Diseño de un LP

* Conceptos como valores, tipos de dato, variables, almacenamiento, ligaduras, alcance y abstracción de procedimientos son tan fundamentales que se encuentran en casi todos los lenguajes de programación.
* Cada lenguaje nuevo de programación debe apoyar la abstracción de datos de alguna forma, la falta de esta es una seria deficiencia en un lenguaje actual.
* Un lenguaje nuevo debería admitir unidades genéricas, polimorfismo por inclusión/herencia o paramétrico (conceptos asociados fundamentalmente con lenguajes imperativos, orientados a objetos y funcionales).
* Es importante que el diseñador tenga en claro para que va a ser utilizado el lenguaje.

Simplicidad

* Propiedad del diseño determinada por el número de componentes básicos, la minimalidad de los conceptos y las consistencias de sus estructuras.
* Siempre debe ser un objetivo del diseño del lenguaje.
* Va a medir que esos conceptos requeridos y la integridad de sus instrucciones o de sus estructuras sean lo suficientemente simples como para ser entendidas por el programador.

Expresividad

* Habilidad que posee un lenguaje para reflejar el significado deseado por el programador lo mas naturalmente posible.
* El programa debe poder modelar con naturalidad la solución del problema y su dominio de aplicación.
* La expresividad de un lenguaje se ve fuertemente afectada por la elección de que símbolo va a denotar cada operación.
* Puede entrar en conflicto con la simplicidad.

Seguridad

* Propiedad del diseño que desalienta la posibilidad de escribir programas con errores.
* Las características que aumentan la seguridad cargan al programador con tareas de especificar todas las declaraciones posibles dentro del programa mientras esta escribiendo ese código.
* Puede reducir la expresividad y la flexibilidad del lenguaje.
* Tipos de datos, chequeo de tipos, declaraciones de variables, eliminación del uso y la aritmética de punteros, la recolección de basura son características del lenguaje que **aumentan la seguridad**.
* Uso de variables globales, parámetros pasados por referencia, uso de alias **disminuyen la seguridad.**

Regularidad

* Cualidad del lenguaje que refleja la adecuada combinación de los conceptos que lo caracterizan al diseñar un lenguaje
* Para lograr la máxima potencialidad con un numero determinado de conceptos, el programador debe poder combinar estas características de manera regular sin restricciones innecesarias ni interacciones raras entre conceptos.
* Puede dividirse en tres conceptos diferentes:
  + Generalidad: se logra eliminando casos especiales en la disponibilidad y uso de los constructores; y combinando constructores sumamente relacionados en uno mucho mas general.
  + Ortogonalidad: atributo del lenguaje de ser capaz de combinar varias características que posee el lenguaje entre si en todas las combinaciones posibles de forma tal que todas tengan significado no generando resultados inesperados.
  + Uniformidad: cosas similares deben verse de forma similar y tener significado similar o a la inversa.
* Existen otros conceptos que van a influir en la regularidad:
  + Integridad de los tipos