1) Suppose a language only allows pass by reference (Fortran IV) by passing addresses of parameters only.

What are the advantages and disadvantages of this design?

The main advantage of this method is the fast accesses to formal parameters in subprograms. The disadvantages are that recursion is rarely useful when values cannot be passed, and also that a number of problems, such as aliasing, occur with the method.

2) Present one argument for and against providing both static and dynamic local variables in subprograms.

It is rather weak, but one could argue that having both adds complexity to the language without sufficient increase in writability.

3) Speculate on the issue of allowing nested subprograms in programming languages—why are they not allowed in many contemporary languages?

Many contemporary languages do not allow nested subprograms because many designers now believe that there are better ways to organize programs. Also, they think the additional complexity of nested subprograms outweighs their value. Finally, there is the problem of the nested structure of programs deteriorating through continued maintenance, leading to largely unstructured programs in the end, regardless of their initial structure.

4) What are at least two arguments against the use of pass-by-name parameters?

Two arguments against pass-by-name parameters are: First, programs that use pass-by-name parameters can be overly complex and difficult to understand. Second, pass-by-name parameters are far less efficient than other parameter-passing methods.