

NOKIA PARO 2022 Sprawozdanie W4

Memory management

1. Example_1

W zadaniu nie była zwalniana pamięć alokowana na stercie w linii 5. Rozwiązanie polegało na dodaniu operatora delete, który zwalnia przydzieloną pamięć.

2. Example_2

Podobnie do poprzedniego zadania, następował wyciek pamięci na skutek przydziału pamięci w funkcji `allocateInts()`, która później nie była zwalniana. W przykładzie istniała już zaimplementowana funkcja dealokująca pamięć, jednak nie była wywoływana, więc należało to zrobić.

3. Example_3

Problemem w tym zadaniu był fakt, że w bloku try funkcji `main`, jeśli wystąpił wyjątek podczas tworzenia obiektu klasy `Resource`, nie była zwalniana alokowana dla niego pamięć. Aby rozwiązać ten problem, importowano bibliotekę `memory`, aby mieć dostęp do mechanizmu smart pointerów. Wykorzystanie `std::unique_ptr` przy tworzeniu obiektu klasy `Resource` pozwoliło na zwolnienie alokowanej pamięci nawet w sytuacji, gdy wystąpił wyjątek. Ponadto zdefiniowano własną klasę `Custom_error`, dziedziczącą po `std::logic_error`.

4. Example_4

W przypadku zadania 4, problem z brakiem zwolnienia alokowanej pamięci występował w konstruktorze klasy `MyPointer`. Funkcja `foo` powodowała powstanie wyjątku, przez co destruktorem dealokujący pamięć na stercie przydzieloną w konstruktorze nigdy nie został wywołany. Wyciekowi zapobiegnięto zamieniając zwykły wskaźnik na `unique_ptr`, i używając `std::make_unique` w konstruktorze.

5. Example_5

W zadaniu 5, podobnie jak w poprzednich zadaniach, na skutek zwijania stosu przy wystąpieniu wyjątku nie były wykonywane instrukcje delete, które miałyby zwalniać przydzieloną wcześniej pamięć. Aby temu zapobiec (i przy okazji poprawić jakość kodu) zrezygnowano z manualnych prób użycia new i delete przy obsłudze obiektów Partner i oparto się na funkcjonalności unique_ptr, które pilnowały, aby została zwolniona pamięć w razie wystąpienia wyjątków.

6. Example_6

Rozwiązanie zadania 6 polegało na stworzeniu ciała funkcji makeFile oraz addToFile. Należało wykorzystać funkcje fopen, fclose, fprintf. W tym zadaniu nie było potrzeby modyfikacji wskaźników, jako że były one już zaimplementowane w optymalny sposób.