



Zadania dla kandydata z C++

- Zastosowanie bibliotek i narzędzi – wedle uznania
- Zadania są z szerokiego zakresu wiedzy, nie jest wymagane od kandydata rozwiązanie 100% zadań
- Doceniamy samodzielną pracę nad ilością rozwiązywanej treści. Zalecamy korzystanie z pomocy jak Google, Stack Overflow itp. (kod analizowany jest pod kątem generowania przez AI)
- Rozwiązanie należy przesłać w postaci linku do repozytorium na platformie GitHub
- Rozwiązanie może mieć dowolną formę (jednego pliku, kilku plików, oddzielnych projektów)
- Czas na rozwiązanie zadań – 5 dni roboczych.
- Odpowiedzi na pytania teoretyczne można zawrzeć w plikach z kodem w postaci komentarza, oddzielnego pliku, odpowiedzi w wiadomości e-mail.

Część teoretyczna

1. Jakie są różnice między wskaźnikami a referencjami w C++?
2. Wyjaśnij, jak działają inteligentne wskaźniki (smart pointers) w C++. Jakie są główne typy inteligentnych wskaźników dostępne w standardzie C++11?
3. Co to jest dziedziczenie wirtualne i kiedy należy go używać?

Część praktyczna

Zadanie 1 – Operacje na listach jednokierunkowych.

1. Wczytać z klawiatury n (n - stała) kolejnych danych o osobach (imię, nazwisko, wiek) i utworzyć z nich listę jednokierunkową w kolejności zgodnej z wczytywaniem.
2. Wydrukować utworzoną listę, korzystając z funkcji drukującej listę zaczynającą się pod jakimś adresem.
3. Usunąć drugą w kolejności osobę na liście oraz dodatkowo dopisać na drugim miejscu listy nowo wczytany z klawiatury element.
4. Ponownie wydrukować listę
5. Skasować utworzoną listę korzystając z funkcji kasującej listę zaczynającą się pod jakimś adresem.
6. Ponownie wydrukować listę.

Uwaga: Wykonując różne działania na liście (pkt 2-6) należy założyć, że nie znamy ilości elementów listy, więc trzeba uważać, aby nie odwoływać się do pola next elementu, który nie istnieje (czyli ma adres NULL); w szczególności należy sprawdzać, czy lista nie jest pusta (adres głowy == NULL) i czy ma ona dosyć elementów (aby nie wykonywać operacji niemożliwych do wykonania).



Zadanie 2 - Implementacja funkcji do obliczania silni liczby

Proszę o implementację funkcji obliczającej silnię liczby naturalnej, używając rekursji. Funkcja powinna być zaimplementowana w taki sposób, aby była bezpieczna i efektywna.

Założenia:

- Funkcja `factorial` powinna zwracać wynik typu `unsigned long long`.
- Dla liczby 0, funkcja powinna zwracać 1.
- Funkcja powinna być odporna na przepełnienia, jeśli wyliczana wartość przekracza zakres zmiennej `unsigned long long`.

Przykład użycia:

```
unsigned long long result = factorial(5); // Powinno zwrócić 120
std::cout << "5! = " << result << std::endl;
```

Zadanie 3 - mplementacja klasy `AutoBuffer` do zarządzania dynamicznie przydzielaną pamięcią.

Zaimplementuj klasę `AutoBuffer`, która będzie zarządzać dynamiczną tablicą danych typu `char`. Klasa ta ma umożliwić bezpieczne przechowywanie i manipulowanie ciągiem znaków.

Założenia:

- Konstruktor klasy powinien przyjmować rozmiar bufora.
- Klasa powinna zapewniać metody do zapisu i odczytu danych z bufora.
- Klasa powinna automatycznie zarządzać pamięcią, tzn. zwalniać bufor przy niszczeniu obiektu oraz prawidłowo obsługiwać operacje kopiowania i przenoszenia.

Przykład użycia:

```
AutoBuffer buf(10);
strcpy(buf.data(), "Hello");
std::cout << buf.data() << std::endl; // Powinno wyświetlić "Hello"
```