

# 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

28 de enero de 2025

## El problema

Queremos especificar el funcionamiento de un sistema 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, lo descrito es similar a un 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 donde cada entrada 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

Se deben especificar las siguientes operaciones:

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

El legajo de un alumno será compartido globalmente entre todas las escuelas que utilicen el sistema, de esta forma se registra el historial completo de un alumno desde el primer grado hasta el último. 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 pasaron las inscripciones, por lo tanto cada estado de cada inscripción 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

$[ALUMNO]$

$GRADO == \mathbb{N}$

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

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

*Escuela*

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

*EscuelaInicial*

*Escuela*

$legajos = \emptyset$

*InscribirAlumnoNuevoOk*

$\Delta Escuela$

$a? : ALUMNO$

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

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

*InscribirAlumnoPromovidoOk*

$\Delta Escuela$

$a? : ALUMNO$

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

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

$1 \leq (last(legajos\ a?)).1 < 12$

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

---

*InscribirAlumnoRepitenteOk*

---

$\Delta Escuela$

$a? : ALUMNO$

---

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

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

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

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

---



---

*InscribirAlumnoGraduadoE*

---

$\Xi Escuela$

$a? : ALUMNO$

---

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

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

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

---



---

*InscribirAlumnoDobleInscripE*

---

$\Xi Escuela$

$a? : ALUMNO$

---

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

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

---

$InscribirAlumnoE == InscribirAlumnoGraduadoE \vee InscribirAlumnoDobleInscripE$

$InscribirAlumnoOk == InscribirAlumnoNuevoOk \vee InscribirAlumnoPromovidoOk$

$\vee InscribirAlumnoRepitenteOk$

$InscribirAlumno == InscribirAlumnoOk \vee InscribirAlumnoE$

---

*CerrarInscripcionOk*

---

$\Delta Escuela$

$a? : ALUMNO$

$e? : ESTADO$

---

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

$e? = \text{promueve} \vee e? = \text{repite}$

$(\text{last } (legajos \ a?)).2 = \text{inscripto}$

$legajos' = legajos \oplus \{a? \mapsto legajos \ a? \frown \langle ((\text{last } (legajos \ a?)).1, e?) \rangle\}$

---



---

*CerrarInscripcionEstadoInvalidoE*

---

$\Xi Escuela$

$e? : ESTADO$

---

$e? = \text{inscripto}$

---



---

*CerrarInscripcionAlumnoNoEncontradoE*

---

$\Xi Escuela$

$a? : ALUMNO$

---

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

---

$CerrarInscripcionE == CerrarInscripcionEstadoInvalidoE \vee CerrarInscripcionAlumnoNoEncontradoE$

$CerrarInscripcion == CerrarInscripcionOk \vee CerrarInscripcionE$

---

*AlumnoEsRepitenteSiOk*

---

$\Xi Escuela$

$rep! : REP$

---

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

$(\text{last } (legajos \ a?)).2 = \text{repite}$

$rep! = \text{alumnoEsRepitente}$

---

---

*AlumnoEsRepitenteNoOk*

$\exists Escuela$

$rep! : REP$

---

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

$(last(legajos\ a?)).2 \neq repite$

$rep! = alumnoNoEsRepitente$

---



---

*AlumnoEsRepitenteNoEncontradoE*

$\exists Escuela$

$rep! : REP$

---

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

$rep! = alumnoNoEncontrado$

---

$AlumnoEsRepitenteE == AlumnoEsRepitenteNoEncontradoE$

$AlumnoEsRepitenteOk == AlumnoEsRepitenteSiOk \vee AlumnoEsRepitenteNoOk$

$AlumnoEsRepitente == AlumnoEsRepitenteOk \vee AlumnoEsRepitenteE$