Rust

● Indicar a que paradigma pertenece el lenguaje y explicar cuáles son las características que lo llevan a ubicarse dentro del mismo. (recuerden que el lenguaje puede tener características de más de un paradigma)

Multi paradigma

* procedimental
  + manejo de memoria de bajo nivel
  + Sistema de ownership
  + keyword “unsafe”
  + borrow checker
    - previene errores concurrentes en el momento de compilar
* orientado a objetos
  + abstracciones de costo cero
    - structs que tienen datos y comportamiento asociado
  + no tiene herencia, tiene traits, que son una forma distinta de herencia.
    - De esta manera implementa polimorfismo ad hoc
      * los traits son por ejemplo Add, Display, etc. Que dependiendo de como esten definidos en cada abstraccion se van a comportar de una manera u otra.
    - Monomorphización
      * una copia del codigo es generada por cada instancia
  + no usa garbage collector
* funcional
  + soporta pattern matching (????)
  + traits son basicamente typeclasses en haskell
  + no optimiza mucho la recursion por la cola.

Tipado estatico y fuerte.

● Criterios de evaluación: indicar si el lenguaje seleccionado favorece a un criterio y perjudica a otro.

Dificulta el aprendizaje al incorporar conceptos nuevos como el “ownership” de una variable y el “borrow checking”. Pero esto mejora enormemente la confiabilidad del lenguaje, permitiendo encontrar y prevenir muchos mas errores en la etapa de compilación.

Borrow checking:

cuando se crea una referencia a una variable entonces esta referencia es solo read only o solo hay una referencia mutable. Esto previene que distintos hilos modifiquen una misma variable.

● ¿Es un lenguaje Fuertemente Tipado? Utiliza vinculación estática o dinámica de tipos.

Si, es un lenguaje fuertemente tipado.

Vinculación:

● Expresiones: ¿el lenguaje permite expresiones mixtas? ¿Utiliza alguna regla de conversión de tipos?

● Tipos de Datos: ¿qué tipos de datos utiliza? ¿Permite tipos de datos definidos por el usuario?

Permite tipos de datos definidos por el usuario.

● ¿El lenguaje tiene algún nivel de polimorfismo?

Permite el uso de generics, variables que comparten un tipo pero no se sabe cual.

● ¿El lenguaje provee estructuras de control? ¿De qué tipo?

● ¿Qué métodos de pasaje de parámetros utiliza? ¿Permite que las unidades sean pasadas como parámetros?

● Excepciones: ¿Cómo trata los eventos inusuales?¿Provee algún manejador?

● ¿Permite programación concurrente?

Si

Los puntos anteriores son a modo de guía, para hacer una investigación profunda sobre el lenguaje es necesario adaptar las mismas a este. Deberán contestar en su investigación

¿porque consideran que el lenguaje fue creado?

* Para tener un sucesor a los lenguajes de bajo nivel que gestionan memoria, pero de una manera mas segura y manteniendo un buen rendimiento.

¿Cuál fue la necesidad de su creación?

Luego de realizar la investigación, deberán realizar una presentación de la misma en la que se evidencien los puntos claves investigados, se muestren ejemplos de código funcionando en los que se observen estas características y entregar la documentación correspondiente.