

СФЕРЫ ИСПОЛЬЗОВАНИЯ ВЫСОКОУРОВНЕВЫХ И НИЗКОУРОВНЕВЫХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

В работе рассматриваются особенности применения высокоуровневых и низкоуровневых языков программирования, производится сравнение их по выбранным параметрам, определяющим особенности их работы.

ВВЕДЕНИЕ

Любая компьютерная программа – это последовательность инструкций, которая предназначена для исполнения вычислительной машиной (компьютером). Сами инструкции задаются человеком при помощи языка программирования – формальной знаковой системой, предназначенной для записи программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, задающих внешний вид программы и действия, которые выполнит исполнитель (компьютер) под её управлением.

I. ВЫСОКОУРОВНЕВЫЙ ЯЗЫК ПРОГРАММИРОВАНИЯ

Язык высокого уровня - это язык программирования, предназначенный для оптимизации компьютерного программирования. Он называется высокоуровневым, потому что он находится в нескольких шагах от исходного кода, выполняемого на центральном процессоре компьютера (ЦП). Исходный код высокого уровня состоит из легко читаемой структуры, которая впоследствии переводится на язык низкого уровня, который может восприниматься и запускаться конкретным процессором. Самые популярные языки программирования считаются языками высокого уровня. Примеры включают в себя: C ++, C, COBOL, Fortran, Java, JavaScript, Pascal, Perl, PHP, Python.

II. НИЗКОУРОВНЕВЫЙ ЯЗЫК ПРОГРАММИРОВАНИЯ

Низкоуровневые языки программирования - это языки программирования, которые работают с аппаратными элементами и ограничениями компьютера. Языки низкого уровня считаются ближе к компьютерам. По сути, их основная функция - маневрировать, администрировать и выполнять вычислительное оборудование и элементы. Программы и приложения, написанные на низкоуровневом языке, непосредственно ре-

ализуются на компьютерном оборудовании без какой-либо интерпретации или перевода.

III. ВЫВОДЫ

Язык высокого уровня - это место, где происходят все инновационные вещи, и его можно отладить гораздо проще, чем на низком уровне. Легко развертываются и могут работать на нескольких платформах и архитектурах. Язык низкого уровня требует большого опыта и знаний. Сложны в понимании и отладки другими программистами. Код, написанный для одного устройства, невозможно запустить на другом, то есть он не переносим. Программирование на ассемблере - крайне длительный процесс. Во множестве случаев, 10-20 страниц кода на ассемблере заменяет страница кода на языке высокого уровня. На ассемблере трудно заниматься написанием визуальных компонентов - на это уйдет много времени. Ассемблер предлагает небольшое количество полезных библиотек. Самым главным плюсом ассемблера это то, что знание языка облегчает понимание архитектуры компьютера и работы его аппаратной части, то, чего не может дать знание языков высокого уровня. Главный плюс высокоуровневых языков - не надо вникать в тонкости платформы и расплываться на элементарные операции, а можно сразу концентрироваться на абстракциях, логике, красивой архитектуре. Это исследование, которое должно дать хорошую базу людям, которые не имеют представления о языках программирования. Каждый начинающий программист должен провести свои исследования, чтобы понять, какие бывают пути в обучении программированию. Если вы хотите научиться управлять памятью, то вам придется работать с низкоуровневыми языками и принять все сопутствующие факторы.

Список использованных источников:

1. <https://ru.wikipedia.org/wiki/Высокоуровневый>
2. <https://ru.wikipedia.org/wiki/Низкоуровневый>
3. <https://tproger.ru/translations/programming-languages-types/>

Сечко Роман Александрович, Маликов Марат Васильевич, студенты кафедры вычислительных методов и программирования БГУИР, strong.roma1999san@gmail.com.

Научный руководитель: Шатилова Ольга Олеговна, старший преподаватель кафедры вычислительных методов и программирования БГУИР, shatilova@bsuir.by .