

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Завідувач кафедри

_____ Скарга-Бандурова І.С.

« ____ » _____ 2018 р.

ДИПЛОМНИЙ ПРОЕКТ БАКАЛАВРА

ПОЯСНЮВАЛЬНА ЗАПИСКА

НА ТЕМУ:

Програмний модуль прогнозування
витоку небезпечних газів з обладнання

Освітньо-кваліфікаційний рівень “Бакалавр”
Напрямок 6.050101– “Комп’ютерні науки”

Науковий керівник роботи:

(підпис)

Лифар О. К.

(ініціали, прізвище)

Консультант з охорони праці:

(підпис)

ст. викл. Критська Я. О.

(ініціали, прізвище)

Студент:

(підпис)

Кукочка О.О.

(ініціали, прізвище)

Група:

КН-14 д

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітньо-кваліфікаційний рівень бакалавр
Напрямок підготовки 6.050101 Комп'ютерні науки
(шифр і назва)
Спеціальність _____
(шифр і назва)

ЗАТВЕРДЖУЮ:

Завідувач кафедри _____
І.С. Скарга-Бандурова
« _____ » _____ 20__ р.

**ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ БАКАЛАВРА**

Кукоцці Олегу Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Програмний модуль прогнозування витоку
небезпечних газів з обладнання
- керівник проекту (роботи) Лифар О. К.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
- затверджені наказом вищого навчального закладу від «14» 05 2018 р. № 117/48
2. Термін подання студентом роботи 15.06.2018 р.
3. Вихідні дані до роботи матеріали переддипломної практики
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналіз роботи та технічних вимог на виконання програмного модуля; вибір засобів для побудови; створення програмного модулю.
Надання рекомендацій по охороні праці та пожежній безпеці.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Електронні плакати

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	ст. викл. Кафедри КНІ Критська Я. О.		

7. Дата видачі завдання _____

Керівник _____

(підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Огляд літератури з теми ДП і постановка задачі	14.05.18-19.05.18	
2	Дослідження матеріалів	20.05.18-25.05.18	
3	Розробка технічного завдання	22.05.18-28.05.18	
4	Розрахунки	26.05.18-02.06.18	
5	Дослідження результатів	03.06.18-06.06.18	
6	Розробка розділу охорона праці	07.06.18-09.06.18	
7	Оформлення електронних плакатів	10.06.18-12.06.18	
8	Оформлення пояснювальної записки	13.06.18-15.06.18	

Студент _____

(підпис)

Кукочка О.О.

(прізвище та ініціали)

Науковий керівник _____

(підпис)

Лифар О. К.

(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту бакалавра: 75 с., 11 рисунка, 6 табл., 19 бібліографічних джерел посилань, 2 додатки.

Об'єкт розробки: Програмний модуль вирахування показників витоку газу з обладнання.

Мета роботи: аналіз та вибір найбільш оптимальних засобів для створення програмного забезпечення для розрахунку витоку газу із обладнання.

В проекті виконано:

1) вибір засобів та технологій що будуть використовуватись для створення програмного модулю;

2) реалізація програмного модулю вирахування показників витоку газу з обладнання та перевірка на працездатність;

3) аналіз заходів з техніки безпеки.

Отримано наступні результати:

В результаті був створений програмний модуль, який вираховує масу газу по відношенню до часу та виводить концентрацію, діаграми результатів.

Практичне значення, галузь застосування роботи: розробка програмного модулю вирахування показників витоку газу з обладнання використовується для аналізу ризиків в промисловості.

Ключові слова:

ПРОГРАМНИЙ МОДУЛЬ, ВИТОК ГАЗУ, АНАЛІЗ РИЗИКУ, РЕЗЕРВУАР, С#, ДІАГРАММИ.

Умови одержання дипломного проекту: СНУ ім. В. Даля, пр. Центральний 59-А, м. Сєвєродонецьк, 93400.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ СКОРОЧЕНЬ І ТЕРМІНІВ.....	7
ВСТУП.....	8
1 ПОСТАНОВА ТА АНАЛІЗ ЗАДАЧІ	9
1.1 Теорія витоку газу через отвір.....	9
1.2 Аналіз відомих рішень.....	13
1.2.1 Програмний комплекс Toxi+Risk5	13
1.2.2 Програмний комплекс PHAST.....	16
1.2.3 Калькулятор “Гідравлічні розрахунки у трубопроводах”	17
1.3 Постановка технічного завдання.....	18
1.4 Обґрунтування програмного засобу для розробки програмного модулю витоку газу	18
1.5 Вимоги до програмного модулю	20
1.5.1 Вимоги до інтерфейсу	20
1.5.2 Вимоги до надійності.....	20
1.5.3 Вимоги до працездатності.....	21
1.5.4 Вимоги до комп’ютера	21
Висновки до розділу 1	21
2 ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛЮ РОЗРАХУНКУ ВИТОКУ ГАЗУ. 23	
2.1 Математична модель фізичного процесу	23
2.1.1 Алгоритм виконання розрахунків	23
2.1.2 Розрахунок витоку газу з обладнання через отвір.....	24
2.1.3 Розрахунок витоку газу з обладнання через систему місцевих і гідравлічних опорів.....	27
2.2 Реалізація структури програмного модуля.....	29
2.2.1 Вхідні данні для проведення розрахунків	29
2.2.2 Визначення допустимих меж для введення вхідних даних.....	31
2.2.3 Класи та компоненти	31
2.2.4 Приклад виконання розрахунків в класі Form1	32
2.2.5 Побудова діаграм	34

2.3 Реалізація інтерфейсу програмного модулю розрахунку витоку газу ...	34
2.3.1 Загальні положення.....	34
2.4 Перевірка отриманих результатів.....	39
Висновки до розділу 2	40
3 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	41
3.1 Загальні питання з охорони праці	41
3.1.1 Правові та організаційні основи охорони праці	42
3.1.2 Організаційно-технічні заходи з безпеки праці	43
3.2 Аналіз стану умов праці	43
3.2.1 Вимоги до приміщення.....	44
3.2.2 Вимоги до організації робочого місця.....	44
3.2.3 Навантаження та напруженість процесу праці	45
3.3 Виробнича санітарія.....	46
3.3.1 Аналіз небезпечних та шкідливих факторів при розробці виробу	46
3.3.2 Пожежна безпека.....	48
3.3.3 Електробезпека.....	48
3.4 Гігієнічні вимоги до параметрів виробничого середовища.....	49
3.4.1 Мікроклімат	49
3.4.2 Освітлення	50
3.4.3 Вентилювання.....	52
Висновки до розділу 3	52
ВИСНОВОКИ.....	53
ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	54
ДОДАТКИ.....	56
ДОДАТОК А.....	56
ДОДАТОК Б	68

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

DLL – (Dynamic Link Library) – динамічна бібліотека Windows

DNV – (Det Norske Veritas) – норвежська компанія, що розробляє програмне забезпечення

HBO – небезпечні виробничі об'єкти

ППС – паливно-повітряна суміш

ВСТУП

На сьогоднішній день потрібно замислитись про безпеку в галузі промисловості, а точніше про безпеку робітників на підприємствах, а також за наше навколишнє середовище. Прикладом можна привести хімічні заводи, котельні, ТЕЦ, міські газові мережі. Серед нафтових і природних газів найбільш небезпечним є сірководень, який викликає параліч серця та органів дихання. Незважаючи на те, що велика увага приділяється рівню безпеки та перевіркам обладнання, аварії стаються щороку, в результаті чого галузь зазнає значних збитків. Ось чому існувала потреба в програмному забезпеченні, яке може попередньо моделювати виток газу з обладнання.

Функції програми включають в себе обчислення кількості викидів та відображення витрат газу в часі, оцінки ризику запалювання газу та оцінки ризику для працівників підприємства. Ця програма може працювати через отвір і через систему трубопроводів і місцевих опорів. Щоб почати обчислення, потрібно ввести початкові параметри, після чого програма буде обробляти їх і відображати результати розрахунків у формі діаграм та текстового звіту.

На підставі отриманих результатів моделювання можна зробити висновки про те, як підвищити якість та надійність обладнання та запобігти виникненню аварій, а також як зберегти життя та здоров'я працівників підприємства в різних сферах ризику. Аналіз результатів роботи програмного модуля дозволить зберегти навколишнє середовище, заощадити сили і час для ручних розрахунків, зробіть цю програму як матеріально вигідною, так і доцільною в роботі для розрахунку ризиків на небезпечних підприємствах.

Метою даного дипломного проекту є розробка програмного модулю який прогнозує виток небезпечних газів з обладнання

1 ПОСТАНОВА ТА АНАЛІЗ ЗАДАЧІ

1.1 Торія витоку газу через отвір

Для виконання розрахунків приймемо наступні допущення і обмеження:

- Витік газу відбувається з ємкості (рис. 1.1.) постійного і відомого обмеженого об'єму V_{cap} при відомих початкових умовах.

- В процесі витоку газу тиск в ємкості регулюється двома процесами: втратою маси газу з ємкості в результаті витікання через отвір з відомою (відносно невеликою) площею F_2 в атмосферу і додатковим приходом маси газу в ємкість через отвір з іншою відомою площею F_1 .

- Всі процеси відбуваються при постійній температурі.

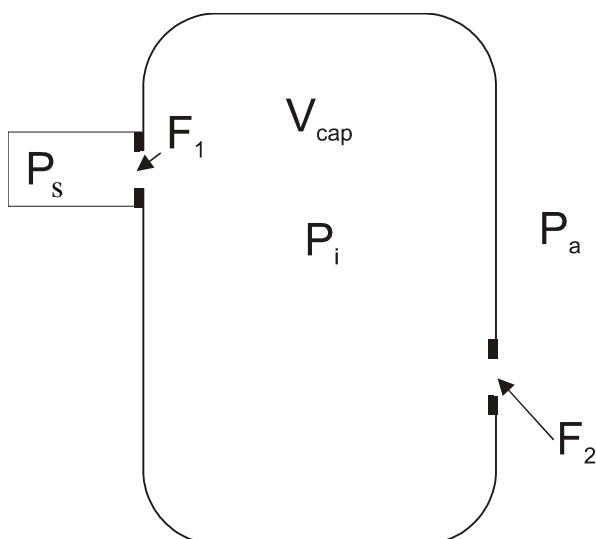


Рисунок 1.1 - Схема витоку газу з ємкості

- Початковий тиск в ємкості задається експертом і дорівнює P_s . Атмосферний тиск дорівнює P_a . Тиск джерела P_s залишається постійним протягом всього часу.

- Джерело постачання газу може бути перекритим протягом заданого експертом часу.

Розглянемо задачу витоку газу з ємкості обмеженого об'єму. При цьому масову витрату будемо розраховувати у відповідності до формули (1.1):

$$\frac{dM(t)}{dt} = \mu \cdot F \cdot \psi_i \sqrt{\frac{P_i}{v_i}}, \quad (1.1)$$

де $\frac{dM(t)}{dt}$ - масова витрата, кг/сек;

μ - коефіцієнт витрати отвору;

F - площа перетину отвору витоку, м²;

P_i - тиск в ємкості в i -й момент часу, Па;

v_i - питомий об'єм в ємкості в i -й момент часу, м³/кг.

Коефіцієнт ψ_i шукаємо як:

$$\psi = \sqrt{2 \left(\frac{k}{k-1} \right) \left[(\beta)^{\frac{2}{k}} - (\beta)^{\frac{k+1}{k}} \right]}, \quad (1.2)$$

де β вибирається з умов:

$$\beta = \begin{cases} \frac{P_1}{P_2} \text{ якщо } \left(\frac{2}{k+1} \right)^{\frac{k}{k-1}} \leq \frac{P_1}{P_2} \\ \left(\frac{2}{k+1} \right)^{\frac{k}{k-1}} \text{ якщо } \left(\frac{2}{k+1} \right)^{\frac{k}{k-1}} > \frac{P_1}{P_2} \end{cases}, \quad (1.3)$$

де P_2 - тиск в просторі, звідки відбувається витікання, Па;

P_1 - тиск в просторі, куди відбувається витікання, Па;

В кінцевих різницях масову витрату з ємкості можна записати як:

$$M_{\tau+1} = M_{\tau} - \mu_2 \cdot F_2 \cdot \psi_{\tau} \sqrt{\frac{P_{\tau}}{v_{\tau}}} \cdot \Delta\tau, \quad (1.4)$$

де M_{τ} - маса газу в ємкості в попередній момент часу, кг;

$M_{\tau+1}$ - маса газу в ємкості в розрахунковий момент часу, кг;

індекси i замінені на τ для позначення приналежності кроку рахунку за часом.

Так як

$$v_{\tau} = \frac{R \cdot T}{M \cdot P_{\tau}}, \quad (1.5)$$

де R - універсальна газова постійна, $\frac{\text{Дж}}{\text{моль} \cdot \text{град}}$;

M - молекулярна маса витоку газу, кг / моль;

T - температура в ємкості, К;

P_{τ} - тиск в ємкості в попередній момент часу, Па,

то вираз для масової витрати з ємкості можна записати як:

$$M_{\tau+1} = M_{\tau} - \mu_2 \cdot F_2 \cdot \psi_{\tau} \cdot P_{\tau} \cdot \sqrt{\frac{M}{R \cdot T}} \cdot \Delta\tau; \quad (1.6)$$

Знову установлений тиск в ємкості розрахуємо як:

$$P_{\tau+1} = \frac{M_{\tau+1} \cdot R \cdot T}{M \cdot V}, \quad (1.7)$$

де V - обсяг ємкості в м^3 .

Після цього необхідно визначити «підживлення» з джерела. Додаткова маса, що приходить із джерела буде розрахована як:

$$\Delta M_{\tau} = \mu_1 \cdot F_1 \cdot \psi_{1\tau} \cdot P_s \cdot \sqrt{\frac{M}{R \cdot T}} \cdot \Delta \tau \quad (1.8)$$

При виборі β для виділення з ємкості в атмосферу:

$$P_1 = P_a, \quad P_2 = P_{\tau}; \quad (1.9)$$

Для виділення з джерела в ємкість:

$$P_1 = P_{\tau+1}, \quad P_2 = P_s. \quad (1.10)$$

Остаточну масу речовини в ємкості присвоюємо як:

$$M_{\tau+1} = M_{\tau+1} + \Delta M_{\tau}. \quad (1.11)$$

Перераховуємо тиск в ємкості з урахуванням додаткової маси:

$$P_{\tau+1} = \frac{M_{\tau+1} \cdot R \cdot T}{M \cdot V}; \quad (1.12)$$

Розрахунок проводимо до зазначеного експертом часу закінчення процесу або ручної зупинки.

1.2 Аналіз відомих рішень

1.2.1 Програмний комплекс Toxi+Risk 5

Звернемо увагу на пакет програмного забезпечення TOXI + Risk 5 (рис. 1.2.) який дозволяє візуалізувати результати розрахунків на планах місцевості, виконаних у векторних та растрових форматах (dxf, dwg, bmp, jpg), оцінити

кількість людей, що потрапили в небезпечні райони, які загинули, а також частотних полів, що перевищують заданий рівень надлишкового тиску та імпульсу з декількох джерел небезпеки для різних сценаріїв аварії, для розрахунку колективних, індивідуальних та соціальних ризиків. Результати моделювання надаються у вигляді графічних діаграм.

За допомогою програмного пакета можна вирішити наступні завдання розрахунку:

– Розрахунок показників ризику на території небезпечного виробництва та за його межами:

1) Розрахунок та візуалізація територіального потенційного ризику;
 2) Побудова розділів потенційного ризику;
 3) Розрахунок колективного, індивідуального, соціального (діаграма F-N) ризику нещасних випадків;

4) Розрахунок індивідуального та соціального ризику пожежної небезпеки;

5) Оцінка можливої кількості загиблих та постраждалих внаслідок нещасних випадків у НВО (Небезпечні виробничі об'єкти);

6) Розрахунок наслідків аварій.

– Розрахунок вибухостійкості будівель та споруд:

1) Детерміністичний підхід до обґрунтування вибухозахищеності будівель та споруд на основі розрахунку ударних зон під час вибуху паливних резервуарів;

2) Імовірнісний підхід до обґрунтування вибухозахисту будівель та споруд на основі кількісного аналізу ризику та частоти перевищення надлишкового тиску та імпульсу ударних хвиль при вибухах паливних резервуарів. Розрахунок територіального поля частот надмірного надлишкового тиску ударних хвиль при вибухах паливних резервуарів, побудова діаграм F-P;

– Здатність до витіснення центру вибуху хмари паливного резервуару з урахуванням дрейфу за рахунок вітру;

– Розрахунок пожежної небезпеки у виробничих та невиробничих будівлях;

- Моделювання розсіювання небезпечних речовин в атмосфері;
- Моделювання вибухів хмари ППС;
- Розрахунок наслідків теплових ефектів від пожежі в протоці, вогненна куля, аварійних факелів;
- Розрахунок зон теплової дії стаціонарних факельних систем;
- Розраховує масову швидкість відтоку горючих газів, зони газового забруднення та зони, що піддаються дії, під час аварій на магістральних газопроводах;
- Розрахунок розміру перелому та частота їх виникнення на основних магістральних газопроводах;
- Розрахунок зон можливого ушкодження фрагментами під час аварійного руйнування ємнісного обладнання.

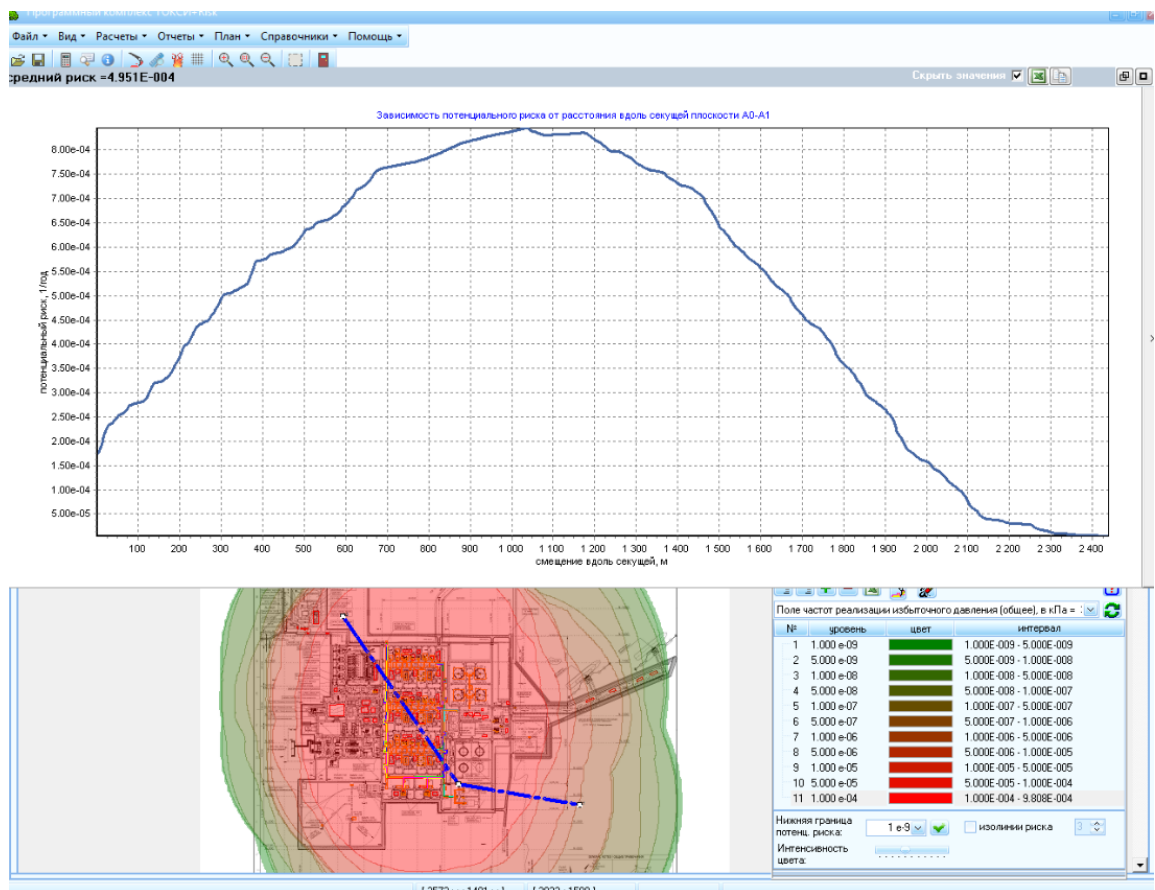


Рисунок 1.2 - Иллюстрация интерфейсу TOXI+RISK5

Серед недоліків варто відзначити високу вартість програмного комплексу, а саме 400 000 грн. за ліцензію на один рік. Забагато вхідних даних робить розрахунки трудомістким. Правильності введених даних висококваліфіковані працівники повинні перевіряти, що означає збільшення часу підготовки персоналу для роботи з пакетом програмного забезпечення TOXI + Risk 5.

Підсумовуючи, ми можемо сказати, що це високоякісний, але дорогий програмний продукт, який потребує значних знань у предметній області та наявності потужного обчислювальної техніки, щоб уникнути довгих розрахунків та зависань, що є неприйнятним у умовах підвищеної небезпеки на промислових об'єктах.

1.2.2 Програмний комплекс PHAST

Програмний комплекс PHAST (рисунок 1.3.) було впроваджено однією із європейських фірм DNV для виконання розрахунків оцінки ризиків, які можуть виникати в результаті викидів токсичних речовин на небезпечних підприємствах.

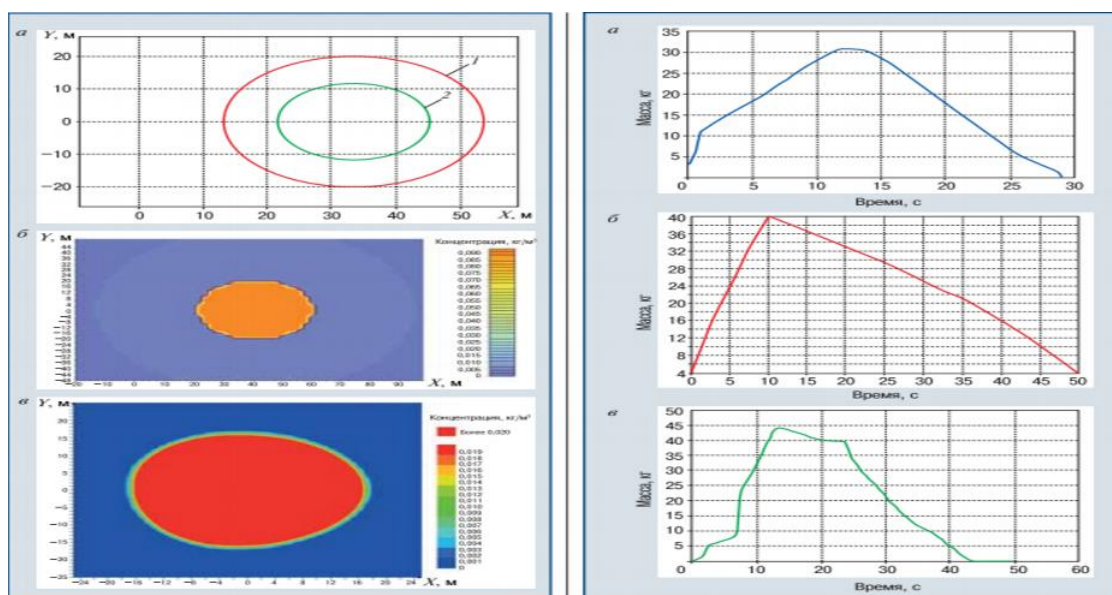


Рисунок 1.3 – Розрахунки, виконані за допомогою програми PHAST

Функції програми включають в себе обчислення можливості викидів шкідливих речовин та графічне відображення концентрацій речовин на території розповсюдження хмари, оцінки ризику загорання речовини та проведення розрахунків індивідуального та колективного ризиків. Обчислення виконуються як для окремого отвору, так і для системи трубопроводів та місцевих опорів. Введення початкових даних виконується висококваліфікованим експертом. Також є можливість вибору методів розрахунків. Результати розрахунків відображаються у вигляді діаграм та текстового звіту.

Підводячи підсумок, можна зробити висновок, що програмний комплекс PHAST - це функціональний інструмент із візуальним інтерфейсом, але необхідність кваліфікованого користувача із знанням фізико-математичних моделей, які можуть правильно вводити вхідні параметри та аналізувати результати моделювання. Даний програмний продукт не має програмного супроводу і не дає можливість користувачеві ознайомитись із фізико-математичними моделями, які закладено в розрахунках. У зв'язку з цим програма PHAST не відповідає вимогам для швидкої підготовки користувачів та простоти використання.

1.2.3 Калькулятор "Гідравлічні розрахунки у трубопроводах"

Ця програма може обчислити об'єм газу з систем трубопроводів при наступних вхідних параметрах:

- за значенням внутрішнього умовного діаметру трубопроводу;
- за довжиною трубопроводу;
- за падінням тиску;
- за коефіцієнтом опору.

Вона розраховує не діаметр аварійного отвору, а об'єм газу згідно заданих параметрів діаметру труби, а значить об'єм витoku підчас аварії не обчислюється,

тому при зазначеній коефіцієнтом опору пропускній здатності трубопроводу не враховується коефіцієнт місцевого опору.

Тому працювати з резервуаром різної геометричної форми завдяки цьому рішенню не можна, провести розрахунок для детально технічних параметрів обладнання зупиняє замала кількість вхідних даних, в текстовій формі надаються результати розрахунку. В результаті проведеного аналізу було зроблено висновок про те, що цей програмний модуль недоречно використовувати на шкідливих виробництвах завдяки малому функціоналу та недолікам.

1.3 Постановка технічного завдання

За заданими параметрами користувачем які повинні обчислити та вивести результат обчислення програмного модуля розрахунку витoku газу з обладнання у відповідності за поставленою задачею.

Для виконання поставленого завдання необхідно виконати наступні розрахунки:

- за відношенням маси випущеного газу до часу;
- за відношення об'єму газу до часу.
- розподіл концентрації отруйної речовини в залежності від атмосферних параметрів;

Закінчивши всі розрахунки користувачу надаються результати у вигляді діаграми.

Щоб створити програмний модуль потрібно виконати основні етапи:

- Аналіз предметної області, провести огляд аналогів які існують;
- Вибрати технологію і засоби які будуть використані в програмному модулі;
- Реалізувати програмний модуль;
- Перевірити програмну реалізацію на коректне отримання результатів та працездатність.

1.4 Обґрунтування програмного засобу для розробки програмного модулю витоку газу

В якості програмного засобу для реалізації даного програмного модуля вибрано C#.

C# - це мова об'єктно-орієнтованого програмування, тому класи в цьому грають фундаментальну роль. Крім того, всі типи даних C#, як вбудовані, так і визначені користувачем, походять від класу базового об'єкта. Іншими словами, на відміну від Java, де примітивні типи даних відокремлені від типів об'єктів, всі типи даних в C# є класами, і їх можна розділити на дві групи:

- посилальні;
- звичайні.

Зовні посилальні і звичайні типи дуже схожі, так як аналогічно C++ в них можна оголошувати конструктори, поля, методи, оператори і т.д. Однак, на відміну від C++, звичайні типи в C# не дозволяють визначати класи і не підтримують успадкування. Вони описуються за допомогою ключового слова `struct` і в основному використовуються для створення невеликих об'єктів. Довідкові ж типи описуються за допомогою ключового слова `class` і є покажчиками, а екземпляри таких типів посилаються на об'єкт, що знаходиться в купі (heap).

Інтерфейсом в C# є тип посилань, що містить тільки абстрактні елементи, що не мають реалізації. Безпосередньо реалізація цих елементів повинна міститися в класі, похідному від даного інтерфейсу. Інтерфейси C# можуть містити методи, властивості і індексатори, але на відміну, наприклад, від Java, вони не можуть містити константних значень.

Розглядаючи класи мови C#, просто не можна обійти таке «нововведення», як властивості. Треба сказати, що тут відчувається вплив мов Object Pascal і Java, в яких властивості завжди були невід'ємною частиною класів. Що ж являють собою ці самі властивості? З точки зору користувача, властивості виглядають практично

так само, як і звичайні поля класу. Їм можна присвоювати деякі значення і отримувати їх назад. У той же час властивості мають бо більшу функціональність, так як читання і зміна їх значень виконується за допомогою спеціальних методів класу. Такий підхід дозволяє ізолювати призначену для користувача модель класу від її реалізації.

Мова програмування C #, як і платформа .NET, знаходиться в розвитку. Зокрема, найближчим часом можна очікувати появи узагальнених шаблонів, які подібно шаблонів мови програмування C++ дозволять створювати сильно типізовані класи-колекції. У будь-якому випадку мова програмування C # вже цілком сформувався для того, щоб його вивчити і почати застосовувати в реальних додатках.

1.5 Вимоги до програмного модулю

1.5.1 Вимоги до інтерфейсу

Виконуючи засоби програмного інтерфейсу було використано вкладки, які дозволяють експерту переключатися між попередньо підготовленими частинами інтерфейсу.

При цьому враховуються додаткові вимоги:

- Повинна забезпечуватись візуальна сумісність програми з користувацьким інтерфейсом Windows XP;
- Розробити варіанти видозміни та модифікації інтерфейсу програмного забезпечення, керуючись тенденціями розвитку галузі застосування цього програмного забезпечення;
- Надати експерту повну візуалізацію результатів процесу моделювання;
- Інтерфейс повинен бути простим, зрозумілим і доступним в застосуванні.

1.5.2 Вимоги до надійності

Модуль програмного забезпечення призначений для виконання розрахунків аварійних ситуацій, тому основною вимогою до нього є надійність та стабільність виконання операцій, які не повинні спричинити несправності системи. Крім того, результати повинні бути достовірними і відповідати вимогам методик розрахунків.

1.5.3 Вимоги до працездатності

Напруга в електричній мережі, показники вологості і температури в приміщенні впливає на роботу програми та отримання достовірних результатів, це все впливає на роботу комп'ютера.

Блок комп'ютера повинен працювати тільки в режимі 110В 60Гц і 220В 50Гц, температура повинна не бути вищою 35⁰С і нижчою 8⁰С, вологість в повітрі потрібно щоб була 65 – 15%.

Ефективність програми та надійність отриманих результатів визначаються показниками температури та вологості в приміщенні та напругою в електричній мережі, при якій комп'ютер буде працювати без перерви.

1.5.4 Вимоги до комп'ютера

Мінімальні вимоги до системного блоку для гарантованого функціонування програмного модуля:

- Оперативна пам'ять 256 Mb;
- Процесор Intel Pentium;
- Клавіатура, миша;

– Кількість вільного місця на носії ЗМБ.

Висновки до розділу 1

Аналізуючи існуючі програмні засоби для моделювання надзвичайних ситуацій в промисловості, було прийнято рішення розробити програмний модуль для розрахунку витоку газу з обладнання на мові програмування С #, який відповідатиме всім вимогам до роботи на небезпечних промислових та хімічних об'єктах.

Низька вартість надасть можливість використовувати цю програму для більшості підприємств, а зручний інтерфейс не вимагатиме висококваліфікованих працівників. На підставі результатів розрахунків модуля програмного забезпечення для витоку газу з обладнання, можна проаналізувати наслідки можливих аварій, розглянути найбільш небезпечні аварійні сценарії та надати кваліфіковані рекомендації щодо зменшення ймовірності виникнення аварійних ситуацій, зменшити збитки для підприємств та населення.

2 ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛЮ РОЗРАХУНКУ ВИТОКУ ГАЗУ

2.1 Математична модель фізичного процесу

2.1.1 Алгоритм виконання розрахунків

Розроблено модель витоку газової фази від складного устаткування з урахуванням серійних елементів, що забезпечують місцевий та гідравлічний зв'язок.

Газ витікає з ємкості постійного та відомого обмеженого об'єму V_{cap} з відомими початковими умовами.

В процесі витікання тиск у посудині регулюється двома процесами: втратою маси газу з ємкості в результаті витікання через отвір з відомою (відносно невеликою) площею F_2 в атмосферу та в результаті додаткового надходження маси газу в ємкість через отвір з іншою відомою площею F_1 (а) або витоку через систему місцевих та гідравлічних опорів (б) (трубопроводи $h_{l,i}$ і запірні арматури $h_{d,i}$).

Вводиться додаткова умова: всі процеси відбуваються при постійній температурі.

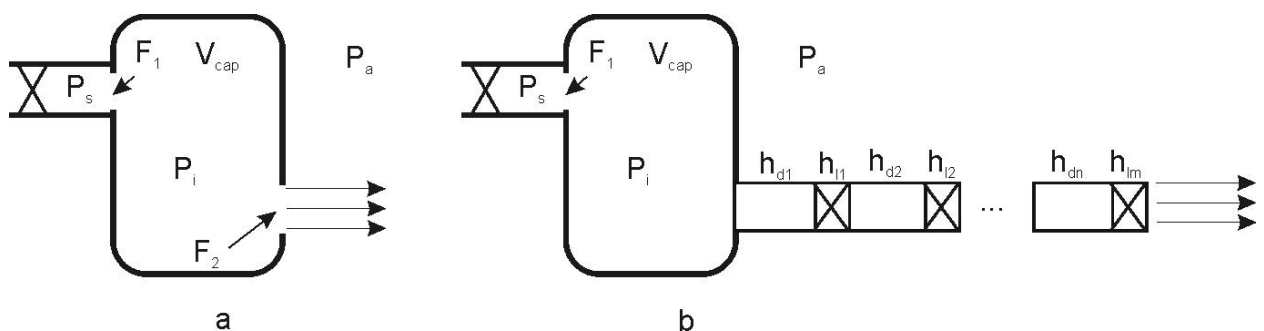


Рисунок 2.1 - Два варіанта витоку з ємкості:

а – через отвір або насадку; б – через систему з'єднань.

Початковий тиск в ємкості встановлюється експертом та дорівнює P_s . Атмосферний тиск дорівнює P_a . Тиск джерела P_s залишається постійним протягом всього часу.

Джерело може бути перекритим протягом часу, який задається експертом.

2.1.2 Розрахунок витоку газу з обладнання через отвір.

Розглянуто задачу витоку газу з ємкості обмеженого об'єму.

Пропонується виконувати розрахунки для показників масової витрати як:

$$\frac{dM(t)}{dt} = \mu \cdot F \cdot \psi_i \sqrt{\frac{P_i}{v_i}}, \quad (2.1)$$

де $\frac{dM(t)}{dt}$ - масова витрата, кг/сек;

μ - коефіцієнт витрати отвору;

F - площа перерізу отвору витоку, м²;

P_i - тиск в ємкості в i -й момент часу, Па;

v_i - питомий об'єм в i -й момент часу, м³/кг.

Коефіцієнт ψ_i визначається як:

$$\psi = \sqrt{2 \left(\frac{k}{k-1} \right) \left[(\beta)^{\frac{2}{k}} - (\beta)^{\frac{k+1}{k}} \right]} \quad (2.2)$$

де β вибирається з умов:

$$\beta = \begin{cases} \frac{P_1}{P_2} \text{ если } \left(\frac{2}{k+1}\right)^{\frac{k}{k-1}} \leq \frac{P_1}{P_2} \\ \left(\frac{2}{k+1}\right)^{\frac{k}{k-1}} \text{ если } \left(\frac{2}{k+1}\right)^{\frac{k}{k-1}} > \frac{P_1}{P_2} \end{cases} \quad (2.3)$$

де P_2 - тиск в просторі, звідки відбувається виток, Па;

P_1 - тиск в просторі, куди відбувається виток, Па;

В кінцевих різницях масову витрату з ємкості можна записати як:

$$M_{\tau+1} = M_{\tau} - \mu_2 \cdot F_2 \cdot \psi_{\tau} \sqrt{\frac{P_{\tau}}{\nu_{\tau}}} \cdot \Delta\tau \quad (2.4)$$

де M_{τ} - маса газу в ємкості в попередній момент часу, кг;

$M_{\tau+1}$ - маса газу в ємкості в розрахунковий момент часу, кг;

індекси i замінені на τ для позначення приналежності кроку рахунка за часом.

Так як:

$$\nu_{\tau} = \frac{R \cdot T}{M \cdot P_{\tau}} \quad (2.5)$$

де R - універсальна газова постійна, $8,31 \frac{\text{Дж}}{\text{моль} \cdot \text{град}}$;

M - молекулярна маса викинутого газу, кг/моль;

T - температура в ємкості, К;

P_{τ} - тиск в ємкості в попередній момент часу, Па,

То вираз для зміни маси в ємкості можна записати як:

$$M_{\tau+1} = M_{\tau} - \mu_2 \cdot F_2 \cdot \psi_{\tau} \cdot P_{\tau} \cdot \sqrt{\frac{M}{R \cdot T}} \cdot \Delta\tau \quad (2.6)$$

Тиск, який буде в наступний момент часу на наступному кроці розрахуємо як:

$$P_{\tau+1} = \frac{M_{\tau+1} \cdot R \cdot T}{M \cdot V} \quad (2.7)$$

де V - об'єм ємкості в м³.

Після цього необхідно визначити «підживлення» з джерела. Додаткова маса, яка приходить з джерела буде розрахована як:

$$\Delta M_{\tau} = \mu_1 \cdot F_1 \cdot \psi_{1\tau} \cdot P_s \cdot \sqrt{\frac{M}{R \cdot T}} \cdot \Delta\tau \quad (2.8)$$

При розрахунку β для витоку з ємкості в атмосферу припускаємо, що $P_1 = P_a$, $P_2 = P_{\tau}$; для витоку з джерела в ємкість: $P_1 = P_{\tau+1}$, $P_2 = P_s$.

Робимо вибір: якщо масова витрата з джерела ΔM_{τ} перевищує продуктивність компресора джерела, то вона стає рівною (кг/с) витраті компресора, яка вводиться користувачем.

Остаточну масу речовини в ємкості присвоюємо як:

$$M_{\tau+1} = M_{\tau+1} + \Delta M_{\tau} \quad (2.9)$$

Знову розрахуємо тиск в ємкості з урахуванням додаткової маси:

$$P_{\tau+1} = \frac{M_{\tau+1} \cdot R \cdot T}{M \cdot V} \quad (2.10)$$

Розрахунок проводиться, поки лічильник часу не перевищить значення, яке задане експертом для закінчення процесу або ручної зупинки.

Підживлення з джерела закінчується розрахунком по досягненні часу перекриття джерела, введеного дослідником (сек).

2.1.3. Розрахунок витoku газу з обладнання через систему місцевих і гідравлічних опорів.

Якщо необхідно виконати розрахунок витoku газу через систему місцевих і гідравлічних опорів, то можна запропонувати наступне рішення:

Припустимо, що виток відбувається через круглі труби (в інших ситуаціях запропоновано метод визначення необхідних коефіцієнтів).

Дослідник повинен ввести (таблично) послідовний набір елементів в системі опору потоку, починаючи від елемента, розташованого в ємкості, закінчуючи елементом, через який відбувається виток в атмосферу. Якщо елемент є гідравлічним опором, дослідник для кожного елемента вводить:

l - довжину трубопроводу в метрах;

d - діаметр в трубопроводу в метрах;

Δ - еквівалентну абсолютну шорсткість труби (мм);

V - кінематичну в'язкість газу ($\text{м}^2/\text{с}$).

Якщо елемент є місцевим опором, необхідно ввести коефіцієнт місцевого опору ζ_l [24], молекулярну масу газу M (кг/моль), температуру газу в середині обладнання T (К).

Для розрахунку необхідно ввести початковий тиск в обладнанні P_s (Па) та атмосферний тиск (за замовчуванням він дорівнює 101000 Па), площа отвору джерела підживлення F_1 (м²),

Перепад тиску між ємкістю витоку і середовищем витоку дорівнює:

$$P_s - P_a = \bar{\rho} \cdot g \cdot \frac{\bar{w}^2}{2 \cdot g} \cdot \left(\sum_{i=1}^m \lambda_i \cdot \frac{l_i}{d_i} + \sum_{j=1}^n \zeta_j \right) \quad (2.11)$$

де \bar{w} - середня швидкість потоку (м/с);

$\bar{\rho} = \frac{P_s + P_a}{2} \cdot \frac{M}{RT}$ - середня щільність потоку (кг/м³);

$\lambda_i = 0.11 \left(\frac{\Delta \cdot 10^{-3}}{d} + \frac{68}{\text{Re}} \right)^{0.25}$ - гідравлічний коефіцієнт тертя (формула

Альтштуля). Так як нас цікавлять суттєві викиди, то припускаємо, що в подальшому будемо мати справу з турбулентними режимами, успішно описаними цією формулою.

$\text{Re} = \frac{\bar{w} \cdot d}{\nu}$ - число Рейнольдса.

Насправді, неправильно визначити гідравлічний опір по середній швидкості потоку, так як на кожній ділянці системи трубопроводів швидкості потоків різні і різні довжини таких ділянок. Таким чином, тиск в системі має перерозподілитися досить швидко (хвиля розрідження повинна пройти по трубі зі швидкістю, близькою до звукової) і на кожній ділянці встановиться свій перепад. Однак, нас цікавлять реальні системи, в яких довільна зміна діаметра трубопроводів або неузгоджені місцеві і гідравлічні опори неприпустимі. З огляду на те, що формула Альтштуля описує турбулентні режими і практично не залежить від перепадів швидкості при значних швидкостях витоку, а в більшій мірі залежить від в'язкості газу і шорсткості стінок труби, обмежимося усередненим наближенням швидкості потоку.

Якщо відомий перепад тиску між ємкістю і середовищем виток, можна обчислити середню швидкість з підсумкового рівняння:

$$\left(\frac{P_s - P_a}{P_s + P_a} \right) \cdot \frac{4RT}{\mu} = \bar{w}^2 \cdot \left(0.11 \cdot \sum_{i=1}^m \left(\frac{\Delta_i \cdot 10^{-3}}{d_i} + \frac{68 \cdot v}{\bar{w} \cdot d_i} \right)^{0.25} \cdot \frac{l_i}{d_i} + \sum_{j=1}^n \zeta_j \right) \quad (2.12)$$

Визначити швидкість \bar{w} можна ітераційно. Так як досягнення критичного режиму виток не дозволяє швидкості течії перевищити швидкість звуку в звичайних трубопроводах, то припускаємо, що середня швидкість витікання не перевищує швидкість звуку.

Масова витрата системи дорівнює :

$$\frac{\Delta M_\tau}{\Delta \tau} = \bar{w} \cdot \frac{\pi \cdot \left(\sum_{i=1}^m d_i \right)^2}{8} \cdot \frac{(P_s + P_a) \mu}{RT} \quad (2.13)$$

Втрата маси з ємкості в одиницю часу розраховується так само, як і в попередньому прикладі. Всі подальші розрахунки проводяться таким же чином.

2.2 Реалізація структури програмного модулю

2.2.1 Вхідні данні для проведення розрахунків

Ця програма створює діаграми відповідно до параметрів, заданих користувачем. Список доступних параметрів, їх назв, позначень в кодї програми та в математичних формулах, область визначення мінімуму до максимуму та одиниці виміру представлені в таблиці 2.1.

Таблиця 2.1 - Вхідні параметри

Найменування	Одиниці виміру	Позначення змінної	Позначення в програмі	Область визначення		Значення за стандартом
				min	max	
Коефіцієнт витрати отвору витоку в атмосферу	-	μ_2	myu2	0.5	1	0.8
Коефіцієнт витрати отвору витоку з джерела в ємкість	-	μ_1	myu1	0.5	1	0.8
Площа перетину отвору витоку в атмосферу	м ²	F_2	F2	0.0001	100	0.02
Площа перетину отвору витоку з джерела	м ²	F_1	F1	0.0001	5	0.02
Тиск в джерелі (початковий тиск в ємкості)	Па	P_s	Ps	101000	101000000	1010000
Атмосферний тиск	Па	P_a	Pa	99000	110000	101000
Обсяг ємкості витоку	м ³	V	V	1	100000	100
Ставлення теплоємностей ($\frac{c_p}{c_v}$)	-	k	k	1	1.6	1.4
Універсальна газова стала	$\frac{Дж}{моль \cdot град}$	R	R	-	-	8,31
Молекулярна маса викинутого газу	кг/моль	M	M	0.001	0.1	0.021
Температура в ємкості	°К	T	T	220	350	293
Час витоку рахунку	сек	t_{end}	T _{end}	1	3600	400
Час перекриття джерела	сек	t_{stop}	T _{stop}	0	3600	300

Вхідні параметри - це змінні, які в подальшому використовуються програмним забезпеченням для вирішення рівнянь та обчислювальних операцій. Ці змінні приймають значення, після того як користувач ввів числові параметри в відповідні текстові поля, але якщо воно буде відповідати областям визначення від найменшого до найбільшого значення.

2.2.2 Визначення допустимих меж для введення вхідних даних

Вводячи дані у текстовому полі програми, експерт може помилитися та вводити невірні значення параметрів. Такі помилкові дані зазвичай можуть привести не лише до неправильних результатів розрахунків, а навіть до збою програмного забезпечення. Щоб не виникало таких ситуацій, було запропоновано виконувати перевірку даних, які вводяться. Якщо вводяться значення, які виходять за допустимі границі, користувач отримує про це попередження.

```
int num = 0;
bool res = int.TryParse(textBox1.Text, out num);
if(res) {
    if (num <= -1 && num >= 10000)
    {
        MessageBox.Show("Неправильное значение, введите от 0 до 10000 .");
        return;
    }
}
```

2.2.3 Класи та компоненти

Програмний модуль був виконаний в середовищі Visual Studio. В основі програми лежить застосування класів та компонентів, які можуть бути вбудовані або можуть бути створені вручну. Структуру класів та компонентів показано на рис.2.2. На початку роботи користувач вводить дані компонентами ComboBox, radioButton і TextBox.

Далі обробляється інформація класу Form1, у відповідності до алгоритму програми.

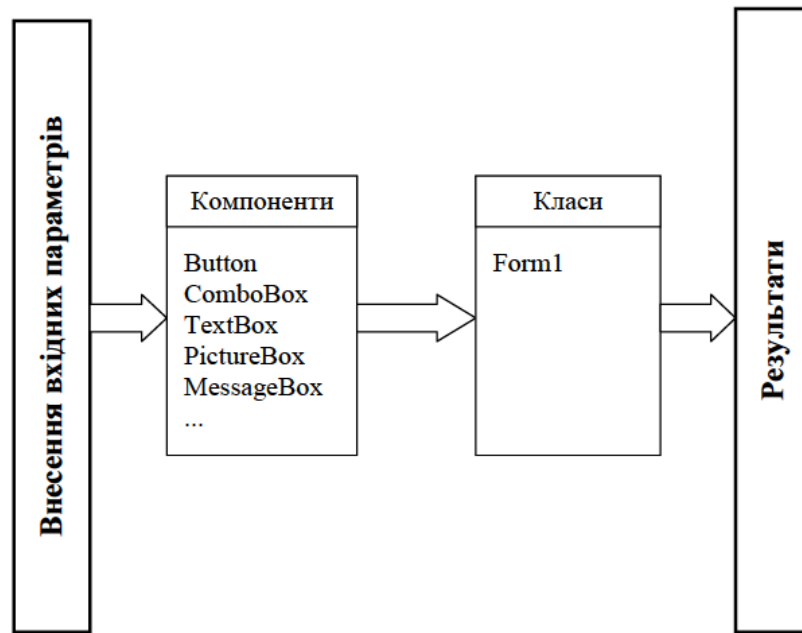


Рисунок 2.2 – Діаграма компонентів та класів

Клас Form1 використовується в програмі, він виконує кілька функцій. Це функція розрахунку, функція для відображення графічних результатів та візуалізації програми. За введення параметрів у програму відповідають інтегровані компоненти

2.2.4 Приклад виконання розрахунків в класі Form1

Всі процеси модуля програми створюються в класі Form1. Розглянемо на прикладі програмного коду який відповідає за розрахунки коефіцієнта витрат, якщо експерт вибере тип витоків "через трубопровідну систему".

```

private void perform_uc_Click(object sender, EventArgs e) .....
{
    double result = 0;
    int n = Convert.ToInt32(input_n.Text);
    int m = Convert.ToInt32(input_m.Text);
    .....
    double S0 = Convert.ToDouble(input_S0.Text); .....
    S0 = Math.Pow(S0, 2.0); // попередньо зводимо в ступінь
    double part_1 = 0;
    double part_2 = 0;
}

double[] in_Si = input_Si.Text.Split(new[] { ';' },
StringSplitOptions.RemoveEmptyEntries).Select(z =>
double.Parse(z.Trim())).ToArray(); ..... double[] in_li =
input_li.Text.Split(new[] { ';' }, StringSplitOptions.RemoveEmptyEntries).Select(z
=> double.Parse(z.Trim())).ToArray(); ..... double[] in_di =
input_di.Text.Split(new[] { ';' }, StringSplitOptions.RemoveEmptyEntries).Select(z
=> double.Parse(z.Trim())).ToArray(); ..... double[] in_LAi =
input_LAi.Text.Split(new[] { ';' },
StringSplitOptions.RemoveEmptyEntries).Select(z =>
double.Parse(z.Trim())).ToArray(); ..... double[] in_SIk =
input_SIk.Text.Split(new[] { ';' },
StringSplitOptions.RemoveEmptyEntries).Select(z =>
double.Parse(z.Trim())).ToArray();
}
}
.....for (int i = 0; i < m; ++i)
{
    .....
    part_1 += in_LAi[i] * (in_li[i] / in_di[i]) * (S0 /
Math.Pow(in_Si[i], 2.0));
}
for (int k = 0; k < n; ++k)
{
    part_2 += in_SIk[k] * (S0 / Math.Pow(in_Sk[k], 2.0));
}
result = Math.Pow(part_1 + part_2, -0.5);
output_result.Text = Convert.ToString(result);
}
}
}

```

По-перше, буде визначено метод і при цьому оголошено змінну з плаваючою точкою, крім того необхідно задати ще й локальні перемінні. Для того, щоб обчислити алгебраїчну суму на мові програмного язика С# потрібно використати цикл. Синтаксис циклу для циклу складається з ініціалізатора, в цьому прикладі початкова змінна - це локальна змінна, умови, за яких цикл буде повторятися, і ітератор, у цьому прикладі його роль полягає в збільшенні значення лічильника циклів.

2.2.5 Побудова діаграм

Для відтворення діаграм на мові програмування C # був використаний інтегрований компонент графіка. Цей компонент включає в себе властивості та методи, наявність великого діапазону яких дозволяє точно відобразити графічні результати. Наприклад, значення титулу дозволяє встановити назву графа, DataSource вказує джерело даних, RenderType вказує метод рендеринга, ширина визначає ширину діаграми у пікселях.

У цьому модулі програми компонент діаграми створює діаграми відповідно до масивів даних, створених у розрахунках, наведених у таблиці 2.2.

Таблиця 2.2 - Опис вихідних параметрів

Розрахунковий тиск в ємкості	Па		Ptau
Масова витрата з ємкості	кг/сек		dMtau/dt
Загальна маса викиду з моменту початку витоку	кг		Mout
Масова витрата з джерела	кг/сек		dM/dt

Також реалізовано можливість зберегти зображення діаграми на комп'ютері. Для цього використовується метод Chart.SaveImage ().

2.3 Реалізація інтерфейсу програмного модулю розрахунку витоку газу

2.3.1 Загальні положення

Пропонується моделювання двох типів витоку: через отвір (при якому не надає істотний вплив гідравлічний опір отвору розгерметизації), через систему трубопроводів і місцевих опорів.

При розрахунках передбачається, що відомий об'єм ємкості витоку, параметри отвору витоку, тиск всередині ємкості та характеристики речовини в ємності. Для обліку можливості надходження речовини з суміжного обладнання, передбачені параметри «приходу» газової фази через площу «приходу» з постійним тиском перед перетином приходу.

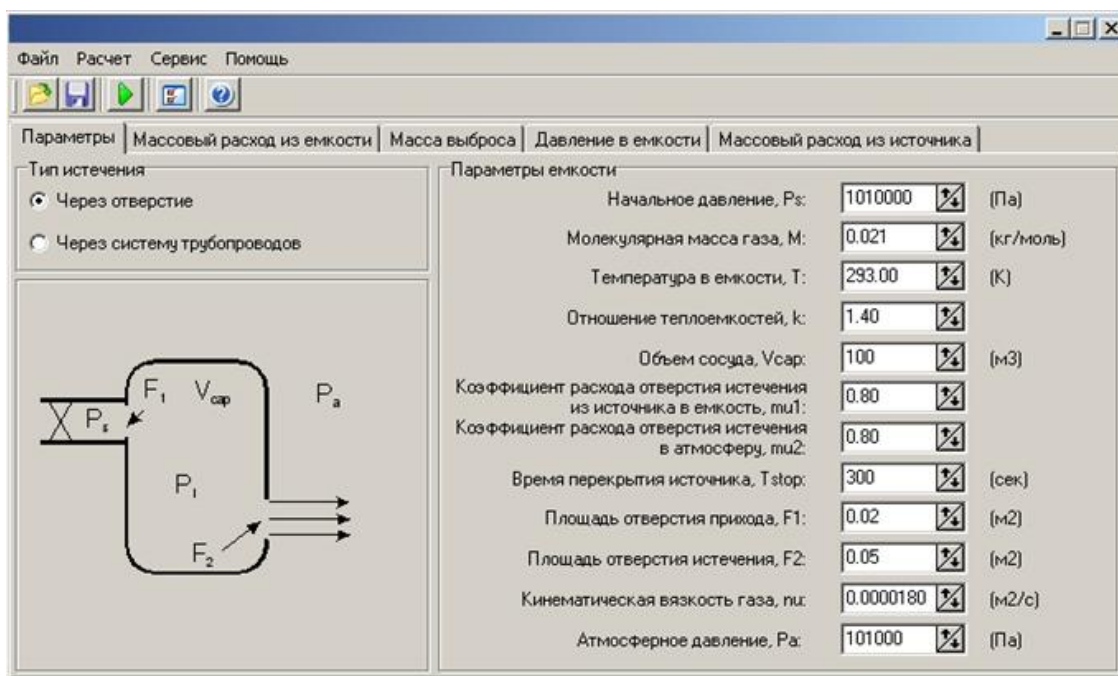


Рисунок 2.3 – Основне вікно модуля «Витоку газу»

Для налаштувань постійних характеристик розрахунків пропонується ввести деякі параметри налаштувань. Їх значення за замовчуванням представлені на рисунку. 2.4.

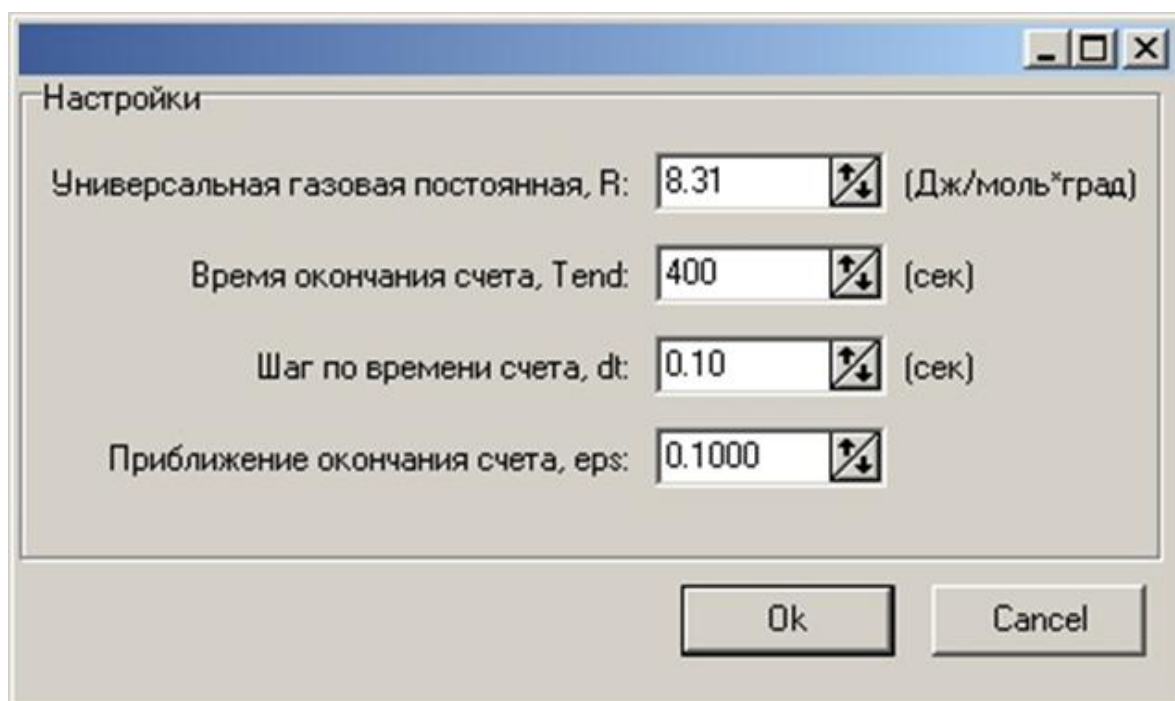


Рисунок 2.4 – Налаштування за замовчуванням модуля «Витоку газу»

Розрахунок проводиться за мінімальним значенням часу перекриття отвору витоку, часу закінчення рахунку або досягнення тиску в ємкості, відповідного «наближенню закінчення рахунку».

Під «Наближенням закінчення рахунку» мається на увазі ситуація, коли поточний тиск в ємкості перевищує атмосферний на величину eps^* (атмосферного тиску).

В якості розрахункових параметрів для побудови графіків при розрахунку моделі витоку через отвір виступають параметри, які змінюються в часі: масова витрата з ємкості, маса витоку, тиск в ємкості, масова витрата з джерела. Для кожного графіка виводиться також середня величина параметра, усереднена інтегрально по всьому часу розрахунку.

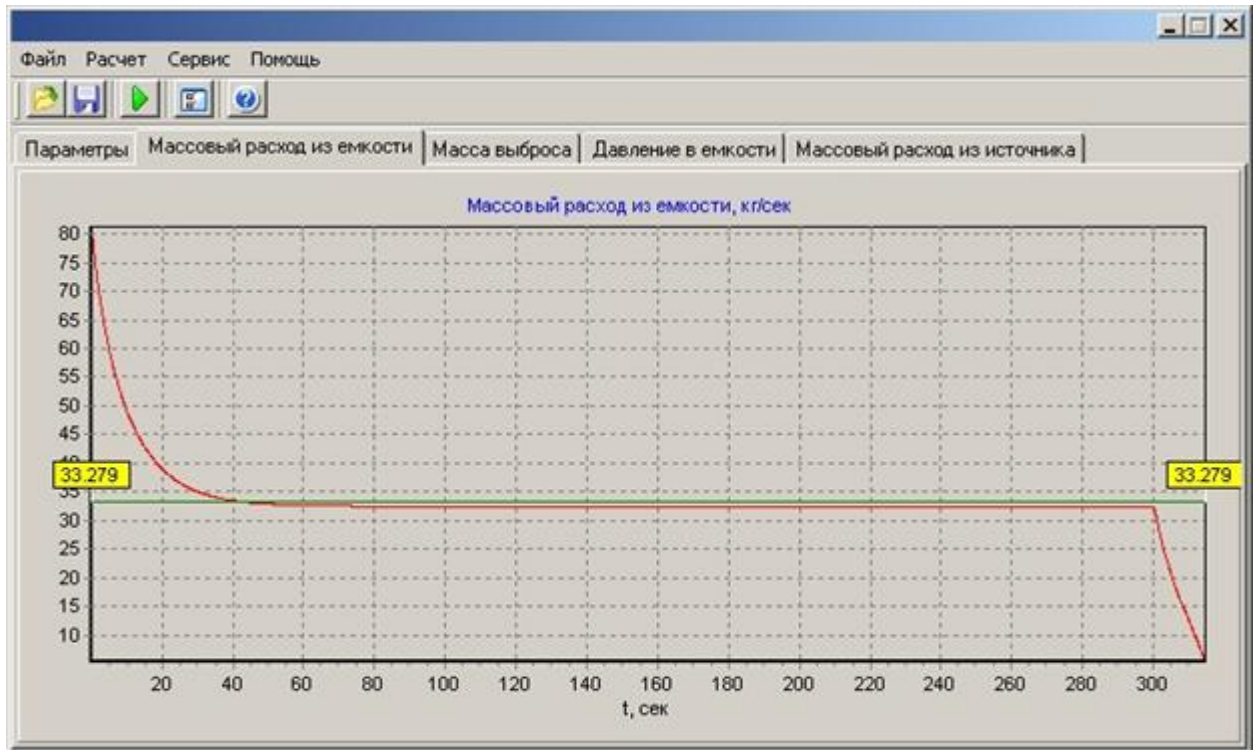


Рисунок 2.5 – Закладка «Масова витрата із ємкості»

Ця величина може бути використана в подальших розрахунках та при розрахунках інших моделей.

У разі витоку через систему гідравлічних і місцевих опорів, необхідно задати набір місцевих (запірна арматура та ін. елементи) і гідравлічних опорів (ділянки трубопроводів).

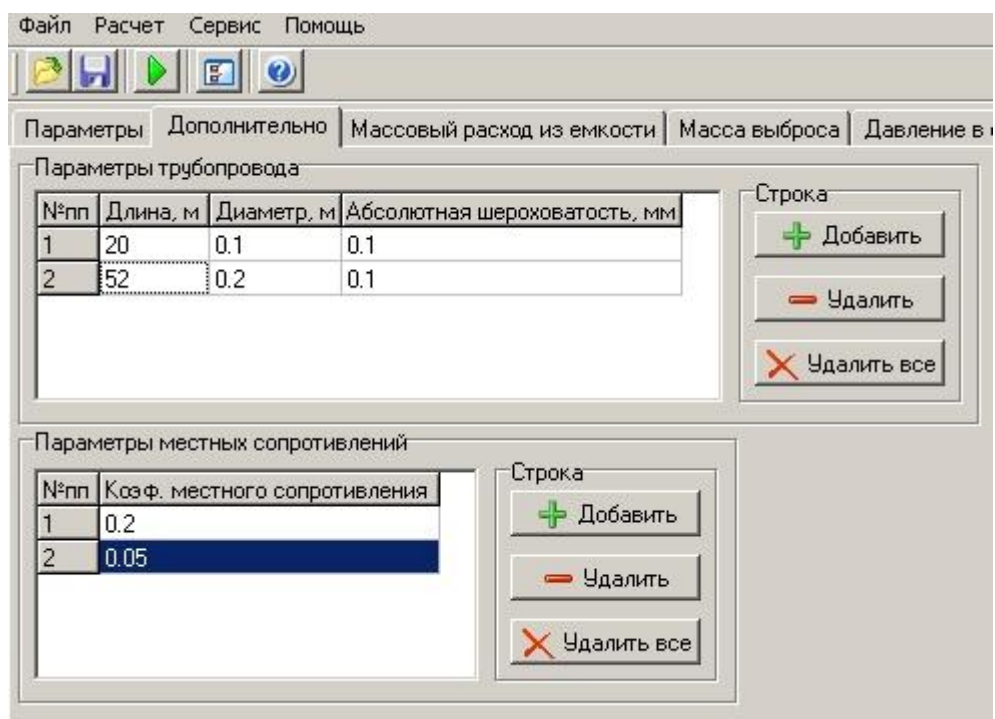


Рисунок 2.6 – Закладка додаткового введення даних при витокі через систему трубопроводів і місцевих опорів.

Коефіцієнти витрати, в'язкість, значення коефіцієнтів місцевих і гідравлічних опорів можна знайти в довідковій літературі та технічних характеристиках обладнання. Ступінь точності введених значень необхідно вибирати виходячи з уявлень впливу параметрів на швидкість процесу витокі.

Пункт меню «Допомога» і «Реєстрація» працюють, так само як і в інших модулях.

Налаштування розрахунку можна зберегти у файлі з розширенням gas за шляхом, який вказує користувач.

Після закінчення розрахунку система повідомляє про це у вікні «Розрахунок закінчено».

2.4 Перевірка отриманих результатів

Отримавши результати розрахунків за допомогою програмного модуля, можемо порівняти їх з аналогічними розрахунками, виконаними в програмному

забезпеченні Mathcad. Діаграми результатів моделювання витоку газу через отвір в залежності від часу, витоку газу через систему трубопроводів і місцевих опорів в залежності від часу та маси газової фази, що вилітає з ємності від часу в середовищі Mathcad (рис.2.7. - 2.8.).

Пропонується моделювання двох типів витоку: через отвір (при якому не надає істотний вплив гідравлічний опір отвору розгерметизації), через систему трубопроводів і місцевих опорів.

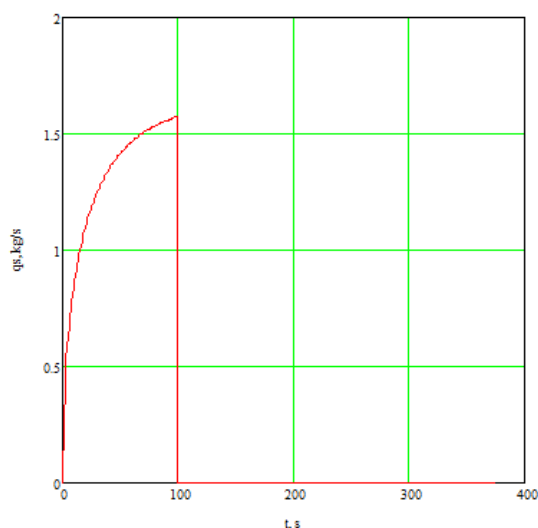


Рисунок 2.7 - Моделювання витоку через отвір в Mathcad

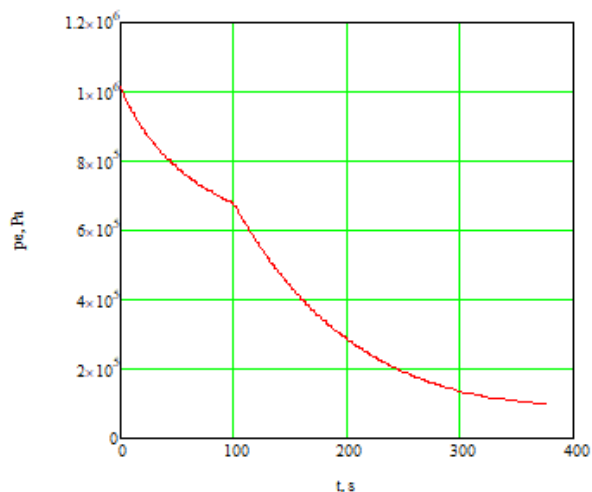


Рисунок 2.8 – Моделювання витоку газу через систему трубопроводів і місцевих опорів в Mathcad

Проведена перевірка правильності отриманих результатів підтверджує, що графік функції отриманий за допомогою модуля програмного забезпечення відповідає графіку функції (в межах допустимої похибки обчислень), що виконується в Mathcad. Це означає, що алгоритм, реалізований в програмному модулі працює вірно, і в результатах розрахунку відсутні відхилення.

Висновки до розділу 2

В результаті вирішення поставленої задачі було отримано програмну реалізацію математичної моделі на мові програмування C#. Представлено модуль програмного забезпечення, який обчислює показники витоку газу з обладнання та формує графічне представлення цього процесу у вигляді діаграм, в залежності від значень вхідних параметрів. Програма була запущена, і отримані результати були проаналізовані. В результаті, програма працює правильно, в процесі обчислень не виникає помилок, не коректних результатів чи збоїв.

3 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даного проекту бакалавра було створення програмного модулю вирахування показників витоку газу з обладнання. Так як процес проектування виконувався у домашніх умовах, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде розроблена комп'ютерна мережа.

3.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» [1] визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними

речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець).

3.1.1 Правові та організаційні основи охорони праці

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави, і особистої відповідальності працівників.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті з Закону України «Про охорону праці» [1] (далі – Закону) законодавство про охорону праці складається з Закону, Кодексу законів про працю України, Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" [2] та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

3.1.2 Організаційно-технічні заходи з безпеки праці

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держнаглядохоронпраці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231/10511 [5].

Також впроваджені організаційні заходи з пожежної безпеки - навчання і перевірку знань відповідно до вимог Типового положення про інструктажі, спеціальне навчання та перевірку знань з питань пожежної безпеки на підприємствах, в установах та організаціях України, затвердженого наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 29.09.2003 N 368, зареєстрованого в Міністерстві юстиції України 11.12.2003 за N 1148/8469 [6].

3.2 Аналіз стану умов праці

Робота над створенням локально програмного модулю проходить в побутовому приміщенні. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером.

3.2.1 Вимоги до приміщення

Геометричні розміри приміщення зазначені у таблиці 3.1.

Таблиця 3.1 – розміри робочого місця

Параметри	Значення
Довжина, м	3
Ширина, м	3
Висота, м	2,5
Площа, м ²	9
Об'єм, м ³	22,5

Згідно з [7] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм – не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

3.2.2 Вимоги до організації робочого місця

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця [8] (табл. 3.2) і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 3.2 – Характеристика робочого місця

Найменування параметрів	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680÷800
Висота простору для ніг, мм	730	> 600
Ширина простору для ніг, мм	660	> 500
Глибина простору для ніг, мм	700	> 650
Висота поверхні сидіння, мм	470	> 400÷500
Ширина сидіння, мм	400	> 400
Глибина сидіння, мм	400	> 400

Висота поверхні спинки, мм	600	> 300
Ширина опорної поверхні спинки, мм	500	> 380
Радіус кривини спинки в горизонтальній площині, мм	400	> 400
Відстань від очей до екрану дисплея, мм	800	> 700÷800

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК та периферійні пристрої можуть бути джерелами електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

За ступенем пожежної безпеки приміщення належить до категорії В. Кабінет оснащений переносним вуглекислотним вогнегасником ВВК-5.

Наявна аптечка для надання долікарської допомоги, а також у кабінеті роблять вологе прибирання та щоденно провітрюють приміщення.

3.2.3 Навантаження та напруженість процесу праці

За фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

Рекомендовано застосування екранних фільтрів, локальних світлофільтрів (засобів індивідуального захисту очей) та інших засобів захисту, а також інші профілактичні заходи [8].

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви тривалістю 15 хв через кожен годину роботи;

3.3 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

3.3.1 Аналіз небезпечних та шкідливих факторів при розробці виробу

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 3.3). Роботу, пов'язану з ЕОП з ВДТ, виконують із забезпеченням виконання , які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують персональні комп'ютери (ПК) та периферійні пристрої.

Таблиця 3.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Нормативні документи
Фізичні		
- підвищена температура поверхонь обладнання	Експлуатація ЕОМ для роботи	[7]
- підвищена або знижена вологість повітря	-//-	[7]
- підвищена або знижена рухливість повітря	-//-	[7]
- підвищений рівень напруги електричної мережі	-//-	[10] [12]
- підвищений рівень статичної електрики	-//-	[10]
- підвищена напруженість електромагнітного поля	-//-	[9]
- недостатність природного світла	Порушення умов праці (вимог до приміщення)	[9]
- недостатнє освітлення робочої зони	Порушення гігієнічних параметрів виробничого середовища	[10]
Психофізіологічні		
- нервово-психічна перевантаження	Розумова робота над проектом	[2] [8]
- фізичні (статичне-сидіння)	Порушення умов праці	[2]

Робочі місця мають відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 [8]. За умов роботи з ПК виникають наступні небезпечні та шкідливі чинники: несприятливі мікрокліматичні умови, освітлення, електромагнітні випромінювання, забруднення повітря шкідливими речовинами та інше.

3.3.2 Пожежна безпека

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем електроживлення ЕОМ, вентиляції і кондиціонування. Небезпека загоряння пов'язана з особливістю комп'ютерів – із значною кількістю щільно розташованих на монтажній платі і блоках електронних вузлів і схем, електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів і транзисторів. Висока щільність елементів в електронних схемах призводить до значного підвищення температури окремих вузлів (80...100°C). Для гасіння пожеж в офісному приміщенні пропонується використовувати порошкові або вуглекислотні вогнегасники, так як вони є універсальними.

Виникнення пожежі можливе, якщо на об'єкті є горючі речовини, окиснювач і джерела запалювання. Вірогідність пожежної небезпеки приймається значною, якщо ймовірна взаємодія цих трьох чинників.

Для відводу теплоти від ЕОМ діє потужна система кондиціонування. Тому кисень, як окиснювач процесів горіння, є в будь-якій точці приміщень обчислювального центру.

Простори усередині приміщень в межах, яких можуть утворюватися або знаходиться пожежонебезпечні речовини і матеріали відповідно до [9] відносяться до пожежонебезпечної зони класу П-Па.

3.3.3 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму

короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три - провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

3.4 Гігієнічні вимоги до параметрів виробничого середовища

3.4.1 Мікроклімат

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. В даному приміщенні проводяться роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, то для нього відповідає категорія робіт Ia. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають нормам ДСН 3.3.6.042-99 [7] і наведені в табл. 3.4.

Таблиця 3.4 – Норми мікроклімату робочої зони об'єкту

Період року	Категорія робіт	С°	Відносна вологість %	V руху повітря, м/с
Холодно	Легка-1 а	22-24	40-60	0,1
Тепло	Легка-1 а	23-25	40-60	0,1

Дане приміщення обладнане системами опалення, кондиціонування повітря або припливно-витяжною вентиляцією. У приміщенні на робочому місці

забезпечуються оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря у відповідності до [7].

Рівні позитивних і негативних іонів у повітрі мають відповідати ДСН 3.3.6.042-99 [7]. Для забезпечення оптимальних параметрів мікроклімату в приміщенні проводяться перерви в роботі співробітників, з метою його провітрювання.

3.4.2 Освітлення

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

У проекті, що виконується, передбачається використовувати суміщене освітлення. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення. Штучне освітлення створюється газорозрядними лампами.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:22 [13]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше $-1/8$, в побутових – $1/10$:

$$S_b = \left(\frac{1}{5} \div \frac{1}{10} \right) \times S_n, \quad (3.1)$$

де S_b – площа віконних прорізів, m^2 ;

S_n – площа підлоги, m^2 .

$S_n = a \cdot b = 3 \cdot 3 = 9 \text{ м}^2$, $S = 1/10 \cdot 9 = 0,9 \text{ м}^2$.

Приймаємо 1 вікно площею $S=0,9 \text{ м}^2$.

Світильники загального освітлення розташовуються над робочими поверхнями в рівномірно-прямокутному порядку. Для організації освітлення в темний час доби передбачається обладнати приміщення, довжина якого складає 5 м, ширина 5 м, світильниками ЛПО2П, оснащеними лампами типа ЛБ (дві по 40 Вт) з світловим потоком 3200 лм кожна.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні.

Розрахунок кількості світильників n виробляється по формулі (3.2):

$$n = \frac{E \times S \times Z \times K}{F \times U \times M}, \quad (3.2)$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, m^2 ; $S = 9 \text{ м}^2$;

Z – поправочний коефіцієнт світильника – 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575;

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 3200лм (для ЛБ-40-2).

Підставивши числові значення у формулу (3.2), отримуємо:

$$n = \frac{300 \times 9 \times 1.1 \times 1.5}{3200 \times 0.575 \times 2} = 1.2 \quad (3.3)$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з 2-х ламп типа ЛБ (по 40 Вт) з світловим потоком 3200 лм кожна.

3.4.3 Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в СНіП.

Також має здійснюватися провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

Висновки по розділу 3

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в кваліфікаційній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника.

Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки.

ВИСНОВКИ

У даному дипломному проекті було створено програмний модуль вирахування показників витoku газу з обладнання, де в програмуванні була застосована мова C# для реалізації математичних моделей фізичних властивостей газу. Завдяки чому було створено програмний інтерфейс, легкий у використанні, зрозумілий користувачу, було досягнуто стабільну роботу програми та швидке виконання.

За розрахунками в програмному модулі було отримано діаграми відношення маси викинутого газу до часу.

В результаті виконаної роботи було створено програмний продукт, який задовольняє всім вимогам, викладеним в технічному завданні.

Було проаналізовано та виконано вимоги з техніки безпеки при роботі з приладами і до робочих місць в розділі "Охорона праці та безпека в надзвичайних ситуаціях".

ПЕРЕЛІК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ

1. Закон України «Про охорону праці»;
2. Стаття 1 Розділу 1 Закону України «Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності <http://www.niss.gov.ua/articles/1302/>© Національний інститут стратегічних досліджень»;
3. НПАОП 0.00-4.12-05 «Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці»;
4. НПАОП 0.00-4.15-98 «Положення про розробку інструкцій з охорони праці»;
5. НПАОП 40.1-1.01-97 «Правила безпечної експлуатації електроустановок»;
6. НАПБ Б.02.005-2003 Типове положення про інструктажі, спеціальне навчання та перевірку знань з питань пожежної безпеки на підприємствах, в установах та організаціях України;
7. ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень»;
8. ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин»;
9. ГОСТ 12.1.044-89 «ССБТ. Вогнестійкість. Номенклатура показників і методи їх визначення»;
10. ГОСТ 12.1.030-81 «Електробезпека. Захисне заземлення, занулення».
11. ГОСТ 12.1.006-84 «ССБТ. Електромагнітні поля радіочастот»;
12. ГОСТ 13109-97«Електрична енергія. Сумісність технічних засобів. Норми якості електричної енергії в системах електропостачання загального призначення»;
13. ДБН В.2.5-28:2015 «Державні Будівельні Норми України. Природне і штучне освітлення»;
14. НАПБ Б.03.002-2007«Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою».

15. Теоретичні основи теплотехніки. Теплотехнічний експеримент .
Довідник / За заг. Ред. чл.-кор. АН СРСР В. А. Григор'єва, В. М. Зоріна. - 2-е вид.- М . Вища школа, 1988.-560 с.;
16. Альтштуль А. Д. Гідравлічний опір. М . Стройиздат, 1973.;
17. З. Ідельчик І. Е. Гідравлічні опору. М . Госенергоіздат, 1954.;
18. Ідельчик І. Е. Довідник з гідравлічних опорам. М. Машино-будування, 1975.
19. Дейч М. Є. Технічна газодинаміка. М . Енергія, 1974.;
20. А. Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд. Язык программирования С#. Классика Computers Science. 4-е изд., 2004.

ДОДАТКИ
Додаток А
Програмний код

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApplication14
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void perform_uc_Click(object sender, EventArgs e)
        {
            double result = 0;

            int n = Convert.ToInt32(input_n.Text);
            int m = Convert.ToInt32(input_m.Text);

            double S0 = Convert.ToDouble(input_S0.Text);
            S0 = Math.Pow(S0, 2.0); // попередньо зводимо в ступінь

            double part_1 = 0;
            double part_2 = 0;

            double[] in_Si = input_Si.Text.Split(new[] { ';' },
StringSplitOptions.RemoveEmptyEntries).Select(z => double.Parse(z.Trim())).ToArray();
            double[] in_li = input_li.Text.Split(new[] { ';' },
StringSplitOptions.RemoveEmptyEntries).Select(z => double.Parse(z.Trim())).ToArray();
            double[] in_di = input_di.Text.Split(new[] { ';' },
StringSplitOptions.RemoveEmptyEntries).Select(z => double.Parse(z.Trim())).ToArray();
            double[] in_LAi = input_LAi.Text.Split(new[] { ';' },
StringSplitOptions.RemoveEmptyEntries).Select(z => double.Parse(z.Trim())).ToArray();

```



```

        double[]    in_Sk    =    input_Sk.Text.Split(new[]    {    ';'    },
StringSplitOptions.RemoveEmptyEntries).Select(z => double.Parse(z.Trim())).ToArray();
        double[]    in_SIk    =    input_SIk.Text.Split(new[]    {    ';'    },
StringSplitOptions.RemoveEmptyEntries).Select(z => double.Parse(z.Trim())).ToArray();

        for (int i = 0; i < m; ++i)
        {
            part_1    +=    in_LAi[i]    *    (in_li[i]    /    in_di[i])    *    (S0    /
Math.Pow(in_Si[i], 2.0));
        }

        for (int k = 0; k < n; ++k)
        {
            part_2 += in_SIk[k] * (S0 / Math.Pow(in_Sk[k], 2.0));
        }

        result = Math.Pow(part_1 + part_2, -0.5);

        output_result.Text = Convert.ToString(result);
    }

}
namespace WindowsFormsApplication1
{
    partial class Form1
    {
        /// <summary>
        /// Требуется переменная конструктора.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        /// <param name="disposing">истинно, если управляемый ресурс должен быть
удален; иначе ложно.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))

```

```

    {
        components.Dispose();
    }
    base.Dispose(disposing);
}

#region Код, автоматически созданный конструктором форм Windows

/// <summary>
/// Обязательный метод для поддержки конструктора - не изменяйте
/// содержимое данного метода при помощи редактора кода.
/// </summary>
private void InitializeComponent()
{
    this.perform_uc = new System.Windows.Forms.Button();
    this.input_n = new System.Windows.Forms.TextBox();
    this.input_m = new System.Windows.Forms.TextBox();
    this.label1 = new System.Windows.Forms.Label();
    this.label2 = new System.Windows.Forms.Label();
    this.label3 = new System.Windows.Forms.Label();
    this.label4 = new System.Windows.Forms.Label();
    this.label5 = new System.Windows.Forms.Label();
    this.label6 = new System.Windows.Forms.Label();
    this.label7 = new System.Windows.Forms.Label();
    this.label8 = new System.Windows.Forms.Label();
    this.label9 = new System.Windows.Forms.Label();
    this.input_S0 = new System.Windows.Forms.TextBox();
    this.input_Si = new System.Windows.Forms.TextBox();
    this.input_di = new System.Windows.Forms.TextBox();
    this.input_li = new System.Windows.Forms.TextBox();
    this.input_LAi = new System.Windows.Forms.TextBox();
    this.input_SIk = new System.Windows.Forms.TextBox();
    this.input_Sk = new System.Windows.Forms.TextBox();
    this.label14 = new System.Windows.Forms.Label();
    this.output_result = new System.Windows.Forms.TextBox();
    this.label10 = new System.Windows.Forms.Label();
    this.label11 = new System.Windows.Forms.Label();
    this.label12 = new System.Windows.Forms.Label();
    this.label13 = new System.Windows.Forms.Label();
    this.label15 = new System.Windows.Forms.Label();
    this.SuspendLayout();
}

```

```
//  
// perform_uc  
//  
this.perform_uc.Location = new System.Drawing.Point(41, 576);  
this.perform_uc.Name = "perform_uc";  
this.perform_uc.Size = new System.Drawing.Size(277, 66);  
this.perform_uc.TabIndex = 0;  
this.perform_uc.Text = "Розрахувати коефіцієнт витрат";  
this.perform_uc.UseVisualStyleBackColor = true;  
this.perform_uc.Click += new System.EventHandler(this.perform_uc_Click);  
//  
// input_n  
//  
this.input_n.Location = new System.Drawing.Point(854, 33);  
this.input_n.Name = "input_n";  
this.input_n.Size = new System.Drawing.Size(46, 20);  
this.input_n.TabIndex = 1;  
this.input_n.Text = "1";  
//  
// input_m  
//  
this.input_m.Location = new System.Drawing.Point(229, 33);  
this.input_m.Name = "input_m";  
this.input_m.Size = new System.Drawing.Size(46, 20);  
this.input_m.TabIndex = 2;  
this.input_m.Text = "1";  
//  
// label1  
//  
this.label1.AutoSize = true;  
this.label1.Location = new System.Drawing.Point(726, 36);  
this.label1.Name = "label1";  
this.label1.Size = new System.Drawing.Size(122, 13);  
this.label1.TabIndex = 3;  
this.label1.Text = "Число місцевих опорів";  
//  
// label2  
//  
this.label2.AutoSize = true;  
this.label2.Location = new System.Drawing.Point(38, 36);  
this.label2.Name = "label2";
```

```
this.label2.Size = new System.Drawing.Size(185, 13);
this.label2.TabIndex = 4;
this.label2.Text = "Число ділянок постійного діаметра";
//
// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(38, 99);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(199, 13);
this.label3.TabIndex = 5;
this.label3.Text = "Площа поперечного перерізу ділянки ";
//
// label4
//
this.label4.AutoSize = true;
this.label4.Location = new System.Drawing.Point(701, 342);
this.label4.Name = "label4";
this.label4.Size = new System.Drawing.Size(131, 13);
this.label4.TabIndex = 6;
this.label4.Text = "Площа місцевого опору";
//
// label5
//
this.label5.AutoSize = true;
this.label5.Location = new System.Drawing.Point(302, 99);
this.label5.Name = "label5";
this.label5.Size = new System.Drawing.Size(49, 13);
this.label5.TabIndex = 7;
this.label5.Text = "Діаметр";
//
// label6
//
this.label6.AutoSize = true;
this.label6.Location = new System.Drawing.Point(302, 342);
this.label6.Name = "label6";
this.label6.Size = new System.Drawing.Size(160, 13);
this.label6.TabIndex = 8;
this.label6.Text = "Гідрравлічний коефіцієнт тертя ";
//
// label7
```

```
//  
  
this.label7.AutoSize = true;  
this.label7.Location = new System.Drawing.Point(38, 342  
this.label7.Name = "label7";  
this.label7.Size = new System.Drawing.Size(54, 13);  
this.label7.TabIndex = 9;  
this.label7.Text = "Довжина";  
//  
// label8  
//  
this.label8.AutoSize = true;  
this.label8.Location = new System.Drawing.Point(701, 99);  
this.label8.Name = "label8";  
this.label8.Size = new System.Drawing.Size(150, 13);  
this.label8.TabIndex = 10;  
this.label8.Text = "Коефіцієнт місцевого опору ";  
//  
// label9  
//  
this.label9.AutoSize = true;  
this.label9.Location = new System.Drawing.Point(338, 36);  
this.label9.Name = "label9";  
this.label9.Size = new System.Drawing.Size(260, 13);  
this.label9.TabIndex = 11;  
this.label9.Text = "Площа поперечного перерізу труби в місці викиду";  
//  
// input_S0  
//  
this.input_S0.Location = new System.Drawing.Point(604, 32);  
this.input_S0.Name = "input_S0";  
this.input_S0.Size = new System.Drawing.Size(46, 20);  
this.input_S0.TabIndex = 12;  
this.input_S0.Text = "1";  
//  
// input_Si  
//  
this.input_Si.Location = new System.Drawing.Point(41, 115);  
this.input_Si.Multiline = true;  
this.input_Si.Name = "input_Si";  
this.input_Si.Size = new System.Drawing.Size(196, 157);
```

```
this.input_Si.TabIndex = 13;
//
// input_di
//
this.input_di.Location = new System.Drawing.Point(305, 115);
this.input_di.Multiline = true;
this.input_di.Name = "input_di";
this.input_di.Size = new System.Drawing.Size(196, 157);
this.input_di.TabIndex = 14;
//
// input_li
//
this.input_li.Location = new System.Drawing.Point(41, 358);
this.input_li.Multiline = true;
this.input_li.Name = "input_li";
this.input_li.Size = new System.Drawing.Size(196, 157);
this.input_li.TabIndex = 15;
//
// input_LAi
//
this.input_LAi.Location = new System.Drawing.Point(304, 358);
this.input_LAi.Multiline = true;
this.input_LAi.Name = "input_LAi";
this.input_LAi.Size = new System.Drawing.Size(196, 157);
this.input_LAi.TabIndex = 16;
//
// input_SIk
//
this.input_SIk.Location = new System.Drawing.Point(704, 115);
this.input_SIk.Multiline = true;
this.input_SIk.Name = "input_SIk";
this.input_SIk.Size = new System.Drawing.Size(196, 157);
this.input_SIk.TabIndex = 17;
//
// input_Sk
//
this.input_Sk.Location = new System.Drawing.Point(704, 358);
this.input_Sk.Multiline = true;
this.input_Sk.Name = "input_Sk";
this.input_Sk.Size = new System.Drawing.Size(196, 157);
this.input_Sk.TabIndex = 18;
```

```

//
// label14
//
this.label14.AutoSize = true;
this.label14.Location = new System.Drawing.Point(38, 275);
this.label14.Name = "label14";
this.label14.Size = new System.Drawing.Size(203, 13);
this.label14.TabIndex = 23;
this.label14.Text = "введення даних через крапку з комою";
//
// output_result
//
this.output_result.Font = new System.Drawing.Font("Microsoft Sans Serif",
27.75F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point,
((byte)204));
this.output_result.Location = new System.Drawing.Point(402, 576);
this.output_result.Multiline = true;
this.output_result.Name = "output_result";
this.output_result.ReadOnly = true;
this.output_result.Size = new System.Drawing.Size(498, 66);
this.output_result.TabIndex = 25;
this.output_result.Text = "0";
this.output_result.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center;
//
// label10
//
this.label10.AutoSize = true;
this.label10.Location = new System.Drawing.Point(302, 275);
this.label10.Name = "label10";
this.label10.Size = new System.Drawing.Size(203, 13);
this.label10.TabIndex = 26;
this.label10.Text = "введення даних через крапку з комою";
//
// label11
//
this.label11.AutoSize = true;
this.label11.Location = new System.Drawing.Point(701, 275);
this.label11.Name = "label11";
this.label11.Size = new System.Drawing.Size(203, 13);
this.label11.TabIndex = 27;

```

```
this.label11.Text = "введення даних через крапку з комою";  
//  
// label12  
//  
this.label12.AutoSize = true;  
this.label12.Location = new System.Drawing.Point(701, 518);  
this.label12.Name = "label12";  
this.label12.Size = new System.Drawing.Size(203, 13);  
this.label12.TabIndex = 28;  
this.label12.Text = "введення даних через крапку з комою";  
//  
// label13  
//  
this.label13.AutoSize = true;  
this.label13.Location = new System.Drawing.Point(302, 518);  
this.label13.Name = "label13";  
this.label13.Size = new System.Drawing.Size(203, 13);  
this.label13.TabIndex = 29;  
this.label13.Text = "введення даних через крапку з комою";  
//  
// label15  
//  
this.label15.AutoSize = true;  
this.label15.Location = new System.Drawing.Point(38, 518);  
this.label15.Name = "label15";  
this.label15.Size = new System.Drawing.Size(203, 13);  
this.label15.TabIndex = 30;  
this.label15.Text = "введення даних через крапку з комою";  
//  
// Form1  
//  
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);  
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;  
this.ClientSize = new System.Drawing.Size(947, 685);  
this.Controls.Add(this.label15);  
this.Controls.Add(this.label13);  
this.Controls.Add(this.label12);  
this.Controls.Add(this.label11);  
this.Controls.Add(this.label10);  
this.Controls.Add(this.output_result);  
this.Controls.Add(this.label14);
```



```

        this.Controls.Add(this.input_Sk);
        this.Controls.Add(this.input_SIk);
        this.Controls.Add(this.input_LAi);
        this.Controls.Add(this.input_li);
        this.Controls.Add(this.input_di);
        this.Controls.Add(this.input_Si);
        this.Controls.Add(this.input_S0);
        this.Controls.Add(this.label9);
        this.Controls.Add(this.label8);
        this.Controls.Add(this.label7);
        this.Controls.Add(this.label6);
        this.Controls.Add(this.label5);
        this.Controls.Add(this.label4);
        this.Controls.Add(this.label3);
        this.Controls.Add(this.label2);
        this.Controls.Add(this.label1);
        this.Controls.Add(this.input_m);
        this.Controls.Add(this.input_n);
        this.Controls.Add(this.perform_uc);
        this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.FixedToolWindow;
        this.Name = "Form1";
        this.Text = "Розрахувати коефіцієнт витрат";
        this.ResumeLayout(false);
        this.PerformLayout();

    }

#endregion

private System.Windows.Forms.Button perform_uc;
private System.Windows.Forms.TextBox input_n;
private System.Windows.Forms.TextBox input_m;
private System.Windows.Forms.Label label1;
private System.Windows.Forms.Label label2;
private System.Windows.Forms.Label label3;
private System.Windows.Forms.Label label4;
private System.Windows.Forms.Label label5;
private System.Windows.Forms.Label label6;
private System.Windows.Forms.Label label7;
private System.Windows.Forms.Label label8;

```

```
private System.Windows.Forms.Label label9;
private System.Windows.Forms.TextBox input_S0;
private System.Windows.Forms.TextBox input_Si;
private System.Windows.Forms.TextBox input_di;
private System.Windows.Forms.TextBox input_li;
private System.Windows.Forms.TextBox input_LAi;
private System.Windows.Forms.TextBox input_SIk;
private System.Windows.Forms.TextBox input_Sk;
private System.Windows.Forms.Label label14;
private System.Windows.Forms.TextBox output_result;
private System.Windows.Forms.Label label10;
private System.Windows.Forms.Label label11;
private System.Windows.Forms.Label label12;
private System.Windows.Forms.Label label13;
private System.Windows.Forms.Label label15;
    }
}
```

Додаток Б
Презентація

Дипломний проект бакалавра на тему

ПРОГРАМНИЙ МОДУЛЬ ПРОГНОЗУВАННЯ ВИТОКУ НЕБЕЗПЕЧНИХ ГАЗІВ З ОБЛАДНАННЯ

Студента гр. КН-14д

Кукочки О. О.

Керівник проекту ст. викл.

Лифар О.К.

1

Проблеми

Технології, що представляють потенційну небезпеку, присутні практично у всіх областях діяльності людини. Незважаючи на те, що велика увага приділяється рівню безпеки та перевіркам обладнання, аварії стаються щороку, в результаті чого промисловість зазнає значних збитків. Тому потрібно замислитись про безпеку робітників на підприємствах, а також за наше навколишнє середовище.

Причини виникнення аварій різні:

- природні і людські фактори;
- знос устаткування.

Не малу роль грає недостатньо якісне проектування виробничої дільниці.

2

Існуючі рішення

TOXI+ Risk 5

Цей програмний модуль дозволяє візуалізувати результати розрахунків на планах місцевості, оцінити кількість людей, що потрапили в небезпечні райони, а також частотні поля, що перевищують заданий рівень надлишкового тиску та імпульсу з декількох джерел небезпеки для різних сценаріїв аварії. Результати моделювання надаються у вигляді графічних діаграм.

3

Калькулятор "Гідравлічні розрахунки у трубопроводах"

Ця програма може обчислити об'єм газу з систем трубопроводів при наступних вхідних параметрах:

- за значенням внутрішнього умовного діаметру трубопроводу;
- за довжиною трубопроводу;
- за падінням тиску;
- за коефіцієнтом опору.

Вона розраховує виток не через аварійний отвір, а об'єм газу згідно заданих параметрів діаметра труби, а значить об'єм викиду під час аварії не обчислюється, тому при зазначених коефіцієнтах опору, пропускній здатності трубопроводу не враховується коефіцієнт місцевого опору.

4

Актуальність

Метою даного дипломного проекту є розробка програмного модулю, який прогнозує виток небезпечних газів з обладнання.

Аналіз результатів роботи програмного модуля дозволить зберегти навколишнє середовище, заощадити сили і час для ручних розрахунків, зробить цю програму як матеріально вигідною, так і доцільною в роботі для розрахунку ризиків на небезпечних підприємствах.

5

Функції програмного модуля

Програмний модуль повинен виконувати наступні розрахунки:

- відношенням маси випущеного газу до часу;
- відношення об'єму газу до часу.
- розподіл концентрації отруйної речовини в залежності від атмосферних параметрів;

6

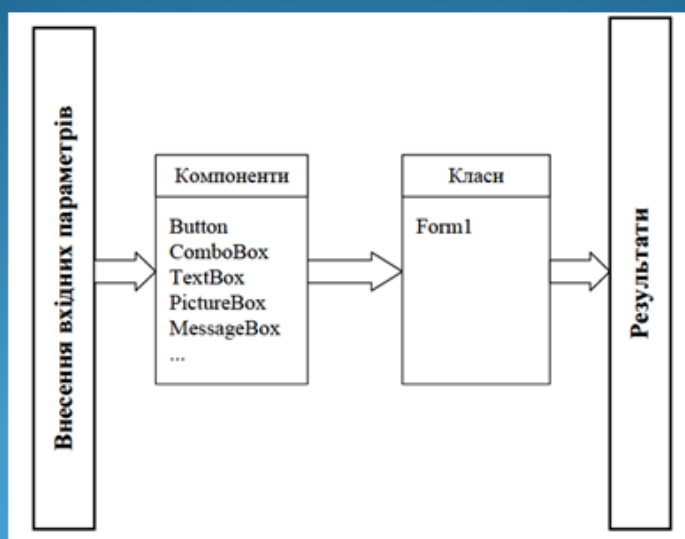
Переваги С#

З переваг можна зазначити:

- С# реалізована можливість включення в програму класів та об'єктів з сторонніх програм;
- легка в освоєнні завдяки великій кількості літератури, яка доступна в інтернеті;
- дана мова має велику кількість бібліотек DLL.

7

Компоненти та класи



8

Основне вікно модуля «Витоку газу»

файл Расчет Сервис Помощь

Параметры | Массовый расход из емкости | Масса выброса | Давление в емкости | Массовый расход из источника

Тип истечения

Через отверстие

Через систему трубопроводов

Параметры емкости

Начальное давление, P_s : 1010000 (Па)

Молекулярная масса газа, M : 0.021 (кг/моль)

Температура в емкости, T : 293.00 (К)

Отношение теплоемкостей, k : 1.40

Объем сосуда, $V_{ср}$: 100 (м³)

Коэффициент расхода отверстия истечения из источника в емкость, μ_1 : 0.80

Коэффициент расхода отверстия истечения в атмосферу, μ_2 : 0.80

Время перекрытия источника, T_{stop} : 300 (сек)

Площадь отверстия прихода, F_1 : 0.02 (м²)

Площадь отверстия истечения, F_2 : 0.05 (м²)

Кинематическая вязкость газа, ν_k : 0.0000180 (м²/с)

Атмосферное давление, P_a : 101000 (Па)

9

Налаштування за замовчуванням модуля «Витоку газу»

Настройки

Универсальная газовая постоянная, R : 8.31 (Дж/моль*град)

Время окончания счета, T_{end} : 400 (сек)

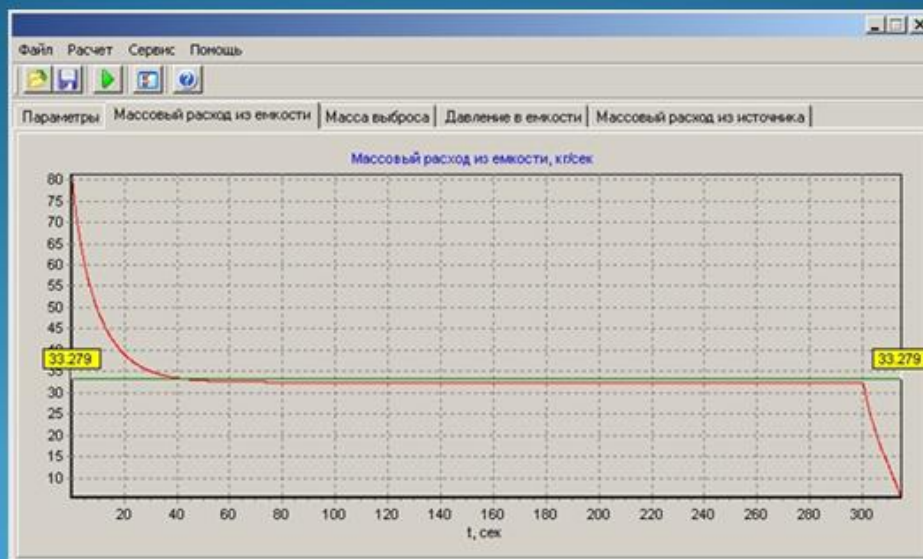
Шаг по времени счета, dt : 0.10 (сек)

Приближение окончания счета, eps : 0.1000

Ok Cancel

10

Закладка «Масова витрата із ємкості»



11

Закладка додаткового введення даних при витокі через систему трубопроводів і місцевих опорів

Файл Расчет Сервис Помощь

Параметры Дополнительно Массовый расход из емкости Масса выброса Давление в

Параметры трубопровода

№пп	Длина, м	Диаметр, м	Абсолютная шероховатость, мм
1	20	0.1	0.1
2	52	0.2	0.1

Строка

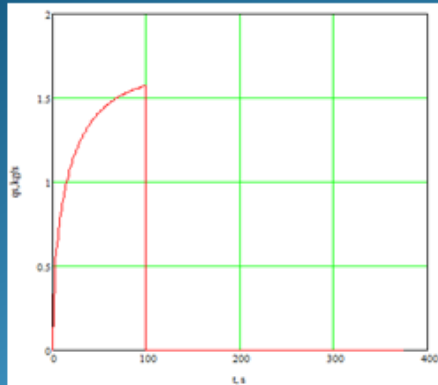
Параметры местных сопротивлений

№пп	Козф. местного сопротивления
1	0.2
2	0.05

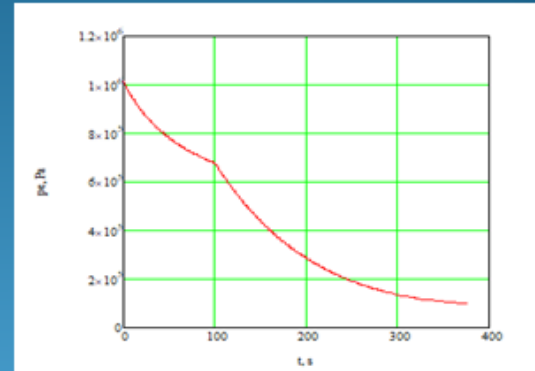
Строка

12

Результат роботи



Моделювання витoku через отвір в Mathcad



Моделювання витoku газу через систему трубопроводів і місцевих опорів в Mathcad

13

ВИСНОВКИ

В дипломному проекті була розроблена програмна реалізація математичної моделі на мові програмування C# у вигляді віконного додатку Windows.

Представлено модуль програмного забезпечення, який обчислює показники витoku газу з обладнання та формує графічне представлення цього процесу у вигляді діаграм, в залежності від значень вхідних параметрів.

Програма була запущена, і отримані результати були проаналізовані. В результаті, програма працює правильно, в процесі обчислень не виникає помилок, не коректних результатів чи збоїв.

Аналіз результатів роботи програмного модулю дасть змогу зберегти навколишнє середовище, заощадити сили і час на ручні розрахунки, що робить цю програму доцільною та актуальною.

14