

```

\begin{zed}
  [ ALUNOS, CURSOS, INSTRUTORES, LIVROS ] \\
  MSG ::= MatriculaUmAlunoOK | MatriculaUmAlunoNOK | InstrutorSubstituidoOK | RelatorioOK |
  AlunoTrocadoOK
\end{zed}

```

```

\begin{schema}{ControleDeCursos}
  cTodosCursos, cCursosAvancados, cCursosBasicos: \power CURSOS \\
  rAlunosCursos: ALUNOS \rel CURSOS \\
  rCursoInstrutor: CURSOS \fun INSTRUTORES \\
  rCursosLivros: CURSOS \rel LIVROS
\where
  cCursosAvancados \cap cCursosBasicos = \emptyset \\
  cTodosCursos = cCursosAvancados \cup cCursosBasicos \\
  \forall c: CURSOS @ \# (rAlunosCursos \rres \{c\}) \geq 10 \\
  \forall c: CURSOS @ \# (\{c\} \dres rCursosLivros) \geq 5
\end{schema}

```

1. Matricular um aluno em um curso com sucesso

```

\begin{schema}{MatriculaAlunoCursoSucesso}
  \Delta ControleDeCursos \\
  alunoMatricula?: ALUNOS \\
  cursoMatricula?: CURSOS \\
  resp!: MSG
\where
  (alunoMatricula?, cursoMatricula?) \notin rAlunosCursos \\
  cursoMatricula? \in cTodosCursos \\
  rAlunosCursos' = rAlunosCursos \cup \{(alunoMatricula?, cursoMatricula?)\} \\
  resp! = MatriculaUmAlunoOK
\end{schema}

```

2. Matricular um aluno em um curso sem sucesso

```

\begin{schema}{MatriculaAlunoCursoSemSucesso}
  \Delta ControleDeCursos \\
  alunoMatricula?: ALUNOS \\
  cursoMatricula?: CURSOS \\
  resp!: MSG
\where
  (alunoMatricula?, cursoMatricula?) \in rAlunosCursos \lor
  cursoMatricula? \notin cTodosCursos \\
  resp! = MatriculaUmAlunoNOK
\end{schema}

```

3. Substituir um instrutor com sucesso

```

\begin{schema}{SubstituirInstrutorSucesso}
  \Delta ControleDeCursos \\
  instrutorEntrar?, instrutorSair?: INSTRUTORES \\
  resp!: MSG
\where
  rCursoInstrutor' = rCursoInstrutor \nrres \{instrutorSair?\} \cup

```

```

(\dom (rCursoInstrutor \rres \{instrutorSair?\}) \cross \{instrutorEntrar?\})\\
resp! = InstrutorSubstituidoOK
\end{schema}

```

4. Listar os alunos que tem materias com tres disciplinas em comum

```

\begin{schema}{AlunosCursosTresDisciplinas}
\Xi ControleDeCursos \\
saida!: \power ALUNOS \\
resp!: MSG
\where
saida! = \{ c1, c2: CURSOS; a: ALUNOS | (c1 \neq c2) \land
(\# (\ran (\{c1\} \dres rCursosLivros) \cap \ran (\{c2\} \dres rCursosLivros) ) > 3) \land
((a, c1) \in rAlunosCursos) @ a \} \\
resp! = RelatorioOK
\end{schema}

```

5. Trocar um aluno de curso

```

\begin{schema}{TrocarAlunoCurso}
\Delta ControleDeCursos \\
alunoTroca?: ALUNOS \\
cursoEntrar?, cursoSair?: CURSOS \\
resp!: MSG
\where
\# (\{alunoTroca?\} \dres rAlunosCursos) > 0 \\
rAlunosCursos' = rAlunosCursos \cup \{(alunoTroca?, cursoEntrar?)\} \setminus
\{(alunoTroca?, cursoSair?)\} \\
resp! = AlunoTrocadoOK
\end{schema}

```