Міністерство освіти та науки України
Харківський національний університет радіоелектроніки

Кафедра програмної інженерії

Звіт з практичного заняття №1
з дисципліни
«Аналіз та рефакторінг коду програмного забезпечення»

Виконав:
гр. ПЗПІ-22-7
Ігнатьєв О.Г.

Перевірив: ст.
Сокорчук І.П.

Харків 2024

1 МЕТА

Навчитися рефакторингу програмного коду, закріпити основні правила оформлення коду.

2 ЗАВДАННЯ

1. Обрати мову програмування для прикладів коду.
2. Створити презентацію на тему «Правила оформлення програмного коду».

3 ХІД РОБОТИ

Обрана мова програмування: C++

У презентації наведено основні рекомендації щодо написання та оформлення програмного коду.

1. Використовуйте зрозумілі імена змінних

Опис:
Змінні повинні мати змістовні назви та відповідати контексту. Використовуйте стиль camelCase.

Поганий приклад:

```
int chislo1 = 10; // Ім'я російською
```

Гарний приклад:

```
int numberOfItems = 10; // Ім'я англійською з використанням camelCase
```

2. Форматуйте відступи правильно

Опис:
Для покращення читабельності використовуйте чотири пробіли для відступів.

Поганий приклад:

```
if(x>10){cout<<"Error";}
```

Гарний приклад:

```
if (x > 10) {
    cout << "Error";
}
```

3. Скорочуйте вкладеність

Опис:
Мінімізуйте вкладення логіки для спрощення читання коду.

Поганий приклад:

```
if (a > 0) {
    if (b > 0) {
        if (c > 0) {
            cout << "All variables are positive";
```

```
        }
    }
}
```

Гарний приклад:

```
if (a > 0 && b > 0 && c > 0) {
    cout << "All variables are positive";
}
```

4. Використовуйте коментарі для пояснення

Опис:
Коментарі повинні бути короткими, зрозумілими й відповідати коду.
Поганий приклад:

```
// Функція щось виконує
void func() {}
```

Гарний приклад:

```
// Обчислює квадрат числа
int square(int x) {
    return x * x;
}
```

5. Обмежуйте довжину рядків

Опис:
Рядки довжиною більше 80 символів слід переносити.
Гарний приклад:

```
string longMessage = "This is a long message that is split into "
                     "two lines for better readability.";
```

6. Дотримуйтеся стандартів іменування класів

Опис:
Класи оформлюйте у стилі PascalCase, методи — у camelCase.
Гарний приклад:

```
class UserProfile {
public:
    void getUserData();
};
```

7. Форматуйте масиви чітко

Опис:
При форматуванні масивів використовуйте структуровані відступи.
Гарний приклад:

```
int numbers[] = {
    1, 2, 3,
```

```
    4, 5, 6
};
```

ВИСНОВКИ

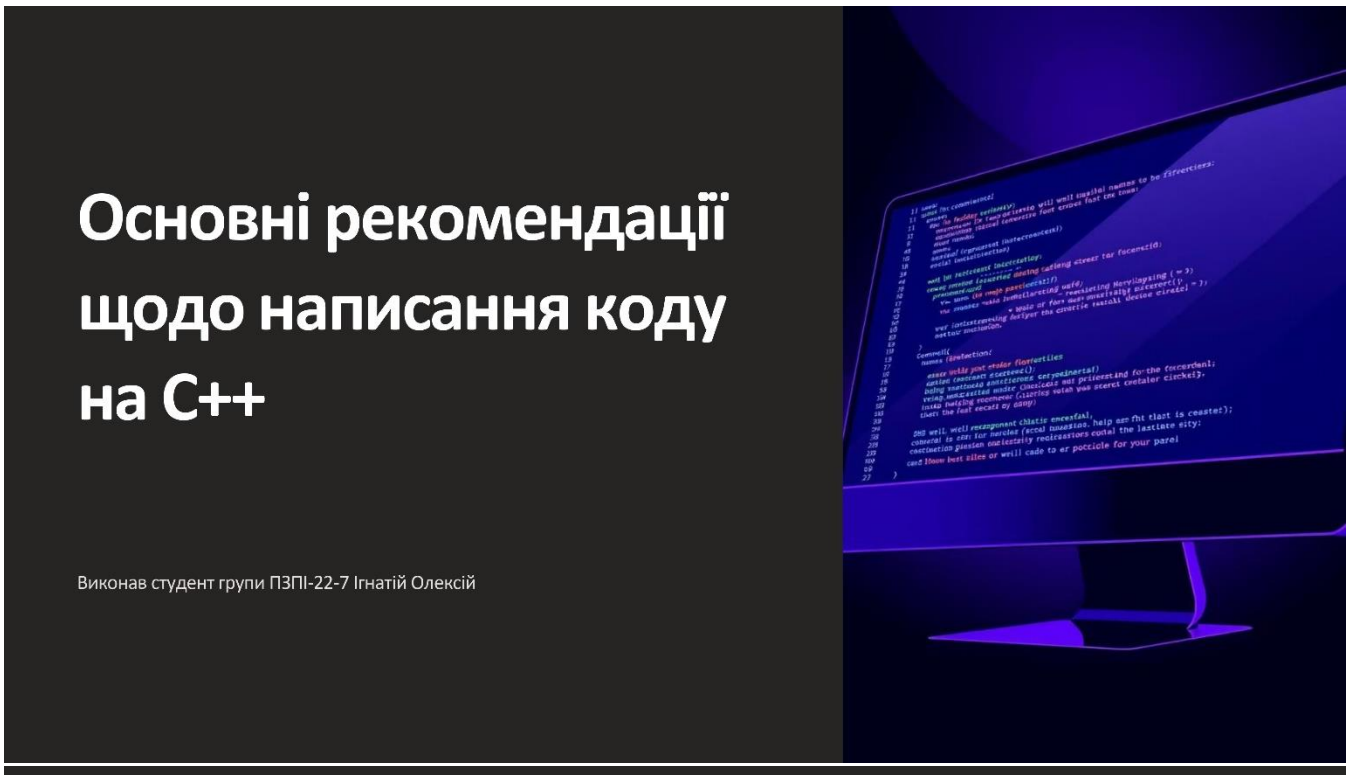
Отже, у ході роботи було набуто навичок рефакторингу програмного коду, детально розглянуто основні правила оформлення коду.

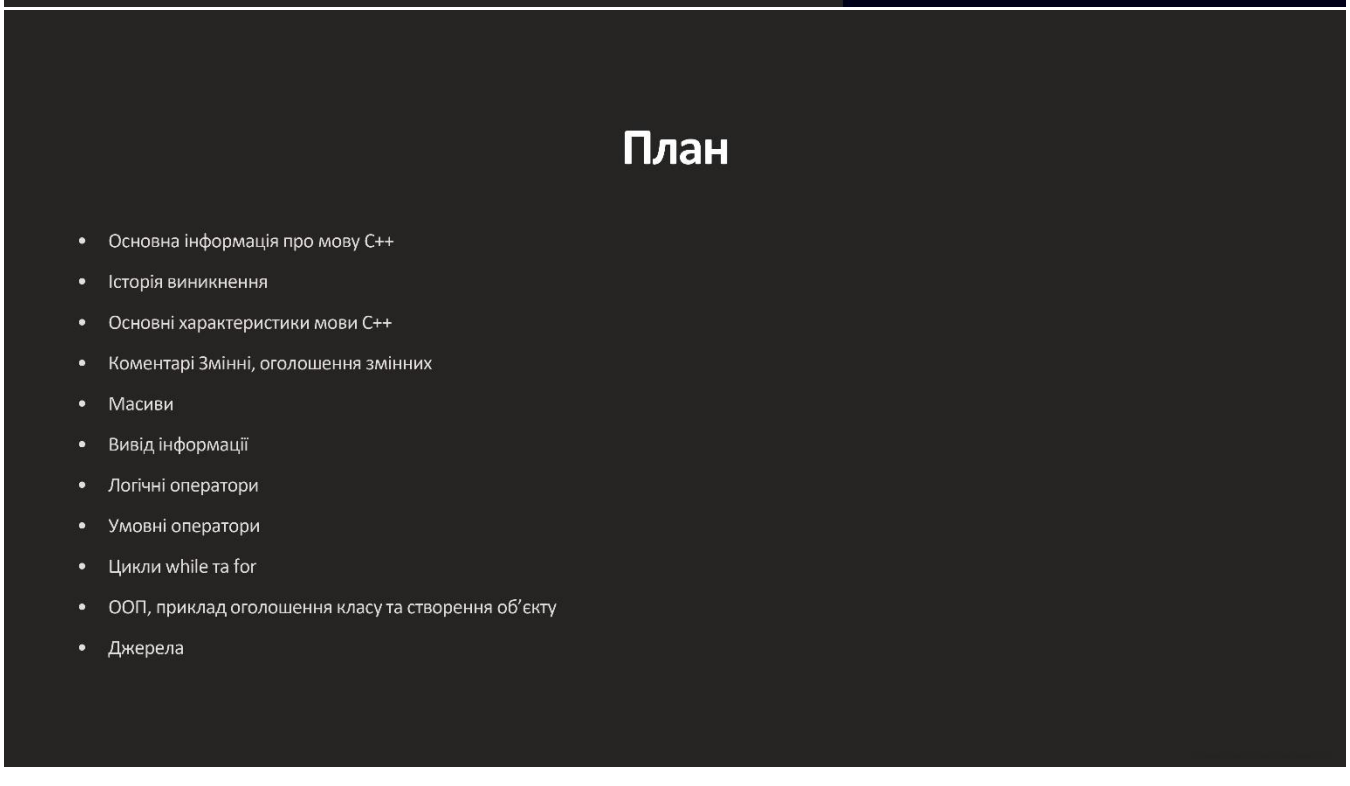
Посилання на youtube:

https://www.youtube.com/watch?v=Q8rvLIKHhxs&t=4s

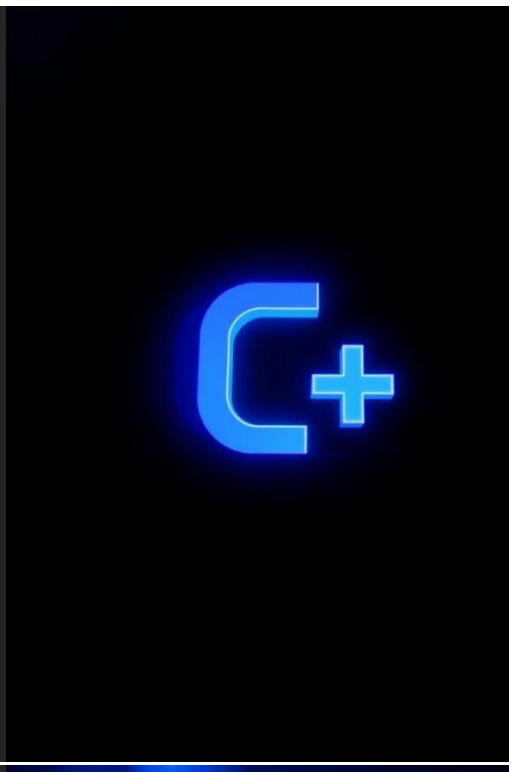
ДОДАТОК А

Презентація на тему «Правила оформлення програмного коду».





Основна інформація про мову C++

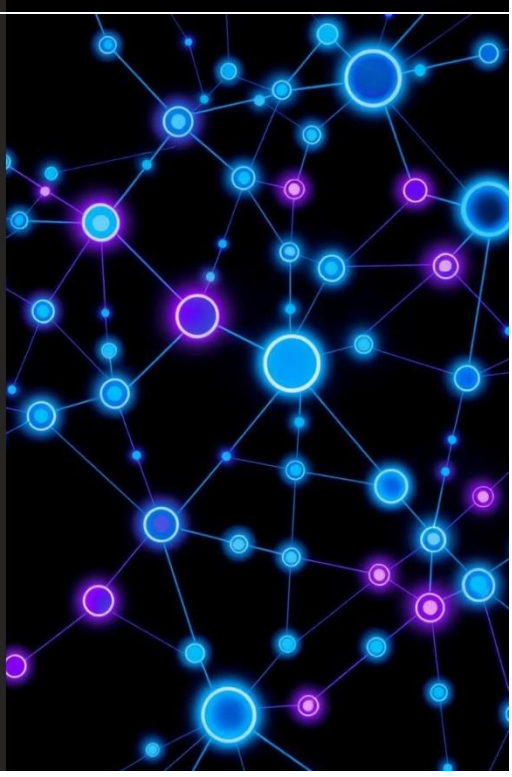


C++ - потужна мова програмування, що поєднує в собі можливості низького та високого рівня. Вона широко використовується для розробки різноманітного програмного забезпечення.

Історія виникнення

C++ був розроблений у 1979 році Б'ярном Страуструпом в Bell Labs як вдосконалення мови C. Спочатку він називався "C with Classes", оскільки додавав об’єктно-орієнтовані можливості до C. У 1983 році мову перейменували на C++.

Основні характеристики мови C++



1. Підтримка об’єктно-орієнтованого програмування (ООП)
2. Наслідування С: можливість низькорівневого програмування.
3. Шаблони (templates) для генерації типів даних.
4. Робота з пам’яттю через вказівники.
5. Поліморфізм, інкапсуляція, наслідування як ключові ООП концепції.

```cpp
int main() {

    // Це однорядковий коментар
    /*
    Це багаторядковий
    коментар
    */

}
```

Коментарі

Коментарі в C++ використовуються для пояснення коду і бувають однорядковими та багаторядковими.

Однорядкові

Починаються з двох символів косої риски (//).

Багаторядкові

Розміщуються між /* та */.

```cpp
int main() {

    int age = 25; // Ціла змінна
    double height = 1.75; // Змінна з плаваючою крапкою
    char initial = 'A'; // Символьна змінна

}
```

Змінні, оголошення змінних

Змінні - це іменовані області пам'яті для зберігання даних.

Оголошення змінних включає вказівку типу даних та імені змінної.

Оголошення змінної:

```
тип_даних назва_змінної;
```

Масиви

Масиви в C++ дозволяють зберігати множину значень одного типу в одному об'єкті. Це особливо корисно, коли потрібно обробляти колекцію даних. Масиви мають фіксовану розмірність, яку необхідно визначити при їх оголошенні.

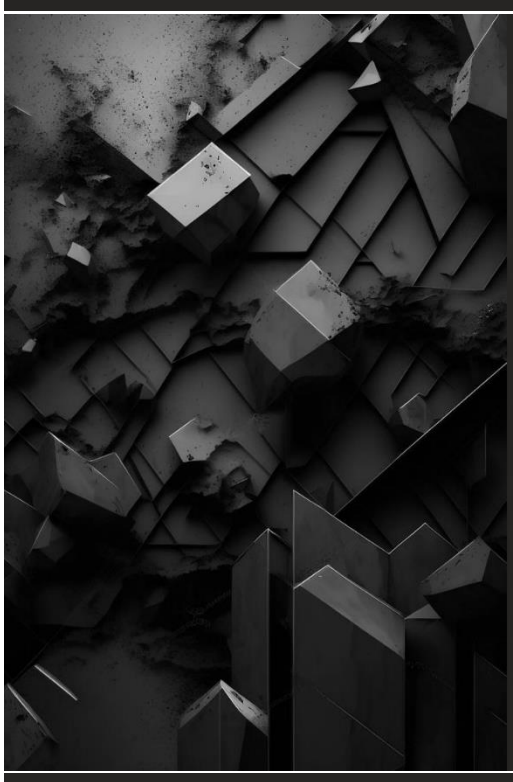
Оголошення масиву:

```
int numbers[5] = {1, 2, 3, 4, 5}; // Масив з 5 елементів
```

Вивід інформації

Для виведення даних на екран використовується оператор `std::cout`

```
#include <iostream>
int main() {
    std::cout << "Hello, C++!" << std::endl;
    return 0;
}
```



Логічні оператори

Оператор	Опис
&&	Логічне "І"
||	Логічне "АБО"
!	Логічне "НІ"

Умовні оператори

Умовні оператори використовуються для прийняття рішень на основі певних умов.

```
int age = 18;
if (age >= 18) {
    std::cout << "Дорослий" << std::endl;
} else {
    std::cout << "Дитина" << std::endl;
}
```

Цикли while та for

Цикли дозволяють повторювати блок коду кілька разів. Цикл `for` зручний, коли відомо кількість ітерацій, тоді як цикл `while` працює, поки умова є істинною.

```
// Цикл for
for (int i = 0; i < 5; i++) {
    std::cout << i << " ";
}

// Цикл while
int j = 0;
while (j < 5) {
    std::cout << j << " ";
    j++;
}
```

ООП, приклад оголошення класу та створення об’єкту

Об'єктно-орієнтоване програмування (ООП) є ключовим аспектом C++. ООП дозволяє створювати класи, які є шаблонами для об'єктів. Класи можуть містити змінні (властивості) і методи (функції).

```
class Car {
public:
    std::string brand;
    int year;

    void drive() {
        std::cout << "The car is driving!" << std::endl;
    }
};

int main() {
    Car myCar; // Створюємо об'єкт myCar класу Car
    myCar.brand = "Toyota";
    myCar.year = 2020;
    myCar.drive(); // Викликаємо метод об'єкту
    return 0;
}
```

Джерела

1. Страуструп Б. "The C++ Programming Language" - книга, яка є основним джерелом для вивчення C++.
2. Сайт cplusplus.com - ресурс з документацією та прикладами коду.
3. Сайт GeeksforGeeks - велика кількість статей та туторіалів з C++.

Дякую за увагу