

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра систем штучного інтелекту



Звіт

про виконання лабораторних та практичних робіт блоку № 2

На тему: «Лінійні алгоритми. Розгалужені алгоритми. Умовні та логічні оператори. Змінні. Константи. Типи даних. Розмір типів даних. Ввід вивід. Базові операції та вбудовані функції. Коментарі.»

з дисципліни: «Основи програмування»

до:

ВНС Лабораторної Роботи № 1
Алготестер Лабораторної Роботи № 1
Практичних Робіт до блоку № 2

Виконав:

Студент групи ІІІ-12
Шийка Стефан Андрійович

Львів 2024

Мета: Ознайомлення з базовими елементами програмування в C++, включаючи змінні, типи даних, умови, логічні оператори та введення/виведення, для розробки простих програм із застосуванням лінійних та розгалужених алгоритмів.

Теоретичні відомості: У роботі розглядаються числові системи та їх перетворення, принципи компіляції програм, типи даних, оператори вводу-виводу, використання бібліотек, базові арифметичні та логічні операції, коментарі у коді, створення лінійних алгоритмів та умовних розгалужень на мові C++.

Лабораторні завдання:

- a. Виконання програмування в рамках VNS Lab 1, завдання 1 і 2.
- b. Завдання програмування в Algotester Lab 1, завдання 1.

Джерела:

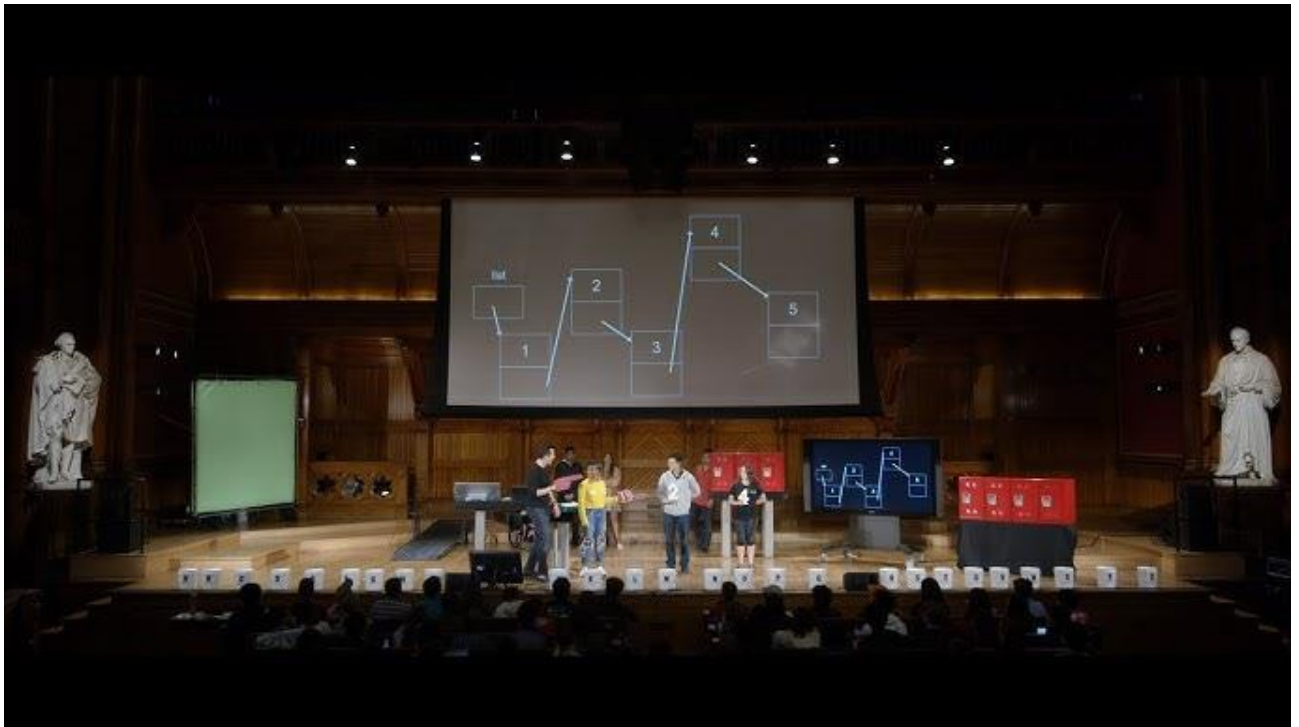
- CS50 lectures and assignments on C, algorithms, computer memory and data structures.
- aCode: C++ syntax, new things from C, like input handling with cin, vectors, bitwise operators.
- The university lectures.

Виконання роботи

Task 1 - Theory Education Activities

Time expected – 1 moth

Spent – 3 month



Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)

Time expected – 1.5 h

Time spent ~ 1.5 h

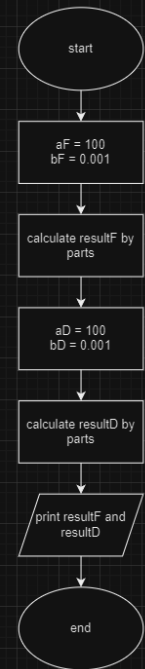
Task 3 - Lab# programming: VNS Lab 1 Task 1

Time expected: 15 min

Spent: 15 min

vns_lab_1_task_1_john_black.cpp > main()

```
1  #include<iostream>
2  #include<math.h>
3
4  using namespace std;
5
6  int main(){
7      float aF = 100;
8      float bF = 0.001;
9      float upperLeftF = pow((aF + bF), 4);
10     float upperRightF = pow(aF, 4) + 4 * pow(aF, 3) * bF + 6 * pow(aF, 2) * pow(bF, 2);
11     float downF = 4 * aF * pow(bF, 3) + pow(bF, 4);
12     float resultF = (upperLeftF - upperRightF) / downF;
13
14     double aD = 100;
15     double bD = 0.001;
16     double upperLeftD = pow((aD + bD), 4);
17     double upperRightD = pow(aD, 4) + 4 * pow(aD, 3) * bD + 6 * pow(aD, 2) * pow(bD, 2);
18     double downD = 4 * aD * pow(bD, 3) + pow(bD, 4);
19     double resultD = (upperLeftD - upperRightD) / downD;
20
21     cout << "Float: " << resultF << endl;
22     cout << "Double: " << resultD << endl;
23 }
```



```
aF = 100
bF = 0.001000000005
upperLeftF = 100004000
upperRightF = 100004000
downF = 4.00001056e-07
resultF = 0

aD = 100
bD = 0.001
upperLeftD = 100004000.06000042
upperRightD = 100004000.06
downD = 4.0000100000000001e-07
resultD = 1.0430786758726462
```

```
Float: 0
Double: 1.04308
```

Пояснення: при використанні float виліляється 4 байти, а на double – 8. Через те, що числа які віднімаються схожі, а різниця між ними не входить до точності float різниця 0, а в double ця мала різниця вміщується і результат не 0. Через використання проміжних обчислень точність зменшується і цифри, які не увійшли в точність просто пропускаються.

Task 4 - Lab# programming: VNS Lab 1 Task 2

Time expected: 10 min

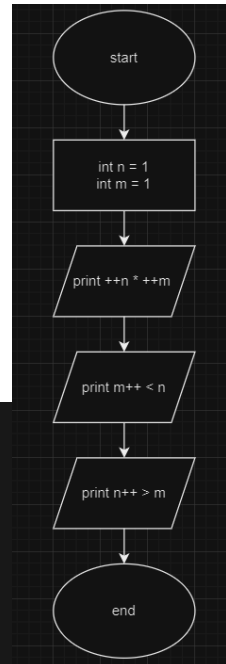
Spent: 15 min

```

vns_lab_1_task_2_john_black.cpp > main()
1  #include<iostream>
2
3  using namespace std;
4
5  int main(){
6      int n = 1;
7      int m = 1;
8      cout << ++n * ++m << endl;
9      cout << (m++ < n) << endl;
10     cout << (n++ > m) << endl;
11
12 }

```

4
0
0



- ☐ Pre-increment (++n): The variable n is incremented first, and then the new value of n is used in the expression.
- ☐ Post-increment (n++): The current value of n is used in the expression, and then n is incremented.

Task 5 - Lab# programming: Algotester Lab 1 Task 1

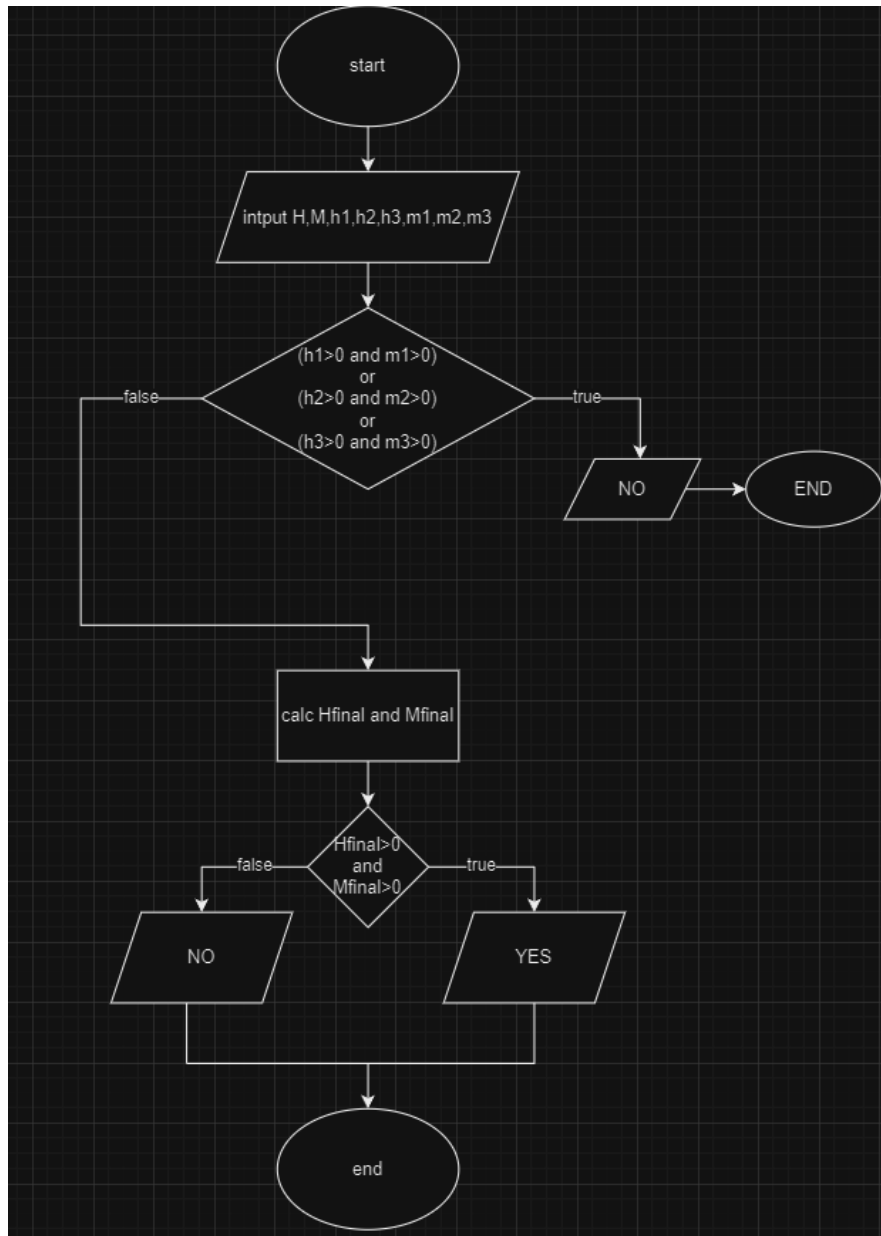
Time expected – 10 min

Spent – 20 min

```

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      long long H, M;
6      cin >> H >> M;
7
8      long long h1, m1, h2, m2, h3, m3;
9      cin >> h1 >> m1;
10     cin >> h2 >> m2;
11     cin >> h3 >> m3;
12
13     if ((h1 > 0 && m1 > 0) || (h2 > 0 && m2 > 0) || (h3 > 0 && m3 > 0)) {
14         cout << "NO" << endl;
15         return 0;
16     }
17
18     long long Hfinal = H - (h1 + h2 + h3);
19     long long Mfinal = M - (m1 + m2 + m3);
20
21     if (Hfinal > 0 && Mfinal > 0) {
22         cout << "YES" << endl;
23     } else {
24         cout << "NO" << endl;
25     }
26 }

```



Created	Problem	Compiler	Result	Time (sec.)	Memory (MiB)	#
a few seconds ago	Lab 1v1 - Lab 1v1	C++ 23	Accepted	0.003	1.277	1809943
4 minutes ago	Lab 1v1 - Lab 1v1	C++ 23	Wrong Answer 3	0.003	1.188	1809941
4 minutes ago	Lab 1v1 - Lab 1v1	C++ 23	Wrong Answer 3	0.003	1.051	1809940
9 minutes ago	Lab 1v1 - Lab 1v1	C++ 23	Wrong Answer 2	0.003	0.945	1809939
9 minutes ago	Lab 1v1 - Lab 1v1	C++ 23	Wrong Answer 2	0.003	1.203	1809938
13 minutes ago	Lab 1v1 - Lab 1v1	C++ 23	Wrong Answer 2	0.002	0.941	1809935
13 minutes ago	Lab 1v1 - Lab 1v1	C++ 23	Wrong Answer 2	0.002	0.926	1809933

Task 6 - Practice# programming: Class Practice Task

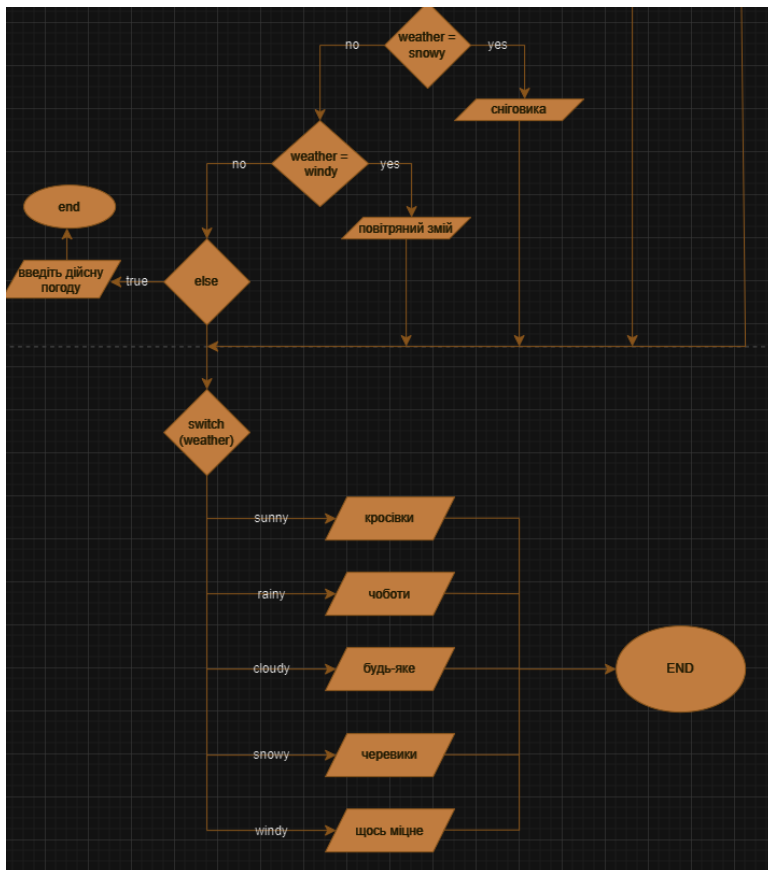
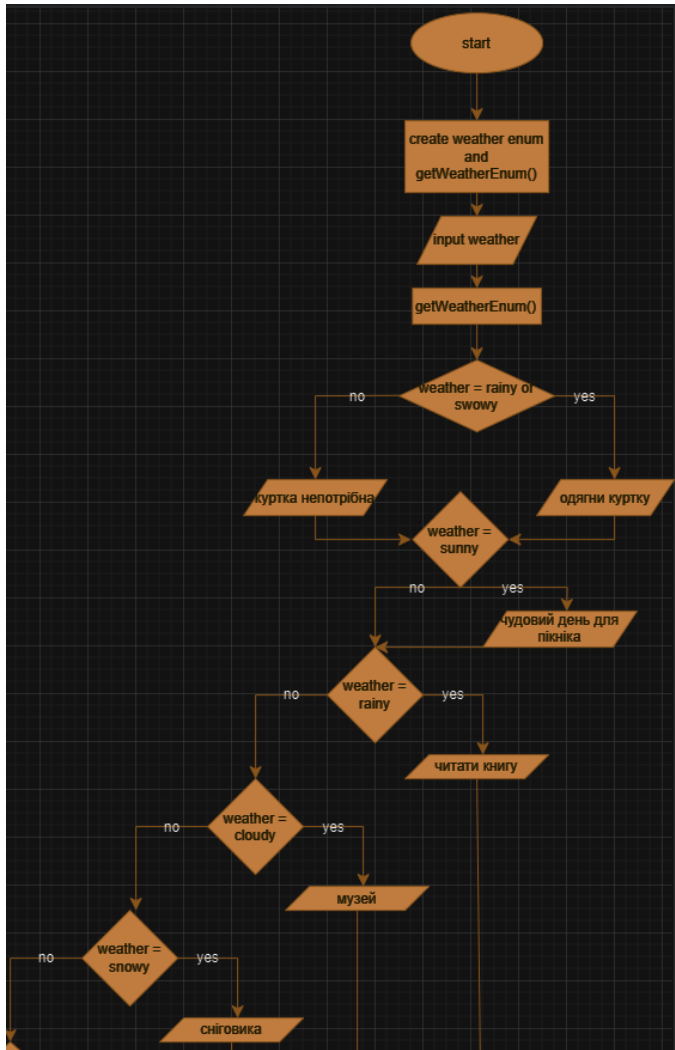
Time expected: 30min

Spent: 1h

```

practice_work_self_algotester_tasks_john_black.cpp > main()
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  enum Weather { SUNNY, RAINY, CLOUDY, SNOWY, WINDY, INVALID };
7
8  Weather getWeatherEnum(const string& weather) {
9      if (weather == "sunny") return SUNNY;
10     if (weather == "rainy") return RAINY;
11     if (weather == "cloudy") return CLOUDY;
12     if (weather == "snowy") return SNOWY;
13     if (weather == "windy") return WINDY;
14     return INVALID;
15 }
16
17 int main() {
18     string weather;
19     cout << "Enter your weather: ";
20     cin >> weather;
21
22     Weather currentWeather = getWeatherEnum(weather);
23
24     if (currentWeather == SNOWY || currentWeather == RAINY) {
25         cout << "Одягни куртку." << endl;
26     } else {
27         cout << "Куртка не потрібна." << endl;
28     }
29
30     if (currentWeather == SUNNY) {
31         cout << "Чудовий день для пікніка!" << endl;
32     } else if (currentWeather == RAINY) {
33         cout << "Ідеальна погода, щоб читати книгу всередині!" << endl;
34     } else if (currentWeather == CLOUDY) {
35         cout << "Може, відвідати музей?" << endl;
36     } else if (currentWeather == SNOWY) {
37         cout << "Як щодо того, щоб зліпити сніговика?" << endl;
38     } else if (currentWeather == WINDY) {
39         cout << "Запустіть повітряного змія, якщо він у вас є!" << endl;
40     } else {
41         cout << "Будь ласка, введіть дійсну погоду." << endl;
42         return 0; // Завершити програму, якщо погода не дійсна
43     }
44
45     switch (currentWeather) {
46     case SUNNY:
47         cout << "Взуй улюблені кросівки!" << endl;
48         break;
49     case RAINY:
50         cout << "Дощові чоботи - гарна ідея!" << endl;
51         break;
52     case CLOUDY:
53         cout << "Сьогодні підходить будь-яке взуття." << endl;
54         break;
55     case SNOWY:
56         cout << "Снігові черевики зігріють ваші ноги!" << endl;
57         break;
58     case WINDY:
59         cout << "Одягніть щось міцне!" << endl;
60         break;
61     default:
62         break;
63     }
64 }

```



Task 7 - Practice# programming: Self Practice Task

Time expected: 1 h

Time spent: 4h+

```
practice_work_self_algotester_tasks_stefan_shyika.cpp > main()
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int n;
7      cin >> n;
8      int count = 0;
9      int rest = 0;
10
11     count += n / 500;
12     rest = n % 500;
13
14     count += rest / 200;
15     rest = rest % 200;
16
17     count += rest / 100;
18     rest = rest % 100;
19
20     count += rest / 50;
21     rest = rest % 50;
22
23     count += rest / 20;
24     rest = rest % 20;
25
26     count += rest / 10;
27     rest = rest % 10;
28
29     count += rest / 5;
30     rest = rest % 5;
31
32     count += rest / 2;
33     rest = rest % 2;
34
35     count += rest;
36
37     cout << count;
38 }
```

Created	Compiler	Result	Time (sec.)	Memory (MiB)	Actions
a few seconds ago	C++ 23	Accepted	0.003	1.172	View
7 minutes ago	C++ 23	Wrong Answer 2	0.002	0.926	View

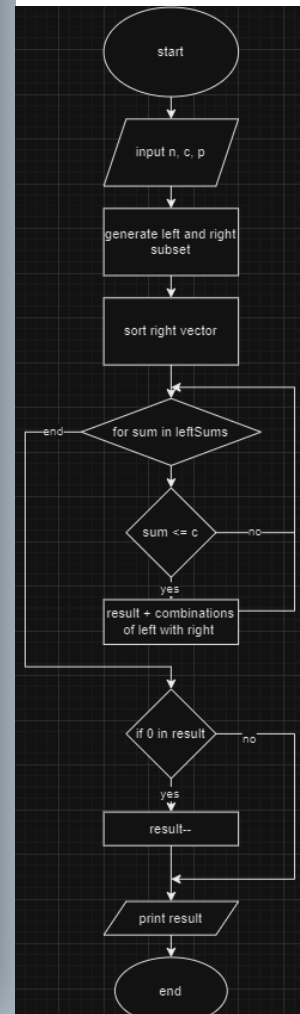
Result	Time (sec.)	Memory (MiB)	Score	#
Accepted	0.003	0.922	-	0001
Accepted	0.002	0.941	-	0002
Accepted	0.002	0.945	-	0003
Accepted	0.002	0.953	-	0004
Accepted	0.002	0.961	-	0005
Accepted	0.002	1.066	-	0006
Accepted	0.002	0.941	-	0007
Accepted	0.002	0.938	-	0008
Accepted	0.002	0.922	-	0009
Accepted	0.002	0.941	-	0010
Accepted	0.003	0.957	-	0011
Accepted	0.003	1.141	-	0012
Accepted	0.002	1.191	-	0013
Accepted	0.002	0.953	-	0014
Accepted	0.003	0.863	-	0015
Accepted	0.002	1.199	-	0016
Accepted	0.002	0.895	-	0017
Accepted	0.002	0.891	-	0018
Accepted	0.002	1.117	-	0019
Memory Limit	1.349	260.160	-	0020

Showing 1 to 20 of 20 rows

```

1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4 #include <set>
5
6 using namespace std;
7
8 void generateSubsetSums(const vector<long long>& arr, int index, long long currentSum, set<long long>& sums, long long c) {
9     // If the sum exceeds c stop
10    if (currentSum > c) return;
11
12    // If we've reached the end of the array, add the sum to the set
13    if (index == arr.size()) {
14        sums.insert(currentSum);
15        return;
16    }
17
18    // Case 1: Exclude the current element and move to the next
19    generateSubsetSums(arr, index + 1, currentSum, sums, c);
20
21    // Case 2: Include the current element and move to the next
22    generateSubsetSums(arr, index + 1, currentSum + arr[index], sums, c);
23 }
24
25 int main() {
26     int n;
27     long long c;
28     cin >> n >> c;
29
30     vector<long long> p(n);
31     for (int i = 0; i < n; i++) {
32         cin >> p[i];
33     }
34
35     vector<long long> left(p.begin(), p.begin() + n / 2);
36     vector<long long> right(p.begin() + n / 2, p.end());
37
38     set<long long> leftSums, rightSums;
39     generateSubsetSums(left, 0, 0, leftSums, c);
40     generateSubsetSums(right, 0, 0, rightSums, c);
41
42     // Convert the rightSums set to a sorted vector for binary search
43     vector<long long> rightSumsVec(rightSums.begin(), rightSums.end());
44     sort(rightSumsVec.begin(), rightSumsVec.end());
45
46     int result = 0;
47     for (long long sum : leftSums) {
48         if (sum <= c) {
49             // Find how many sums in rightSums can be paired with this sum
50             result += upper_bound(rightSumsVec.begin(), rightSumsVec.end(), c - sum) - rightSumsVec.begin();
51         }
52     }
53
54     //sum == 0 is included by default, subtract 1 if leftSums contains 0
55     if (leftSums.count(0) > 0) {
56         result--;
57     }
58
59     cout << result;
60 }
61
62

```



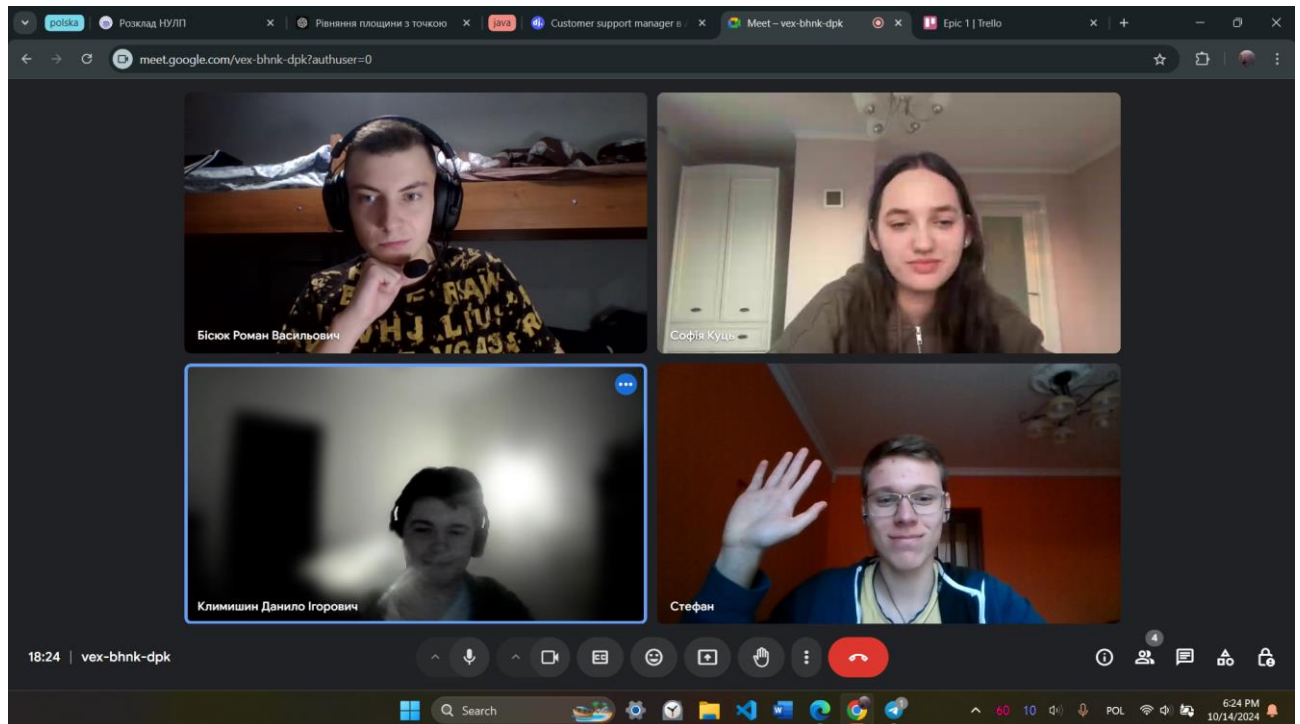
Task 8 - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)

Time expected: 30 min

Spent: 40 min

Task 9 - Results Evaluation and Release

Meets: розібралися з дошкою в Trello, допомогли один одному з написанням практичного завдання та блок-схеми до нього.



Pull: https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/68

Висновок: за цей епiк я навчився швидко та якiсно робити блок-схеми в draw.iо закрiпив базовi знання синтаксису C++ (enums, if, if-else, else, switch). Пригадав базовий функцiонал math.h та переповнення типiв.