

## Diseño del TDA

### TDA Jugador

- Nombre: Jugador
- Invariantes:
  - nombre de tipo cadena
  - vidas  $\in \mathbb{N}$
- Operaciones:
  - Jugador : entero x cadena
  - deseaCrearUsuario :  $\rightarrow$  booleano
  - crearUsuario :
  - pedirNombre :  $\rightarrow$  cadena
  - pedirVidas :  $\rightarrow$  entero
  - obtenerVidas :  $\rightarrow$  entero
  - quitarVidas : entero
  - asignarNombre : cadena
  - asignarVida: entero

### Detalle de las operaciones

- Operación Jugador
  - Descripción: construye un jugador en base a los dos parámetros recibidos
  - Pre condicion: el parametro entero debe ser mayor a 3, el parámetro cadena tiene que ser distinto de vacío
  - Post condicion: construye un jugador con los parámetros recibidos. En el caso de no haberlos, el entero será igual a 3 y la cadena a “Juan/a Perez”
- Operación deseaCrearUsuario
  - Descripción: le pregunta al usuario si desea crearse un perfil como jugador
  - Pre condicion: -
  - Post condicion: devuelve verdadero si el usuario desea crearse un perfil
- Operación crearUsuario
  - Descripción: crea un usuario llamando a los métodos pedirNombre, pedirVidas, asignarNombre y asignarVidas
  - Pre condicion: -
  - Post condicion: crea un usuario
- Operación pedirNombre
  - Descripción: le pide al usuario que ingrese un nombre
  - Pre condicion: -
  - Post condicion: devuelve la cadena ingresada por el usuario
- Operación pedirVidas

- Descripción: le pide al usuario que ingrese una cantidad de vidas
  - Pre condición: -
  - Post condición: devuelve el entero ingresado por el usuario
- Operación obtenerVidas
  - Descripción: devuelve la cantidad de vidas que tiene el jugador
  - Pre condición: -
  - Post condición: devuelve la cantidad de vidas que tiene el jugador
- Operación quitarVidas
  - Descripción: le saca entero cantidad de vidas al jugador
  - Pre condición: el entero debe ser mayor que cero
  - Post condición:
- Operación asignarNombre
  - Descripción: le asigna la cadena al nombre del jugador
  - Pre condición: la cadena debe estar compuesta por letras unicamente
  - Post condición: -
- Operación asignarVida
  - Descripción: le asigna la cantidad de vidas recibidas al jugador
  - Pre condición: el entero debe ser mayor a cero
  - Post condición: -

## TDA CharDinamico

- Nombre: CharDinamico
- Invariantes:
  - tamaño  $\in \mathbb{N}$
  - palabra de tipo puntero a carácter
- Operaciones:
  - CharDinamico : puntero a cadena x entero
  - CharDinamico : direccion de memoria de cadena
  - ~CharDinamico :
  - obtenerElemento : entero  $\rightarrow$  carácter
  - obtenerTamanio :  $\rightarrow$  entero
  - asignarPalabra : cadena
  - mostrarCaracteres :
  - insertarCaracter : caracter x entero
  - redimensionar : entero
  - checkCaracterEnPalabra : caracter  $\rightarrow$  booleano
  - asignarNuloAlVector : entero x entero
  - copiarDatos : puntero a cadena x entero x entero

## Detalle de las operaciones

- Operación CharDinamico
  - Descripcion: crea un vector de tipo caracter con la cadena recibida de tamaño entero y muestra por pantalla en qué lugar se reservó la memoria
  - Pre condicion: el entero tiene que ser mayor a uno y la cadena distinta de vacío
  - Post condicion: crea un vector dinámico de caracteres
- Operación CharDinamico
  - Descripcion: es el constructor de copia, se encarga de copiar los datos del vector a uno nuevo
  - Pre condición: la dirección de memoria de la cadena debe ser válida
  - Post condicion: -
- Operación ~CharDinamico
  - Descripcion: libera la memoria previamente reservada
  - Pre condicion: -
  - Post condicion: -
- Operación obtenerTamanio
  - Descripcion: devuelve un entero que corresponde al tamaño del vector dinámico
  - Pre condicion: -
  - Post condicion: devuelve un entero que corresponde al tamaño del vector dinámico
- Operación obtenerElemento
  - Descripcion: recibe una posición y devuelve el elemento que se encuentra allí

- Pre condición: el entero es mayor a 0 y menor al tamaño del vector
  - Post condición: devuelve el carácter que se encuentra en la posición entero del vector
- Operación asignarPalabra
  - Descripción: recorre la cadena recibida y asigna cada carácter a la cadena al atributo palabra en la posición correspondiente
  - Pre condición: la cadena debe incluir únicamente letras de la a A a la Z exceptuando la Ñ
  - Post condición: -
- Operación mostrarCaracteres
  - Descripción: recorre la cadena y muestra todos los caracteres del atributo palabra con un espacio de por medio
  - Pre condición: -
  - Post condición: muestra por pantalla todos los caracteres del atributo palabra con un espacio de por medio
- Operación insertarCaracter
  - Descripción: inserta el carácter en la posición entero de la cadena
  - Pre condición: el carácter debe ser una letra de la A a la Z exceptuando la Ñ, el entero debe ser mayor a cero y menor al tamaño de la palabra
  - Post condición: -
- Operación redimensionar
  - Descripción: si el tamaño recibido es distinto al atributo tamaño, inicializa un puntero auxiliar que apunte a la misma dirección que el atributo palabra, en palabra se reserva un nuevo espacio de memoria del tamaño del entero recibido, copia los datos originales (a los que se accede mediante el puntero auxiliar) al nuevo espacio de memoria, libera la memoria apuntada por auxiliar, si el tamaño recibido es mayor al atributo tamaño asigna nulos (‘ ’) a todas las posiciones desde tamaño + 1 hasta el nuevo tamaño. Finalmente se iguala el atributo tamaño a tamaño recibido
  - Pre condición: el entero es mayor a cero
  - Post condición: -
- Operación checkCaracterEnPalabra
  - Descripción: recorre las posiciones a las que apunta el puntero palabra, si el carácter está en la cadena devuelve un booleano verdadero, de lo contrario devuelve uno falso
  - Pre condición: el carácter debe ser una letra de la A a Z exceptuando la Ñ
  - Post condición: devuelve un booleano si el carácter está en la cadena
- Operación asignarNuloAlVector
  - Descripción: si el atributo tamaño es mayor a cero, recorre las posiciones a las que apunta el puntero palabra desde el primer entero hasta el segundo y le asigna un nulo (‘ ’)
  - Pre condición: el primer entero es mayor a cero y menor al segundo, el segundo entero es menor o igual al tamaño de la palabra
  - Post condición: -
- Operación copiarDatos

- Descripción: si el atributo tamaño es mayor a cero, recorre las posiciones a las que apunta el puntero palabra desde el primer entero hasta el segundo, y le asigna al atributo palabra el carácter que se encuentra en la misma posición de la cadena recibida
- Pre condición: el puntero a caracteres apunta a una dirección de memoria que tiene por lo menos un carácter. El primer entero es mayor a cero y menor al segundo, el segundo entero es menor o igual al tamaño de la palabra
- Post condición: -

## TDA Ahorcado

- Nombre: Ahorcado
- Invariantes:
  - intentosFallidos  $\in \mathbb{N}$
  - estadoJuego  $\in \mathbb{N}$
  - palabraAAdivinar de tipo CharDinamico
  - palabraSecreta de tipo CharDinamico
  - jugador de tipo Jugador
- Operaciones:
  - Ahorcado : Jugador x cadena x entero
  - nuevoJuego :
  - deseaJugarDeNuevo :  $\rightarrow$  booleano
  - mostrarMensajeGanoOPerdio :
  - mostrarDespedida :
  - arriesgar : caracter
  - arriesgar : cadena
  - actualizarAhorcado :
  - mostrarDibujoAhorcado :

## Detalle de las operaciones

- Operación Ahorcado
  - Descripción: crea un ahorcado inicializando: el estadoJuego en EMPEZO\_JUEGO (constante de tipo entero igual a 0), los intentos fallidos en 0, el jugador en jugadorOut recibido, la palabraAAdivinar con la cadena y el entero recibidos, además redimensiona la palabraSecreta al tamaño entero recibido y le asigna un guion bajo ('\_') a cada posición de la palabraSecreta
  - Pre condición: jugadorOut debe ser un Jugador valido, el entero debe ser mayor a cero, la cadena debe ser distinta de vacio
  - Post condición: crea un ahorcado
- Operación nuevoJuego
  - Descripción: comienza un nuevo juego
  - Pre condición: -
  - Post condición: -
- Operación deseaJugarDeNuevo
  - Descripción: le pregunta al usuario si desea jugar de nuevo y devuelve un booleano verdadero si el carácter ingresado fue 's' o 'S'
  - Pre condición: -
  - Post condición: devuelve un booleano verdadero si el jugador desea volver a jugar, de lo contrario falso
- Operación mostrarMensajeGanoOPerdio

- Descripción: muestra un mensaje predeterminado de acuerdo a si el estado del juego es gano o perdió
  - Pre condición: el estadoJuego debe ser igual a GANO\_JUEGO o PERDIO\_JUEGO
  - Post condición: muestra por pantalla el mensaje correspondiente al estadoJuego
- Operación mostrarDespedida
  - Descripción: muestra por pantalla un mensaje de despedida con el nombre del jugador
  - Pre condición: -
  - Post condición: -
- Operación arriesgar
  - Descripción: permite que el usuario arriesgue un carácter. Si el carácter pertenece a la palabra, recorre la palabraAAdivinar hasta que se encuentra la primera posición en la que aparece dicho carácter, y lo inserta en la palabraSecreta. Si la palabraSecreta es igual a la palabraAAdivinar, modifica el estado del juego a GANO\_JUEGO (constante de tipo entero cuyo valor es 1). Si el carácter no se encuentra en la palabraAAdivinar incrementa en uno los intentosFallidos y le quita 1 vida al jugador. Finalmente llama al método actualizarAhorcado para mostrar los cambios por pantalla
  - Pre condición: el carácter debe ser una letra de la A a la Z exceptuando la Ñ
  - Post condición: -
- Operación arriesgar
  - Descripción: permite al usuario que arriesgue una cadena. Si es igual a la palabraAAdivinar le asigna a palabraSecreta dicha cadena, y modifica el estado del juego a GANO\_JUEGO. De lo contrario incrementa en 2 los intentos fallidos y le quita 2 vidas al jugador. Finalmente llama al método actualizarAhorcado para mostrar los cambios por pantalla
  - Pre condición: la cadena debe estar compuesta unicamente por letras de la A a la Z exceptuando la Ñ
  - Post condición: -
- Operación actualizarAhorcado
  - Descripción: llama al método mostrarDibujoAhorcado y muestra los caracteres de palabraSecreta
  - Pre condición: -
  - Post condición: muestra por pantalla el dibujo del ahorcado con la palabraSecreta
- Operación mostrarDibujoAhorcado
  - Descripción: dependiendo de la cantidad de intentosFallidos, va mostrando las distintas etapas del ahorcado
  - Pre condición: -
  - Post condición: muestra por pantalla el dibujo del ahorcado

