

## **АСЕМБЛЕР: ПОЧАТОК ПРОГРАМУВАННЯ**

Більшість прикладного і значна частина системного програмного забезпечення створюється мовами високого рівня. Програми на мовах високого рівня легше пишуться і читаються, значно швидше розробляються і відлагоджуються, їх легко переносити на інші платформи.

Мова Асемблера належить до мов програмування низького рівня. Мовою Асемблера пишуться найбільш критичні ділянки коду, системне програмне забезпечення, драйвери та утиліти. Проблема навчання мовам програмування високого рівня присвячено багато досліджень вітчизняних та зарубіжних вчених. Питання підвищення ефективності засвоєння Асемблера практично не піднімаються – вважається, що мова Асемблера – це доля професіоналів.

Аналіз труднощів, з якими мають справу студенти, особливо на перших кроках вивчення Асемблера, свідчить, що підвищення ефективності засвоєння мови Асемблера студентами вищих навчальних закладів можна досягти шляхом раціональної організації навчальних занять, розробки проблемних навчальних завдань, розвитку загальних навичок розв'язування задач і алгоритмізації, залученням навчальних інтегрованих середовищ розробки програм, спеціально призначених для вивчення мови Асемблера.

Метою даної статі є обґрунтування доцільності використання інтегрованого середовища програмування та методика використання концепції “від простого до складного” при вивченні мови Асемблера.

Для швидкого та якісного старту у вивченні Асемблера автором пропонується використати інтегроване середовище, що включає:

- Редактор тексту: Fasm Editor 2.0; (<http://asmworld.ru/instrumenty/fasm-editor-2-0/>);
- Компілятор: Flat assembler (<http://flatassembler.net/>);
- Де-багер: OllyDbg (<http://www.ollydbg.de/>).

Інтегроване середовище вже містить в собі редактор тексту, компілятор та де-багер. Після налагодження середовища програмування (шлях до компілятора, де-багера та під'єднаних файлів), можна приступати до безпосередньо написання коду програми на Асемблері. Після написання коду, код компілюється. Якщо відсутні синтаксичні помилки, то відразу створюється .exe файл. Робота створеного .exe файлу аналізується за допомогою де-багера. Фактично, за допомогою де-багера, крок за кроком, аналізується хід виконання програми (робота алгоритму, вміст регістрів тощо). Це дає змогу на початкових кроках поєднати лекційний матеріал з практичними знаннями.

Основою (початковою) вивчення програмування Асемблером є:

1. Використання двійкової, шістнадцяткової систем числення;
2. Представлення від'ємних чисел у доповнюючому коді;
3. Використання регістрів для збереження даних;
4. Найпростіші команди (обміну, логічні, арифметичні, зсув);
5. Команди порівняння (розгалуження обчислювального процесу).

Використовуючи інтегроване середовище програмування студенти, заносючи різні (по величині та знаку) числа, мають можливість перевірити власні знання про перетворення чисел з різних систем числення, формування доповнюючого коду для від'ємних чисел, а також здатності певних регістрів зберігати числа певного діапазону (розрядність регістрів). Після засвоєння систем представлення чисел студенти переходять до аналізу команд (обміну, логічних, арифметичних, зсуву), приділяючи увагу роботі як з додатними, так і від'ємними числами. Як один з варіантів розгалуження обчислювального процесу аналізується команда порівняння та різні можливості використання цієї команди.

Таким чином, новизна представленої роботи полягає у використанні інтегрованого середовища, що включає редактор тексту, компілятор, де-багер для швидкого старту у вивченні програмування Асемблером. Як показав практичний досвід 2012-2016 років поєднання інтегрованого середовища з методологією “від простого до складного” в значній мірі полегшує складність перших кроків у засвоєнні програмування на Асемблері у навчальному закладі та підвищує продуктивність навчального процесу в цілому.