

# Verificación de software

Verificación de un sistema bla bla

Autor:

Juan Ignacio Farizano

Trabajo Práctico Final - Ingeniería de Software II

Departamento de Ciencias de la Computación

Facultad de Ciencias Exactas, Ingeniería y Agrimensura

Universidad Nacional de Rosario

Rosario, Santa Fe, Argentina

11 de febrero de 2025

## El problema

Queremos especificar el funcionamiento del sistema *Mis Estudiantes* que utilizan directivos de escuelas primarias y secundarias para inscribir y reinscribir alumnos (por ej. promoción de grado, repitente, etc) en la base de datos del Ministerio de Educación de la provincia de Buenos Aires, lo descrito es similar al sistema existente pero simplificado para mantener una dificultad razonable para este trabajo práctico.

Un directivo desea inscribir o reinscribir a un alumno en su escuela. Cada alumno cuenta con un legajo compuesto por entradas donde cada una representa un paso del alumno por el sistema educativo; cada registro describe el grado en que el alumno fue inscripto y tiene tres estados posibles:

- **Inscripto:** el alumno fue inscripto y es habilitado a cursar el grado registrado
- **Promueve:** el alumno cumplió los requisitos para promocionar el grado al cual se encuentra inscripto y es habilitado a ser inscripto en el grado siguiente o a graduarse si se encontraba en el 12avo grado
- **Repite:** el alumno no cumplió los requisitos para promocionar de grado y debe ser inscripto en el mismo grado

Además se desea llevar un registro fácilmente accesible de los alumnos graduados, es decir alumnos que hayan promocionado 12avo grado.

Se deben especificar las siguientes operaciones:

1. **Inscribir alumno:** se inscribe por primera vez a un alumno a primer grado
2. **Reinscribir alumno:** se inscribe un alumno pre-existente al grado que corresponda según su último estado de inscripción
3. **Cerrar inscripción:** la inscripción actual es cerrada correspondientemente a si cumplió o no los requisitos para promocionar de grado
4. **Consultar repitencia:** se desea consultar si un alumno repitió el último grado que haya cursado

Se asume que se registrará el historial completo del alumno, empezando siempre desde primer grado hasta que finalice. Para simplificar no diferenciamos entre primaria y secundaria, registramos desde 1er grado hasta 12avo, el último del secundario. Los requisitos para promocionar de grado se encuentran por fuera del sistema y no deben ser tenidos en cuenta. Se quieren saber todos los estados que fueron registrados en cada legajo, por lo tanto cada uno es inmutable, si un alumno repite se produce un nuevo

registro con estado **Inscripto** en el mismo grado y si promociona se registra nuevamente con el estado **Promueve**.

**Nota:** Esto quizás es demasiado para el enunciado, mucho puede moverse a los comentarios entre los esquemas más abajo

## Designaciones

## Especificación en Z

$[ALUMNO]$

$GRADO == \mathbb{N}$

$ESTADO ::= inscripto \mid promueve \mid repite$

$REP ::= alumnoEsRepitente \mid alumnoNoEsRepitente \mid alumnoNoEncontrado$

$MisEstudiantes$

$legajos : ALUMNO \rightarrow \text{seq}(GRADO \times ESTADO)$

$graduados : \mathbb{P} ALUMNO$

$GraduadoTieneLegajoInv$

$MisEstudiantes$

$graduados \subseteq \text{dom } legajos$

$GraduadosRequisitosInv$

$MisEstudiantes$

$\forall a : \text{dom } legajos \bullet (a \in graduados) \Leftrightarrow ((\text{last } (legajos \ a)).1 = 12 \wedge (\text{last } (legajos \ a)).2 = promueve)$

*MisEstudiantesInicial*

*MisEstudiantes*

$legajos = \emptyset$

$graduados = \emptyset$

*InscribirAlumnoOk*

$\Delta$ *MisEstudiantes*

$alumno? : ALUMNO$

$alumno? \notin \text{dom } legajos$

$legajos' = legajos \cup \{alumno? \mapsto \langle (1, inscripto) \rangle\}$

$graduados' = graduados$

*InscribirAlumnoRegistradoE*

$\Xi$ *MisEstudiantes*

$alumno? : ALUMNO$

$alumno? \in \text{dom } legajos$

$InscribirAlumno == InscribirAlumnoOk \vee InscribirAlumnoRegistradoE$

*ReinscribirAlumnoPromovidoOk*

$\Delta$ *MisEstudiantes*

$alumno? : ALUMNO$

$alumno? \in \text{dom } legajos$

$(\text{last } (legajos \text{ alumno?})).1 < 12$

$(\text{last } (legajos \text{ alumno?})).2 = \text{promueve}$

$legajos' = legajos \oplus \{alumno? \mapsto legajos \text{ alumno?} \wedge \langle ((\text{last } (legajos \text{ alumno?})).1 + 1, inscripto) \rangle\}$

$graduados' = graduados$

*ReinscribirAlumnoRepitenteOk*

$\Delta$  *MisEstudiantes*

*alumno?* : *ALUMNO*

*alumno?*  $\in$  dom *legajos*

$(last\ (legajos\ alumno?)).1 \leq 12$

$(last\ (legajos\ alumno?)).2 = repite$

$legajos' = legajos \oplus \{alumno? \mapsto legajos\ alumno? \frown \langle ((last\ (legajos\ alumno?)).1, inscripto) \rangle\}$

$graduados' = graduados$

*ReinscribirAlumnoNoEncontradoE*

$\Xi$  *MisEstudiantes*

*alumno?* : *ALUMNO*

*alumno?*  $\notin$  dom *legajos*

*ReinscribirAlumnoGraduadoE*

$\Xi$  *MisEstudiantes*

*alumno?* : *ALUMNO*

*alumno?*  $\in$  dom *legajos*

$(last\ (legajos\ alumno?)).1 = 12$

$(last\ (legajos\ alumno?)).2 = promueve$

*ReinscribirAlumnoDobleInscripE*

$\Xi$  *MisEstudiantes*

*alumno?* : *ALUMNO*

*alumno?*  $\in$  dom *legajos*

$(last\ (legajos\ alumno?)).2 = inscripto$

$ReinscribirAlumnoE == ReinscribirAlumnoGraduadoE \vee ReinscribirAlumnoDobleInscripE$   
 $\vee ReinscribirAlumnoNoEncontradoE$   
 $ReinscribirAlumnoOk == ReinscribirAlumnoPromovidoOk \vee ReinscribirAlumnoRepitenteOk$   
 $ReinscribirAlumno == ReinscribirAlumnoOk \vee ReinscribirAlumnoE$

*CerrarInscripcionNoGraduadoOk*

$\Delta MisEstudiantes$

$alumno? : ALUMNO$

$estado? : ESTADO$

$alumno? \in \text{dom } legajos$

$estado? = repite \vee (estado? = promueve \wedge (last(legajos\ alumno?)).1 < 12)$

$(last(legajos\ alumno?)).2 = inscripto$

$legajos' = legajos \oplus \{alumno? \mapsto legajos\ alumno? \wedge \langle ((last(legajos\ alumno?)).1, estado?) \rangle\}$

$graduados' = graduados$

*CerrarInscripcionGraduadoOk*

$\Delta MisEstudiantes$

$alumno? : ALUMNO$

$estado? : ESTADO$

$alumno? \in \text{dom } legajos$

$estado? = promueve$

$(last(legajos\ alumno?)).1 = 12$

$(last(legajos\ alumno?)).2 = inscripto$

$legajos' = legajos \oplus \{alumno? \mapsto legajos\ alumno? \wedge \langle ((last(legajos\ alumno?)).1, estado?) \rangle\}$

$graduados' = graduados \cup \{alumno?\}$

*CerrarInscripcionEstadoInvalidoE*

$\Xi MisEstudiantes$

$estado? : ESTADO$

$estado? = inscripto$

*CerrarInscripcionAlumnoNoEncontradoE* \_\_\_\_\_

$\Xi$  *MisEstudiantes*

*alumno?* : *ALUMNO*

*alumno?*  $\notin$  dom *legajos*

*CerrarInscripcionE* == *CerrarInscripcionEstadoInvalidoE*  $\vee$  *CerrarInscripcionAlumnoNoEncontradoE*

*CerrarInscripcionOk* == *CerrarInscripcionNoGraduadoOk*  $\vee$  *CerrarInscripcionGraduadoOk*

*CerrarInscripcion* == *CerrarInscripcionOk*  $\vee$  *CerrarInscripcionE*

*AlumnoEsRepitenteSiOk* \_\_\_\_\_

$\Xi$  *MisEstudiantes*

*alumno?* : *ALUMNO*

*rep!* : *REP*

*alumno?*  $\in$  dom *legajos*

(last (legajos *alumno?*)).2 = *repite*

*rep!* = *alumnoEsRepitente*

*AlumnoEsRepitenteNoOk* \_\_\_\_\_

$\Xi$  *MisEstudiantes*

*alumno?* : *ALUMNO*

*rep!* : *REP*

*alumno?*  $\in$  dom *legajos*

(last (legajos *alumno?*)).2  $\neq$  *repite*

*rep!* = *alumnoNoEsRepitente*

*AlumnoEsRepitenteNoEncontradoE*

$\exists \text{MisEstudiantes}$

*alumno?* : *ALUMNO*

*rep!* : *REP*

*alumno?*  $\notin \text{dom legajos}$

*rep!* = *alumnoNoEncontrado*

*AlumnoEsRepitenteE* == *AlumnoEsRepitenteNoEncontradoE*

*AlumnoEsRepitenteOk* == *AlumnoEsRepitenteSiOk*  $\vee$  *AlumnoEsRepitenteNoOk*

*AlumnoEsRepitente* == *AlumnoEsRepitenteOk*  $\vee$  *AlumnoEsRepitenteE*