MODELO DE EXAMEN FINAL LIBRE – MODALIDAD A DISTANCIA

PARTE A: COMPRENSIÓN

✹ **Lea el texto con atención y responda las siguientes preguntas en <u>español,</u> tratando de usar sus palabras.**

System benchmarks

We have noted that the best way to compare two or more computer systems is the most direct way possible: Run one's application of interest on each system and measure the time taken to compute a result (or perform typical tasks associated with the application). The system that gets that specific task or tasks done the fastest is the one that has the highest performance in the only sense that really matters. However, it is not always possible to run one's actual code on every system that merits consideration. One may not have physical access to the systems prior to making the decision, or it may not be possible to compile and run the program on every platform of interest, or one may just not have enough time to do so. One may even be specifying a system to run a future application that has not yet been developed. In this case, the next best thing is to identify a standard *benchmark* program, or suite of programs, that performs tasks similar to those of interest.

A benchmark is a program or set of programs chosen to be representative of a certain type of task. The idea is that if a system performs well on the benchmark, it will likely perform well on other applications with similar characteristics; if it performs poorly on the benchmark, it is not likely to be well suited for applications with similar demands.

Although we discussed CPU, memory, and I/O performance separately, it is extremely rare that any of these alone is the determining factor in performance on a real application. Real performance depends to some degree on the behavior of all three major subsystems, with the relative importance of each depending on the demands of a given application.

Thus, many benchmark programs have been developed that exercise different parts of a system to different degrees and in different ways. Some benchmarks are very CPU-intensive, being dominated by the integer (or floating-point) performance of the system processor. Others are very memory-intensive, heavily I/O oriented, or well balanced between different types of operations.

Benchmark: programa de prueba

1. Si consideramos varios sistemas informáticos, ¿cuál es el que tiene mejor rendimiento y cómo podemos comprobarlo?
2. ¿En qué situaciones no es posible llevar a cabo comparaciones entre distintos sistemas?
3. ¿Cuál es el objetivo de usar un programa de prueba (*benchmark*)?

4. ¿Es necesario que el programa de prueba siempre reproduzca aplicaciones existentes?
5. Explique por qué los programas de prueba ejercitan distintas partes de un sistema de distinta manera y en distintos grados.
6. Ejemplifique cómo el programa de prueba puede ejercitar una parte del sistema más que otra.

PARTE B: TRADUCCIÓN

- **Traduzca el siguiente texto.**

A number of benchmarks have proven useful over the years and become classics, *de facto* standards for evaluating the performance of systems intended for certain types of applications. Scientific application performance, for example, is often benchmarked using LINPACK, an adaptation of a Fortran linear algebra package (later translated into C). The LINPACK benchmark solves a large system of simultaneous equations with single or double precision floating-point coefficients set up in large (100 by 100 or 1000 by 1000) matrices. There is also a version called Highly Parallel Linpack, in which the size of the matrices can be adjusted in order to match the problem to the best capabilities of the machine under test.

To benchmark: realizar una comparación