

Додаткове завдання: Реалізувати Discord-бота для компілятора

Виконав: ст. гр. КІ-309 Мерунович З.

Дисципліна: Системне програмування.

Використані інструменти

- **Мова програмування:** C++.
- **Бібліотеки:** D++ (DPP) для роботи з Discord API, Windows API для керування процесами.
- **Середовище:** Visual Studio 2022.
- **Зовнішні компоненти:** Власний транслятор (cw_sp2_2025_2026.exe) та компілятор GCC (MinGW).

Опис реалізації

Створено багатопотоковий додаток-бот, який виконує роль "обгортки" для консольного компілятора. Алгоритм роботи:

1. Обробка команд: Бот перехоплює повідомлення !run та виділяє код програми з тіла повідомлення.
2. Файлові операції: Отриманий код зберігається у тимчасовий текстовий файл.
3. Запуск транслятора: Бот створює дочірній процес із транслятором, передаючи йому збережений файл.
4. Ланцюгова компіляція: Після успішної генерації С-коду транслятором, бот автоматично викликає gcc для створення виконуваного файла (.exe).
5. Керування вводом/виводом (I/O Redirection): Запуск фінальної програми відбувається з перенаправленням стандартних потоків вводу/виводу через анонімні канали (Pipes). Це дозволяє боту читати результат роботи програми та передавати їй вхідні дані від користувача з чату в реальному часі.
6. Фільтрація: Реалізовано очищення виводу від службових логів транслятора.

Інструкція з налаштування

Для запуску бота на іншому комп'ютері необхідно:

1. Перейдіть на [Discord Developer Portal](#), натисніть кнопку "New Application", введіть назву вашого бота (наприклад, CompilerBot) та натисніть "Create".
2. У лівому меню перейдіть у вкладку "Bot". Знайдіть секцію "Build-A-Bot" (якщо бот не створений автоматично, натисніть "Add Bot"). Тут можна встановити аватарку та ім'я, яке бачитимуть користувачі.
3. На вкладці "Bot" прокрутіть вниз до секції "Privileged Gateway Intents". Необхідно увімкнути перемикач "Message Content Intent". Без цього бот не зможе читати команди (наприклад, !run) та код програми у повідомленнях.

Також рекомендується увімкнути "Server Members Intent" та "Presence Intent". Збережіть зміни ("Save Changes").

4. Отримання токена: У секції "Build-A-Bot" натисніть кнопку "Reset Token". Скопіюйте згенерований рядок символів. Це і є ваш ключ доступу.
5. Збереження токена: Створіть файл token.txt у папці з виконуваним файлом бота (.exe) і вставте туди скопійований токен (без зайвих пробілів чи нових рядків).
6. Запрошення на сервер:
 - Перейдіть у вкладку "OAuth2" -> "URL Generator".
 - У полі "Scopes" поставте галочку на "bot".
 - У полі "Bot Permissions" виберіть "Administrator" (або "View Channels", "Send Messages", "Read Message History").
 - Скопіюйте згенероване посилання внизу сторінки ("Generated URL"), відкрийте його в браузері та додайте бота на свій сервер.
7. Встановіть MinGW (GCC) та додати шлях до нього у змінні середовища (PATH).

Bot Permissions:

Bot Permissions
Need some help with bit math? Use the tool below to calculate the permissions integer for your bot based on the features it needs.

General Permissions <input type="checkbox"/> Administrator <input type="checkbox"/> View Audit Log <input type="checkbox"/> Manage Server <input type="checkbox"/> Manage Roles <input type="checkbox"/> Manage Channels <input type="checkbox"/> Kick Members <input type="checkbox"/> Ban Members <input type="checkbox"/> Create Instant Invite <input type="checkbox"/> Change Nickname <input type="checkbox"/> Manage Nicknames <input type="checkbox"/> Manage Expressions <input type="checkbox"/> Create Expressions <input type="checkbox"/> Manage Webhooks <input checked="" type="checkbox"/> View Channels <input type="checkbox"/> Manage Events <input type="checkbox"/> Create Events <input type="checkbox"/> Moderate Members <input type="checkbox"/> View Server Insights <input type="checkbox"/> View Server Subscription Insights	Text Permissions <input checked="" type="checkbox"/> Send Messages <input type="checkbox"/> Create Public Threads <input type="checkbox"/> Create Private Threads <input type="checkbox"/> Send Messages in Threads <input type="checkbox"/> Send TTS Messages <input type="checkbox"/> Manage Messages <input type="checkbox"/> Pin Messages <input type="checkbox"/> Manage Threads <input type="checkbox"/> Embed Links <input type="checkbox"/> Attach Files <input checked="" type="checkbox"/> Read Message History <input type="checkbox"/> Mention Everyone <input type="checkbox"/> Use External Emojis <input type="checkbox"/> Use External Stickers <input type="checkbox"/> Add Reactions <input type="checkbox"/> Use Slash Commands <input type="checkbox"/> Use Embedded Activities <input type="checkbox"/> Use External Apps <input type="checkbox"/> Create Polls <input type="checkbox"/> Bypass Slowmode <input type="checkbox"/> Send Voice Messages	Voice Permissions <input type="checkbox"/> Connect <input type="checkbox"/> Speak <input type="checkbox"/> Video <input type="checkbox"/> Mute Members <input type="checkbox"/> Deafen Members <input type="checkbox"/> Move Members <input type="checkbox"/> Use Voice Activity <input type="checkbox"/> Priority Speaker <input type="checkbox"/> Request To Speak <input type="checkbox"/> Use Embedded Activities <input type="checkbox"/> Use Soundboard <input type="checkbox"/> Use External Sounds <input type="checkbox"/> Set Voice Channel Status
---	--	--

4. Скомпілювати проект бота та запустити отриманий "discord_bot.exe". У Discord чаті ввести !run та код програми.

Висновок

У ході виконання завдання було успішно інтегровано консольний компілятор у середовище Discord. Реалізовано складну логіку керування дочірніми процесами у Windows та синхронізацію потоків для забезпечення стабільної роботи бота.