АНАЛИЗ ДОСТОИНСТВ И НЕДОСТАТКОВ РАЗЛИЧНЫХ КОМПИЛЯТОРОВ С ПОДДЕРЖКОЙ АВТОМАТИЧЕСКОЙ ПАРАЛЛЕЛИЗАЦИИ

к.т.н., доц. И.С. Зыков, магистр Н.В. Кутя, С.Г. Межерицкий, Национальный технический университет "ХПИ", г. Харьков

Проанализированы достоинства и недостатки с точки зрения кроссплатформенности, подробности вывода, скорости компиляции, размера и производительности получаемого исполняемого файла для компиляторов языков C и C++ с поддержкой автоматической параллелизации. В сравнении участвовали компиляторы GCC, Visual Studio, clang, Open64 и Intel C/C++ Compiler.

В результате сравнения были сделаны выводы о том, что:

− по кроссплатформенности найлучшим компилятором является GCC, который поддерживает практически все существующие на сегодня аппаратные платформы и большую часть современных операционных систем;

− по подробности вывода найлучшим является компилятор clang, который выводит предупреждения, не выводимые другими компиляторами;

− скорость компиляции и параметры получаемого исполняемого файла варьируются в зависимости от системы, в которой запущен компилятор, и платформы, для которой создаётся исполняемый файл, а так же сложности задачи.

В результате анализов не было найдено данных о зависимости параметров получаемого исполняемого файла в зависимости от сложности задачи. В силу этого были предложены тесты, позволяющие судить о производительности автоматической параллелизации в зависимости от компилятора, сложности задачи и количества ядер, доступных программе.

С помощью ручной автоматизации, предоставляемой OpenMP, определены оптимальные параметры параллелизации для задач различной сложности и различного количества ядер. Сделано сравнение компиляторов с поддержкой автоматической параллелизации с параллельными модификациями языков C и C++, такими как Intel Cilk, Intel Cilk+ и Unified Parallel C. Проанализированы отличия от языков, изначально ориентированных на параллельность, таких как Ada, Go и Limbo. Также рассмотрена возможность проведения вычислений на аппаратных графических ускорителях с помощью CUDA и OpenCL.

Сделаны выводы о необходимости улучшения критериев автоматической параллелизации в компиляторах C и C++ или переносе принятия решения об автоматической параллелизации на стадию времени исполнения.