Github repository with your code: https://github.com/savioarthu/HPCesga_Tools.git

	Optimization+platform dependent options for compilation and execution times for my_dgesv	
	icc (18.0.1)	**gcc (8.1.0)**
No opt.	Options: -O0 Exec time (small): 56.99s Exec time (medium): 1194.96s Exec time (large): 12854.36s	Options: -O0 Exec time (small): 65.35s Exec time (medium): 708.43s Exec time (large): 10009.24s
Opt level 1	Options: -O1 Exec time (small): 32.12s Exec time (medium): 449.15s Exec time (large): 5788.26s	Options: -O1 Exec time (small): 56.09s Exec time (medium): 547.85s Exec time (large): 7868.10s
Opt level 2 + specific arch	Options: -march=native -O2 Exec time (small): 23.64s Exec time (medium): 374.96s Exec time (large): 4468.79s	Options: -march=native -O2 Exec time (small): 55.88s Exec time (medium): 556.08s Exec time (large): 7882.48s
Opt level 3 + specific arch	Options: -march=native -O3 Exec time (small): 24.07s Exec time (medium): 354.44s Exec time (large): 4428.27s	Options: -march=native -O3 Exec time (small): 56.22s Exec time (medium): 549.81s Exec time (large): 7926.92s
Opt level fast + specific arch	Options: march=native -Ofast Exec time (small): 24.06s Exec time (medium): 360.55s Exec time (large): 4438.17s	Options: -march=native -Ofast Exec time (small): 56.66s Exec time (medium): 558.40s Exec time (large): 7930.67s
Opt level fast + specific arch + interproc opt/anal [ipo (icc) / -fipa-pta (gcc)]	Options: -march=native -Ofast -xcore-avx2 -ipo Exec time (small): 24.15s Exec time (medium): 348.33s Exec time (large): 4408.24s	Options: -march=native -Ofast -fipa-pta Exec time (small): 56.58s Exec time (medium): 567.08s Exec time (large): 7927.74s
All previous opts + pgo	Options: -march=native -Ofast -xcore-avx2 -ipo -prof-gen Exec time (small): 60.23s Exec time (medium): 1109.47s Exec time (large): 14642.08s	Options: -march=native -Ofast -fipa-pta -fprofile-generate Exec time (small): 82.57s Exec time (medium): 793.61s Exec time (large): 10994.04s
Others: autovectorizing	Options: -qopt-report-phase=vec -O3 Exec time (small): 23.03s Exec time (medium): 339.76s Exec time (large): 4459.44s	Options: -ftree-loop-vectorize -O3 -fopt-info-vec Exec time (small): 38.98s Exec time (medium): 491.56s Exec time (large): 7312.11s
Autoparallelism	Options: -parallel Exec time (small): 23.08s Exec time (medium): 359.20s Exec time (large): 4416.71s	

Matrix A size for executions, according to Makefile:

- small size: 1024 x 1024 (execute as ./dgesv 1024)
- medium size: 2048 x 2048 (execute as ./dgesv 2048)
- large size: 4096 x 4096 (execute as ./dgesv 4096)

I executed my task1 dgesv program 3 times per combination and each time I took the median value. I slightly modified the sizes so that it doesn't take too long to run.

Description and relevant information extracted from the results:

In general, icc compilation is faster than gcc. In both cases, the program takes a long time to run. The level 1 and level 2 option for icc significantly reduces the execution time while the other optimization combinations don’t improve the time any further. For gcc, the times are the same for between level 1 and level 2 but remain slightly better than without optimization.
The combination of all the options associated with pgo significantly increases the time for both ICC and gcc.
Autovectorizing is finally the most efficient in both cases (especially for gcc).
Finally, the parallelization option for icc doesn’t change anything since the code is difficult to parallelize.