## Realizar Inscrições

iniciarRealizacaoInscricao(matriculaAluno)

solicitarInscricoes(listaDisciplinas)

confirmarInscricoes()

## Atender Listas de Espera

* selecionarListaEspera(nomeDisciplina)
* montarGrade(código, dias, intervalos)
* montarTurma(matriculaProfessor, salas, quantidadeAlunos)
  + (pós) um objeto Turma foi criado
  + (pós) uma associação foi criada entre o professor e a turma recém-criada
  + (pós) as associações entre os alunos e a lista selecionada foram removidas.
  + (pós) Foram criadas associações entre os alunos e a turma recém-criada.
  + (pós) Foi criado um objeto Inscricao para cada aluno retirado da lista de espera.
* enviarInscricoes(listaDisciplinas)

## Definir Habilitações

* obterListaHabilitacoes(matriculaProfessor)
  + (pré) matrícula informada deve ser válida.
  + (pós) N/A.
* obterListaDisciplinasDisponiveis(matriculaProfessor)
  + (pré) existem disciplinas cadastradas.
  + (pré) matrícula é válida.
  + (pós) N/A.
* adicionarHabilitacao(nomeDisciplina)
* removerHabilitacao(nomeDisciplina)
  + (pré) existe a disciplina na lista de origem.
  + (pós) lista de origem não pode conter o item transferido.
  + (pós) A lista de destino deve conter o item transferido.
  + (pós) as listas de origem e destino devem permanecer diferentes.
* confirmarHabilitacao()
  + (pós) lista de habilitações do professor foi atualizada.

## Lançar avaliações

* listarTurmas()
* listarAlunos()
* inserirNotasFaltas(nota1, nota2, quantidadeFaltas)
  + (pós) notas e faltas lançadas na inscrição do aluno.
* avaliarAluno()
  + aluno avaliado na inscrição
* exibirAvaliacao()

## Atender Listas de Espera

* obterListaEspera()
* obterDetalheListaEspera(listaEspera)
* obterDisponibilidadeTurma(dia, horarioInicial, horarioFinal)
* alocarAlunoTurma(codigoTurma, listaAlunos)
  + alunos selecionados saem da lista de espera e entram na turma selecionada
* faturarInscricoes(listaAlunos)
  + sistema fatura inscrições dos alunos
* validarQuantidadeAlunos()