

Comenzado el	jueves, 1 de febrero de 2024, 13:12
Estado	Finalizado
Finalizado en	jueves, 1 de febrero de 2024, 13:18
Tiempo empleado	6 minutos 6 segundos
Puntos	16,00/16,00
Calificación	50,00 de 50,00 (100%)

PREGUNTA 1

Correcta

Se puntúa 13,00  
sobre 13,00

En su curso de Arquitectura Empresarial usted aprendió que los nombres eran una importante herramienta de organización y comunicación en los sistemas. Igualmente, aprendió que un API simple para modelar los sistemas de nombres era:

value  $\leftarrow$  resolve(name, context), retorna el valor asignado al nombre en el contexto

value  $\leftarrow$  BIND (name, value, context), en el contexto asigna el valor al nombre.

status  $\leftarrow$  UNBIND (name, context), en el contexto borra el nombre y el enlace que tenía

list  $\leftarrow$  ENUMERATE (context), Lista los nombres disponibles en el contexto

result  $\leftarrow$  compare(name1, name2), responde verdadero o falso si los nombres son "iguales", pero la definición de iguales varía dependiendo del sistema.

A usted le han pedido crear un sistema de caché, que es una memoria temporal y rápida, que permite por ejemplo que un sistema acceda a datos que usa frecuentemente sin necesidad de ir hasta sistemas de persistencia más demorados y complejos como por ejemplo una base de datos.

Usted decidió crear un sistema basado en una estructura de datos en forma de árbol. Su diseño contempla una estructura de datos que almacene objetos y que permita navegar hasta los objetos usando nombres completamente calificados. Un nombre completamente calificado contiene el contexto y el nombre del objeto buscado. Por ejemplo si la estructura de datos le piden retornar el objeto con el nombre "/a/b/daniel/maria", el sistema sabe que debe buscar el objeto "maria" en el contexto "/a/b/daniel/". Su sistema implementa el API descrito arriba.

Su estructura de datos almacena referencias a los objetos.

Considere los siguientes llamados a su estructura de datos, con Objeto1 y Objeto2 representando objetos diferentes:

BIND ("daniel", Objeto1, "/")

BIND ("pedro", Objeto2, "/")

Objeto3 = Objeto2 (Copia de referencias)

BIND ("a", Objeto3, "/daniel/c/")

Si la función "compare" retorna verdadero si los nombres son iguales, es decir el nombre y el contexto son iguales:

- cuál es el resultado de compare("/daniel", "/daniel") Verdadero ✓
- cuál es el resultado de compare("/daniel", "/juana") Falso ✓
- cuál es el resultado de compare("/daniel", "/pedro") Falso ✓
- cuál es el resultado de compare("/pedro", "/daniel/c/a") Falso ✓

Si la función "compare" retorna verdadero si los objetos representados por el nombre son iguales:

- cuál es el resultado de compare("/daniel", "/pedro") Falso ✓
- cuál es el resultado de compare("/daniel", "/daniel/c") Falso ✓
- cuál es el resultado de compare("/daniel/c", "/pedro") Falso ✓
- cuál es el resultado de compare("/daniel/c/a", "/pedro") Verdadero ✓

Ahora considere:

1. que el sistema almacena el contexto del valor asociado con el último nombre que retornó en un llamado resolve.
2. El contexto no cambia si resolve no encuentra el nombre, en este caso retorna nulo pero sin cambiar el contexto almacenado.
3. Así, si recibe un llamado a la función resolve pero el nombre no tiene información del contexto, entonces usa el contexto almacenado como el contexto de búsqueda por defecto.
4. El sistema inicia con el contexto por defecto igual a "/".
5. La función "compare" retorna verdadero si los objetos representados por el nombre son iguales:

Considerando lo anterior si las siguientes instrucciones se ejecutaran en secuencia, qué valores retornarían cada una de las sentencias:

- resolve("pedro") Objeto2 ✓
- resolve("daniel/c/a") Objeto2 ✓
- resolve("pedro") nulo ✓

resolve("a")

Objeto2



compare("/daniel/c/a", "/pedro")

Verdadero



## PREGUNTA 2

Correcta

Se puntúa 1,00  
sobre 1,00

Un interprete se puede modelar con:

Seleccione una:

- ☐ a. execute(Next); islnRepertoire(Next); get(Next)
- ☐ b. execute(); get(value); set (value)
- ☒ c. Una referencia a la siguiente instrucción; Un repertorio de acciones; Una referencia a un entorno ✓
- ☐ d. call(Next); get(Name); set(Value)

Respuesta correcta

La respuesta correcta es: Una referencia a la siguiente instrucción; Un repertorio de acciones; Una referencia a un entorno

PREGUNTA **3**

Correcta

Se puntúa 1,00  
sobre 1,00

Los enlaces de comunicación se pueden modelar con las siguientes operaciones (API)

Seleccione una:

- ☒ a. `send (nombre_Del_Enlace, espacio_memoria_salida); receive (nombre_Del_Enlace, espacio_memoria_entrada)` ✓
- ☐ b. `write(name, value); value <—— read (name)`
- ☐ c. `sendMessage(Message); receiveMessage(Message)`

Respuesta correcta

La respuesta correcta es: `send (nombre_Del_Enlace, espacio_memoria_salida); receive (nombre_Del_Enlace, espacio_memoria_entrada)`

PREGUNTA **4**

Correcta

Se puntúa 1,00  
sobre 1,00

La abstracción de memoria se puede modelar con las siguientes dos operaciones.

Seleccione una:

- ☐ a. `send (nombre_Del_Enlace, espacio_memoria_salida); receive (nombre_Del_Enlace, espacio_memoria_entrada)`
- ☐ b. `get(value); set(value)`
- ☐ c. `write(name, value) ; value <—— send (name)`
- ☒ d. `write(name, value) ; value <—— read (name)` ✓

Respuesta correcta

La respuesta correcta es: `write(name, value) ; value <—— read (name)`