Computer Science 301 - 2017

Programming Language Translation

Practical 1, Week beginning 17 July 2017

Comparison of the sizes of the output from various compilers

	Empty	Sieve	NPrimes	Fibonacci	Queens1	Queens2	Queens3
FPC Pascal EXE Windows 7-32							
FPC Pascal EXE Windows 10-64							
FPO Pascal EXE Optimised							
FPS Pascal EXE Stripped							
Turbo Pascal 6 EXE (VDosPlus/DosBox) TP6							
Turbo Pascal 6 EXE (VDosPlus/DosBox) TP60							
Turbo Pascal 3 COM (VDosPlus/DosBox)							
Parva PVM							
Parva -O optimize PVM							
C# EXE							
Parva2toC# EXE							
Borland C EXE							
Borland C++ EXE							
Watcom C EXE							
Watcom C++ EXE							

Note that the C# compiler does not really produce a standalone EXE file. The .COM file produced by TurboPascal 3 is the closest one gets to an EXE file on that system.

What do you observe and what conclusions can you draw? Note that the source programs in each case are pretty well equivalent - fairly close manual translations of one another.

Comparison of the run time speeds of the executables produced by various compiler.s.

Try to choose iteration counts and limits so that the fastest programs still give measurable times (at least half a second) while the slowest ones still run without going on for ever

	Iterations & Limit	Sieve	Queens1	Queens2	Queens3
Iterations					
Limit					
FPC Pascal EXE Windows 10-64					
FPO Pascal EXE Windows 10-64					
Turbo Pascal 6 EXE VDosPlus TP6					
Turbo Pascal 6 EXE VDosPlus TP60					
Turbo Pascal 3 COM VDosPlus					
Turbo Pascal 6 EXE DosBox TP6					
Turbo Pascal 6 EXE DosBox TP60					
Turbo Pascal 3 COM DosBox					
Parva PVM					
Parva -O optimize PVM					
C# EXE Windows 10-64					
Borland C EXE Windows 10-64					
Borland C++ EXE Windows 10-64					
Watcom C EXE Windows 10-64					
Watcom C++ EXE Windows 10-64					

What do you observe and what conclusions can you draw? Note that the source programs in each case are pretty well equivalent - fairly close manual translations of one another.