

Zaawansowane techniki programowania – lista zadań laboratoryjnych (Data kompilacji: 4 października 2022, t: 10:12)

1 Spis zadań do zrealizowania

1.1 Laboratorium nr 1 – Programowanie uogólnione i metaprogramowanie

Zadania związane z językiem C++ oraz metaprogramowaniem przy zastosowaniu wzorców:

1. Wykonać instalację biblioteki Boost w wybranym środowisku pracy dla języka C++ (zalecana jest obsługa najnowszych standardów C++).
2. Przygotować podstawowy przykład szablonu z funkcjami suma oraz iloczynu, a także zrealizować przykład z tzw. prostą tablicą.
3. Zaimplementować typ lista z użyciem szablonu. Przygotować zestaw testów do sprawdzenia poprawności listy.
4. Przygotować meta implementacje funkcji silnia, ciągu Fibonacciego, pierwiastka kwadratowego, bezpośrednio na wzorcach oraz przy wykorzystaniu pakietu MPL.
5. Wykorzystując pakiet HOF z biblioteki Boost, opracować przykłady prezentujące pojęcie funkcji wyższego rzędu.
6. Wykorzystując pakiet HOF z biblioteki Boost, zaprezentować przykłady funkcji typu print, sum, mult, max z wykorzystaniem podejścia „point-free”.
7. Wykonać przykłady z wykładu nr 1, dot. analizy liczb rozdzielonych przecinkiem oraz liczb rzymskich.
8. W oparciu o Boost.Metaparse lub Spirit (pakiet Spirit jest tu lepszym wyborem) zbudować parser dla wyrażeń arytmetycznych.

1.1.1 Różne uwagi związane z ćwiczeniem

Zadania wymagają nowoczesnego kompilatora języka C++ oraz instalacji biblioteki Boost w co najmniej wersji 1.6x lub najnowszej 1.7x. Należy też zwrócić uwagę na obsługę nowych wersji standardu C++, kompilator powinien obsługiwać nowe standardy tj. C++11, C++14, C++17, i najnowszy C++20 (C++0x, C++1y, C++1z, C++2a).

1.2 Laboratorium nr 2 – Erlang

Podstawowe zadania związane z obsługą Erlanga¹, tj. napisać, zaimplementować, podać kod realizujący następujące zadania:

1. Uruchomić kompilację z poziomu konsoli wg przykładu na wykładzie.
2. Napisać klasyczny program do rozwiązywania równania kwadratowego w Erlangu.
3. Zaimplementować rekurencyjne wersje funkcji silnia, ciągu Fibonacciego.
4. Zaprezentować działanie trzech dowolnie wybranych funkcji z modułu lists.
5. Operacje na listach:
 - (a) odczyt ostatniego elementu listy,

¹Przydaną stroną do nauki podstaw Erlanga jest: <https://learnyousomeerlang.com/>.

- (b) odczyt przedostatniego elementu listy,
 - (c) odczyt n-tego elementu listy,
 - (d) ustalenie długości listy, tj. liczby elementów znajdujących się na liście,
 - (e) usunięcie n-tego elementu listy.
 - (f) odwrócenie kolejności elementów na liście,
 - (g) sprawdzenie, czy lista jest palindromem.
6. Utworzenie listy zawierającej liczby od wartości a do b np. range 3 9 \rightarrow 3,4,5,6,7,8,9.
 7. Podać rekurencyjne konstrukcje podobne do pętli while oraz for.
 8. Podać kod tworzący trójkąt Pascala.
 9. Implementacja sortowania przez scalanie, oraz alg. sortowania szybkiego.
 10. Zakładając że mamy listę, sekwencję liczb, usunąć np. liczby parzyste, lub nieparzyste za pomocą tzw. filtru².
 11. Utworzyć własną wersję konstrukcji fold. I pokazać jej zastosowanie
 12. Zaimplementować kalkulator w odwrotnej notacji polskiej.
 13. Zaimplementować typ zbiorowy (ang. sets).
 14. Zaimplementować sito Eratostenesa.
 15. Zdefiniować rekord/krotkę opisujący osobę, film, książkę.

1.3 Laboratorium nr 3 – Procesy i współbieżność, OTP

Lista zadań związana z procesami i współbieżnością jaką oferuje Erlang.

1. Wykonać podstawowy przykład dotyczący procesu self, jakie znajduje się na wykładzie.
2. Zrealizować następujące przykłady:
 - (a) z procesem „bounce”,
 - (b) z procesem „loop”,
 - (c) z maszyną stanu.
3. Pokazać na podstawowych przykładach zastosowanie after oraz pojęcia „timeout”.
4. Opracować przykłady naśladowujące za pomocą procesów działanie podstawowych typów pętli.
5. Opracować przykład ping-pong.
6. Opracować przykład demonstrujący technikę podmieniania kodu.
7. Opracować komunikator do przesyłania krótkich komunikatów tekstowych do zalogowanych użytkowników.

1.4 Laboratorium nr 4 – ETS, DETS, Mnesia

Lista zadań związanych z systemem bazy danych Mnesia.

1. Opracować bazę danych typu książka telefoniczna, biblioteka.
2. Podać definicję niezbędnych rekordów dla wybranej bazy danych.
3. Podać definicję klauzul, dla łatwiejszego wprowadzania, usuwania oraz modyfikacji danych.
4. Wprowadzić możliwość odszukiwania danych. Można wykorzystać QLC celem uproszczenia procesu tworzenia zapytań.

²Np. dla listy: `Numbers = lists:seq(1,10).`, konstrukcja `user:filter(fun(X) -> X rem 2 == 0 end, Numbers).`, utworzy nową listę z liczbami parzystymi.

1.5 Laboratorium nr 5 – Podstawy informatyki kwantowej

1. Mamy wektory:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

wyznaczyć wartość następujących iloczynów tensorowych

- (a) $|0\rangle \otimes |0\rangle, |1\rangle \otimes |0\rangle, |1\rangle \otimes |1\rangle$
- (b) $|010\rangle, |111\rangle$
- (c) $|0\rangle, |1\rangle, |2\rangle, |3\rangle, |4532\rangle$

2. W C^2 istnieją wektory: $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle), |\theta\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Obliczyć następujące wektory:

- (a) $|\psi\rangle \otimes |\theta\rangle, |\theta\rangle \otimes |\psi\rangle$
- (b) $|\psi\rangle \otimes |\psi\rangle \otimes |\theta\rangle, |\theta\rangle^{\otimes 4}$

3. Oblicz iloczyny tensorowe (I, X, Y, Z to macierze Pauliego):

- (a) $X \otimes Y, X \otimes Z$
- (b) $I \otimes X, X \otimes I$
- (c) $X \otimes Y \otimes Z, X \otimes X \otimes X$

1.6 Laboratorium nr 6 – Podstawowe protokoły i algorytmy informatyki kwantowej

1. Ślad macierzy M jest sumą elementów leżących na głównej przekątnej:

$$\text{Tr}(M) = \sum_{i=1}^n M_{ii}.$$

Pokazać iż zachodzą następujące własności:

- (a) $\text{Tr}(A + B) = \text{Tr}(A) + \text{Tr}(B)$
- (b) $\text{Tr}(cA) = c\text{Tr}(A)$
- (c) $\text{Tr}(AB) = \text{Tr}(BA)$

2. Narysować oraz sprawdzić w symulatorze obwód który wygeneruje stan:

$$\frac{1}{\sqrt{8}} \sum_{x \in \{0,1\}^3} |x\rangle = \frac{|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle}{\sqrt{8}}$$

dla układu trzech qubitów ze stanu początkowego $|0\rangle$.

3. Zbudować obwód do generowania liczb losowych:

- w zakresie od zero do $2^n - 1$, podać ogólną postać,
- w określonym zakresie np. od 0 do 127,
- dla określonych liczb np. od 33 do 63.

4. Zaprojektować obwody do tworzenia par Bella i stanów GHZ tj.:

- $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), |\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),$
- $|\psi\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), |\psi\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle),$
- $|\psi\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle).$

5. Stan dwóch niesplątanych qubitów, czyli stan separowalny można przedstawić jako następujący iloczyn tensorowy: $|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle$. Pokazać, że nie można przedstawić stanu: $|\psi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ (nazywanego również stanem Bella) w postaci iloczynu tensorowego dwóch qubitów.
6. Wykazać, dlaczego dla stanu czystego reprezentowanego przez macierz gęstości ρ , prawdziwa jest następująca zależność:

$$\text{Tr}(\rho^2) = \text{Tr}(\rho) = 1$$

uogólniając:

$$\text{Tr}(\rho^2) = \begin{cases} 1 & \text{dla stanów czystych} \\ s, s \in [0, 1] & \text{dla stanów mieszanych} \end{cases}$$

7. Pokazać na przykładzie obwodu kwantowego działanie alg. Deutch-Jozsy.
8. Zaimplementować standardowy protokół teleportacji kwantowej.
9. Opracować przykład prezentujący alg. Grovera. Zwrócić uwagę na postać wyroczni oraz realizacji operatora U_f oznaczającego stan znakiem minus.
10. Opracować przykład prezentujący alg. Shora rozkładu liczby na czynniki.
11. Pokazać działanie protokołu BB84 na krótkim np. ósmioelementowym ciągu bitowym.

2 Dalsze informacje

Przegląd pozycji książkowych (oraz adresów w sieci Internet) dotyczących zagadnień z przedmiotu „Zaawansowane techniki programowania”.

Literatura

- [1] David Abrahams, Aleksey Gurtovoy, Język C++. Metaprogramowanie za pomocą szablonów, Helion, 2005.
- [2] Boost Library, <https://www.boost.org/>, 2021.
- [3] The Boost C++ Libraries, <https://theboostcpplibraries.com/>, 2021.
- [4] Język Erlang, <https://www.erlang.org>, 2021.
- [5] Walter Greiner: Quantum Mechanics An Introduction, Fourth Edition, Springer-Verlag Berlin Heidelberg 1989, 1993, 1994, 2001
- [6] Giuliano Benenti, Giulio Casati, Giuliano Strini: Principles of Quantum Computation and Information Volume I: Basic Concepts, World Scientific 2005
- [7] Julian Klukowski, Ireneusz Nabiałek: Algebra dla studentów, Wydawnictwo Naukowo-Techniczne 2004
- [8] Mika Hirvensalo: Algorytmy kwantowe, Wydawnictwa Szkolne i Pedagogiczne 2004
- [9] Krzysztof Giaro, Marcin Kamiński: Wprowadzenie do algorytmów kwantowych, Akademicka Oficyna Wydawnicza Exit, Warszawa 2003