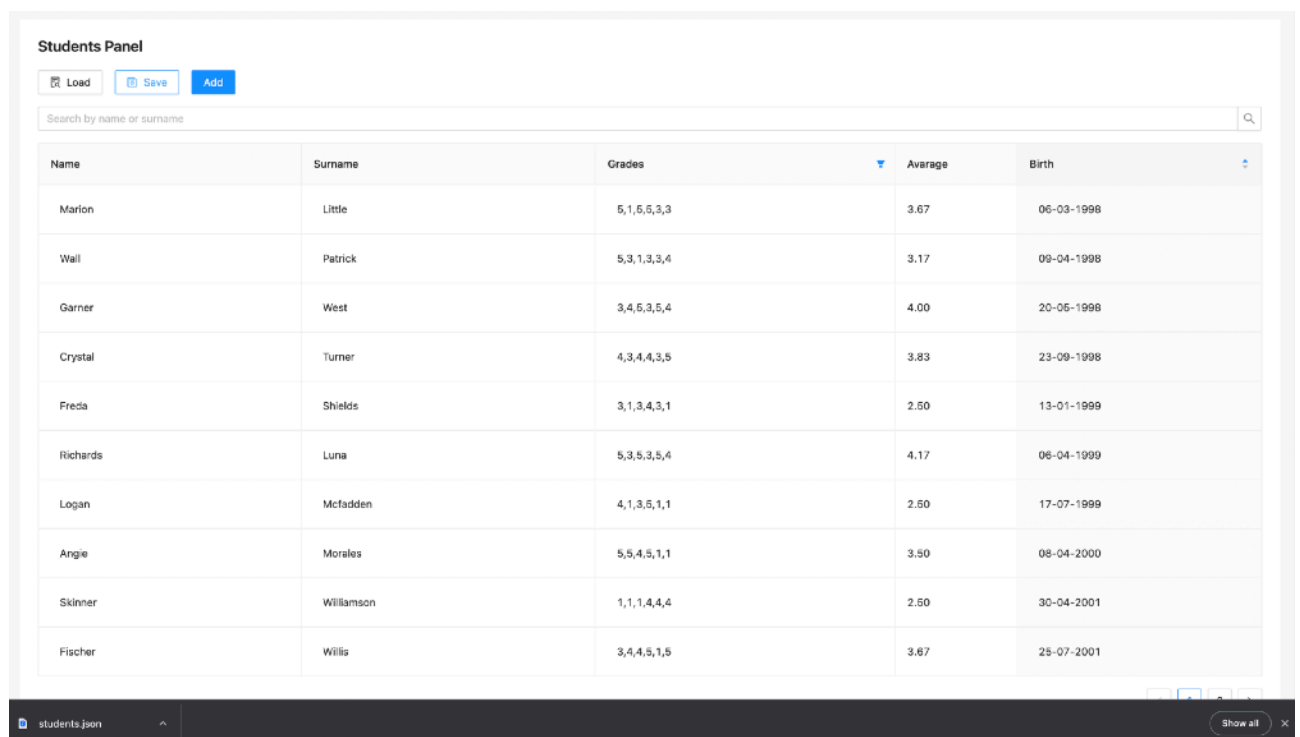


Рис. Збереження списку студентів у файл в форматі json

Students Panel

Load

Save

Add

Search by name or surname

Name	Surname	Grades	Avarage	Birth
<div><div></div><div>No Data</div></div>				

✖

Type error: Must be string

✕

Please fix the detected error: at /99/name

✖

Grade must be in range of 1-5

✕

Please fix the detected error: at grades of  
{  
 "key": 99, "name": 40, "surname": "McKnight",  
 "grades": [15, 4, 5, 2, 4, 1], "birthDate": "Tue Jul  
18 2030 06:31:23 GMT+0300 (Eastern  
European Summer Time)"}  
}

✖

Birth date can't be in the future

✕

Please fix the detected error: at birthDate of  
{  
 "key": 99, "name": 40, "surname": "McKnight",  
 "grades": [15, 4, 5, 2, 4, 1], "birthDate": "Tue Jul  
18 2030 06:31:23 GMT+0300 (Eastern  
European Summer Time)"}  
}

Рис. Повідомлення про помилки при зчитуванні з файлу (Перевірка типів даних та валідація дати народження і оцінок)