Вивчить собі Хаскела на велике щастя!

Автор: Міран Ліповача

Переклад здійснили: Ганна Лелів, Семен Тригубенко, Богдан Пеньковський, Марина Стрельчук і Тетяна Богдан

Мовні редактори: Тетяна Богдан і Ганна Лелів
Науковий редактор: Семен Тригубенко

Переклад виконано за підтримки
Словненя

2017-05-21T00:06:06Z
Версія v4.7-54-gda41cf2

Зміст

Передн̀є слово

0.1 Щодо мотивації

Мотивацій створення української версії «Learn You a Haskell for Great Good!» було декілька. Першопочатково дуже хотілося привернути увагу українського програміста до цієї абсолютно дивовижної мови і допомогти її опанувати. Хаскел — сама по собі непроста мова програмування, і бар'єрів розуміння там предостатньо й так — тож нам дуже хотілося прибрати хоча б один із них — мовний.

У Великобританії Хаскел викладають на перших курсах деяких університетів. Першокурсники — і ті, хто вже добре програмує імперативно, і ті, хто взагалі ще не програмував, — на цих курсах починають разом «з нуля». І синтаксис, і абстракції, і функційний стиль — все це настільки далеке від «звичайного», що, вкупі із браком досвіду і знань, отримуємо старт без гандикапу для усіх. Це допомагає як викладачам, так і студентам. Але є іще одне: **закон першості** [primacy law] в навчанні: речі, вивчені уперше, справляють найсильніше враження, яке потім дуже важко стерти. Тому вивчити C, а потім Хаскел — це не одне й те саме, що вивчити Хаскел, а потім C.

В мене є мрія, щоб Хаскел почали викладати і в українських навчальних закладах — і цей переклад є одним із кроків до її здійснення. В Могилянці ми писали на Fortran, C/C++, Mathematica і Maple, а Хаскел я вивчив через десять років власноруч. Але якби я міг повернутися назад в 1997-ий, я б вивчив саме Хаскел. Чому? Хаскел — це фундаментальна, безкомпромісна мова, яка вимагає якісних програм, а не просто «дає змогу писати якісні програми». Вчити інші мови програмування, а потім все життя запам'ятовувати ідіоми, які дозволяють не вистрелити собі у ногу в невдалий момент — це один із неоптимальних шляхів до програмістської нірвани. Кращий шлях — вивчити функційне програмування одразу і перейти на новий рівень в боротьбі із **складністю** [complexity], якою і є, по суті, програмістське життя.

Але мені хотілося б, щоб студенти вчили Хаскел не тільки тому, що він такий класний. Насправді, Хаскел має й недоліки (в кого їх немає?). Головним є те, що Хаскел документує чесноти, які варто берегти, і власне розуміння то-

го, із чим Хаскел бореться і за що воює, — це вже неабиякий крок до **просвіти** [enlightenment], незалежно від того, чи перемагає Хаскел в цій боротьбі, чи програє, і це стане в пригоді навіть якщо ви далі будете писати код якимись іншими мовами.

0.2 Про цей переклад

Половину цієї книги переклала Ганна Лелів — і без її внеску ми б ніколи не завершили переклад. Іноді вона перекладала швидше, ніж ми вичитували і компілювали сирці LaTeX. Так чи так — Ганна Лелів або суперлюдина суперперекладач, або це команда перекладачів, яка працює під цим ім'ям.

Ми намагалися не надавати переваги усталеній термінології тільки тому, що вона усталена, а обирали термінологію, балансуючи усталеність із доречністю. Через то деяку термінологію було розроблено спеціально для цього перекладу. Неабиякий внесок в розробку нових термінів зробила Тетяна Богдан (всі терміни, які вам здаються влучними і доречними, розробила вона, а невдала термінологія — то є насправді компроміс, який було досягнуто в результаті довгої редакторської боротьби із іншими редакторами і читачами :D). Тетяна розробила набагато більше термінів, ніж нам вдалося тут використати, але ми їх усіх десь зберегли. Тож, якщо вам потрібен новий термін — напишіть нам — можливо він в нас вже є!

Оригінал містить неабияку кількість жартів і вони, разом із малюнками й стилем подання матеріалу, і роблять цю книгу цією книгою. Більшість з цих жартів перекласти «в лоб» було неможливо. Для деяких вдалося знайти заміну «на місці» — наприклад, підрозділ шостого розділу «Згортком і батька легше бити» в оригіналі називався «Лише згортки і коні», перекручуючи фразу «Лише дурні і коні» («Only fools and horses»). Хай там як — чи то «дурням везе на перегонах», чи то «тільки дурні й коні працюють» — в оригінал цієї книги ця фраза потрапила лише тому, що (1) вона дуже розповсюджена в англомовних колах (ще й британський сітком так називається) і (2) бо слова «fool» і «fold» трохи схожі. В таких випадках ми намагалися знайти схожий за стилем український відповідник. Але бувало, що жарти занадто «впліталися» в зміст книги, і «прикрутити» український жарт в тому ж місці в перекладі означало б менш природні формулювання чи втрату змісту заради жарту. В таких випадках ми перекладали зміст із втратою жарту, але намагалися додати власний жарт десь іще (закон збереження кількості жартів в LUaH :-)).

Багато неточностей в перекладі траплялося через плутанину навколо дієслів означа́ти, озна́чити, озна́чувати, познача́ти, позна́чити, визнача́ти і ви́-

значити[*]. Хоча в українській мові іменники «означе́ння» і «ви́значення» для деяких значень цих багатозначних слів[†] є взаємозамінними еквівалентами, в цій книзі ми використовували іменник «означе́ння» суто для перекладу англійського іменника «definition». Відповідні дієслова до іменника «означе́ння» — «означи́ти» і «означа́ти» (доконаний і недоконаний види, відповідно) — використовувалися отак: «We need to define…» перекладалося доконаним «Нам треба означи́ти…», в той час як *процес* надання означення перекладався недоконаним «означа́ти» (а не підступним «означа́ти»[‡]!): «When defininings functions we…» — «Коли ми означа́ємо функції…». Загальне «We define…» також перекладалося недоконаним: «We define this only when…» — «Ми означа́ємо це лише коли…». Щодо іменника «ви́значення», то він використовувався виключно в значенні «процес з'ясовування чогось», як-от, наприклад, в «оператор визначення зони видимості» чи в «функція для визначення загальної категорії». Відповідні йому дієслова — ви́значити і визнача́ти — перекладали англійські дієслова «to determine», «to figure out», «to find out» і таке інше. Синонімічне «to infer» не перекладалося як «визнача́ти» — скрізь використовувалося більш точне в контексті Хаскела «вивóдити». Дієслово «означа́ти» перекладало англійське «to mean», а «познача́ти» і «позна́чити» — «to denote».

Англомовний варіант називає конструкції із ключовими словами instance і class оголошеннями (instance declarations і typeclass declarations, відповідно), а ми перекладаємо їх як instance-означення і class-означення. У цих випадках межа між оголошенням і означенням може бути доволі розмитою і, здається, для загальних концепцій «означення» і «оголошення», ці конструкції є ближчими до означень ніж до оголошень, бо в конструкціях із instance власне й означуються тіла функцій, яких типоклас, що втілюється, вимагає, і в такий спосіб означується втілення, а в конструкціях із ключовим словом class, хоча там зазвичай оголошуються функції, яких вимагає типоклас, все ж таки, в такий спосіб і означується типоклас, — себто, оголошенням функцій, яких він вимагає. До того ж, в означеннях типокласів можна надавати й означення тіл функцій, яких вимагає той типоклас, за замовчуванням, і це робить цю кон-

[*]Іноді кришталева ясність щодо розуміння цих слів поодинці і в цілому в деяких крайових випадках переходила в туманну аморфність; нас турбувала доля в українській мові слів «позна́чування», «визна́чування», «позна́чувати» і «визна́чувати» (яких тут не вистачає для повного комплекту), і непокоїло те, що в слові «ви́значити» для симетрії наголос мав би бути в іншому місці.

[†]No pun intended.

[‡] Якби ми не були знайомі із парочкою «означа́ти» і «означи́ти», то могли б подумати, що вони є недоконаним і доконаним видами дієслова якогось одного значення (за аналогією із «познача́ти» і «позна́чити» та «визнача́ти» і «ви́значити» [от тільки в «ви́значити» наголос не там, хай йому ґрець!]). Але насправді вони є недоконаним і доконаним видами дієслів різних значень.

струкцію ще близькішою до концепції «означення».

«Як» в українській мові перевантажено його синонімією до «що». Для «як» в значенні «що», речення «Ми побачили, як типокласи можуть представляти круті концепції» і «Ми побачили, що типокласи можуть представляти круті концепції» означають одне й те саме, і перекладаються як «we've seen *that* typeclasses can represent cool concepts». Проте для його значення «в який спосіб» ці два речення перекладаються англійською як «we've seen *how* typeclasses can represent cool concepts» і «we've seen *that* typeclasses can represent cool concepts», відповідно. В другому випадку, «ми побачили, що типокласи можуть представляти...» і все, а в першому — ще й дізналися, *як саме* вони «можуть представляти». В цій книзі ми намагалися використовувати «що» замість ««як» в значенні «що»», тому, якщо ви бачите «як», швидше за все йдеться про «в який спосіб».

В хаскельному коді розрізняти параметри й аргументи не так важливо (в порівнянні, наприклад, із C++, де параметр і відповідний аргумент можуть мати різні типи, і які саме перетворення аргументів в параметри відбуваються при переході крізь цей інтерфейс має неабияке значення). Але ми в цьому перекладі все одно використали точнішу термінологію. Наприклад, в означенні `f x = x * x` і виразі `f y where y = 2` маємо: `x` є параметром функції `f`, а `y` — аргументом функції `f`. `y` має значення `2` і є доступним в тілі функції `f` під ім'ям `x`. Отже, параметри «живуть» в типосиґнатурі і тілі функції, і це є те, із чим аргументи зв'язуються в точці виклику функції — і тому функції можуть брати і аргументи, і параметри — залежно від того, чи йдеться мова про те, як функцію було означено, чи про те, як саме вона викликається.

Рядок — багатозначне слово в контексті цієї книги — може означати як структуру даних (`String`), так і просто рядок — тексту чи коду. Хоча зазвичай з контексту зрозуміло, про який рядок йдеться, ми структуру даних позначили просто «рядок», а в решті випадків писали «рядок тексту» і «рядок коду».

Мені особисто довелося прочитати цю книгу в середньому шість разів (на рівні розділів: двічі англійською, і тричі українською — перший раз англійською, потім українською і англійською одночасно, потім українською повільне читання, а потім українською — швидке; і один раз усієї книги українською). «В середньому», бо деякі місця довелося перекладати `n` разів «до збіжності» :), а деякі — повертатися і переперекладати, бо з'являлися зауваження і пропозиції редакторів і читачів, чи то просто нові ідеї. Іноді редагували, доки усі вимоги не було задоволено — редакторів, читачів, перекладачів, або було доведено, що немає розв'язку і тоді лишали авторський варіант (перекладача), або не перекладали. Хоча й текст перекладено багатьма, ми спробували витримати деякі інваріанти — консистентності мови, стилю і термінології по усій книзі — тому довелося повертатися багато разів до вже відшліфованого матеріалу,

зітхати, ламати його і будувати знову. Тож ми сподіваємося, що ви прочитаєте цю книгу принаймні двічі, а деякі місця цитуватимете при нагоді в повсякденному житті! :D (жартую). Якщо серйозно — якщо ви помітили помилку, неточність, втрату змісту в перекладі, або ж маєте якісь загальні коментарі чи пропозиції щодо покращення змісту, будь ласка, листуйте за адресою semen [at] trygub.com, і ми втілимо слушні зауваження в життя. Звичайно ж, якщо вам сподобалася книга і переклад — однаково пишіть, для балансу :) та й нам цікаво буде почути, кому і в який спосіб ця книга стала в пригоді!

Автори українського перекладу «Вивчить собі Хаскела...» складають щиру подяку Дмитру Сіренку, без критичних зауваг якого обробка результатів вичитування ранніх версій була б завершена на два порядки швидше, і Тарасу Бунику, хто був першим «справжнім» читачем цього підручника, а також (в автобіографічному порядку): Олегу Бурхаю, Дмитру Рудзону, Петру Єрмоленку, Олексію Сивоконю, Матвєю Аксьонову, і Михайлу Іванкову.

Семен Тригубенко

Покажчик