

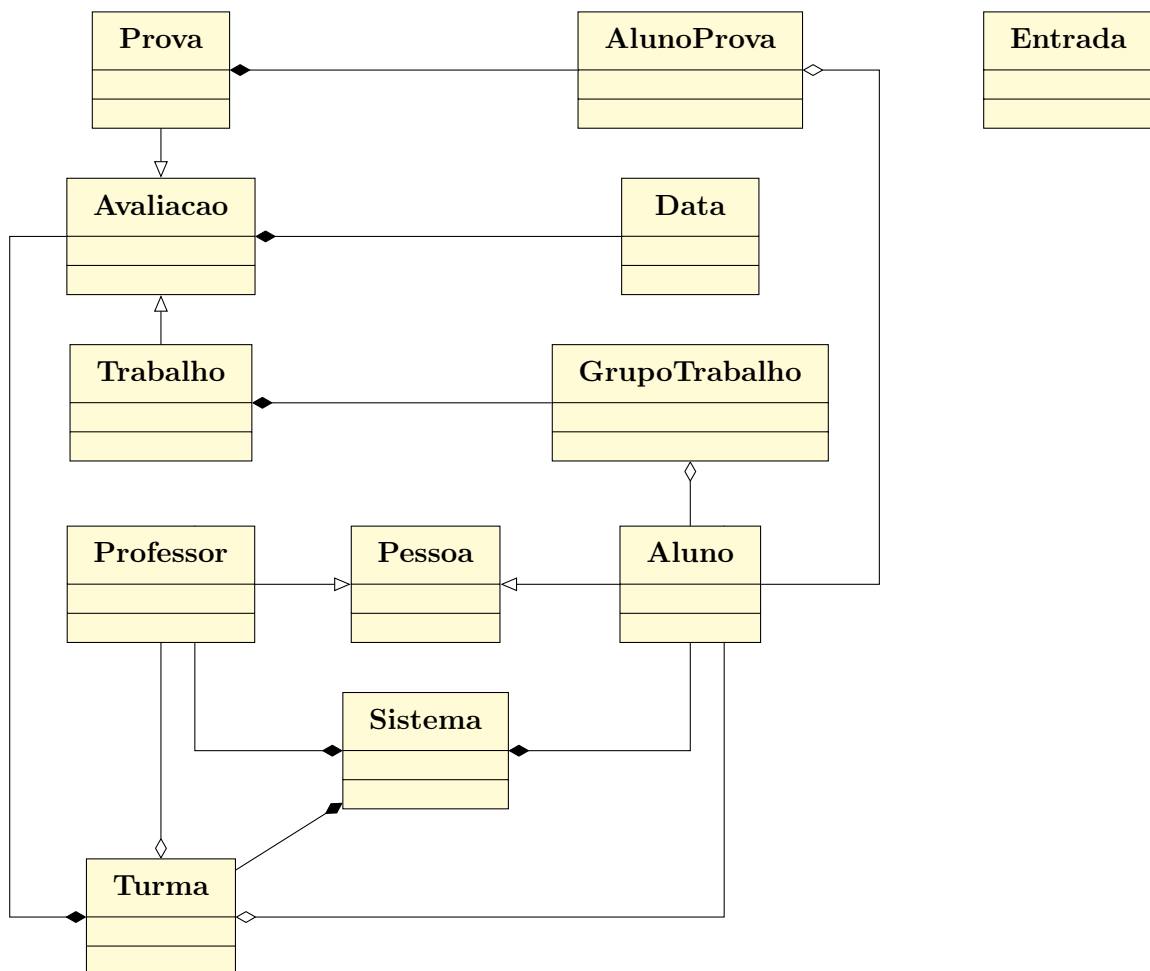
Atividade 3: Implementação

POO - BSI - Ifes Serra

25 de junho de 2025

1 Diagrama de Classes UML

Nesta atividade, você deve continuar a implementação do sistema feito na Atividade 2. As classes do sistema e as relações entre elas seguem as mesmas:



A leitura dos dados segue sendo feita pela classe **Entrada** (agora, tanto os dados lidos do teclado como os dados carregados dos arquivos textos).

Nesta terceira Atividade, os dados das pessoas e turmas devem ser salvos em um único arquivo texto chamado “*dados.txt*”. Este mesmo arquivo deve ser carregado automaticamente quando o programa for iniciado, e os dados contidos no arquivo devem ser lidos automaticamente. Por questões de praticidade, apenas os seguintes dados serão armazenados por enquanto:

- Professores: nome, cpf, salario (nesta ordem).

- Alunos: nome, cpf, matrícula (nesta ordem).
- Turmas: nome, ano, semestre, cpf do professor, número de alunos, matrícula de cada aluno, número de avaliações e dados de cada avaliação.
- Cada avaliação armazena: nome, data de aplicação, valor máximo e os dados específicos de prova/trabalho:
 - Se for uma prova: número de questões e, para cada aluno, sua nota em cada questão.
 - Se for um trabalho: número de máximo de integrantes, número de grupos e, para cada aluno no grupo: número de alunos no grupo, matrícula de cada aluno no grupo e nota do grupo.

Um professor será identificado no arquivo texto pela sigla PROF, um aluno será identificado pela sigla ALU, uma turma pela sigla TUR, uma prova pela sigla PROV e um trabalho pela sigla TRAB. A sigla FIM indica o final da leitura do arquivo.

Segue um exemplo do arquivo de entrada, contendo dados de 1 administrador e dois alunos:

```

PROF
hilario
123.456.789-00
1000.0
PROF
ernani
234.567.890-00
2000
ALU
Ana
111.111.111-11
A111
ALU
Bruno
222.222.222-22
B222
ALU
Carla
333.333.333-33
C333
ALU
Daniel
444.444.444-44
D444
ALU
Enzo
555.555.555-55
E555
ALU
Fred
666.666.666-66
F666
TUR

```

Prog2
2025
2
123.456.789-00
4
B222
C333
D444
E555
3
PROV
Prova 1
3
9
2025
30
3
10
10
10
9
8
7
10
10
10
10
10
5
TRAB
Trabalho 1
10
10
2025
40
2
2
2
B222
C333
35
2
D444
E555
42
PROV
Prova 2
12
12
2025

30
3
0
0
0
1
1
1
10
10
10
3
3
3
FIM

É necessário ainda:

- Incluir tratamento de erros e exceções para as interações do sistema com o usuário. O sistema deve ser robusto, de modo a não ser encerrado caso o usuário cometa algum erro.
- Transformar classes e métodos em classes abstratas e métodos abstratos onde for pertinente.
- Ao listar as turmas, deve ser adotada a seguinte ordem:
 1. A turma deve ser ordenada semestralmente, da mais recente para a mais antiga. Em caso de empate, em ordem crescente do nome da disciplina. Caso ainda haja empate, pelo nome do professor.
 2. Em cada turma, a listagem de alunos deve ser ordenada da maior nota final para a menor nota final. Em caso de empate, em ordem crescente do nome do aluno. Caso ainda haja empate, pela matrícula do aluno.

Exemplo: [EM BREVE]

2 Observações

1. A atividade vale 40 pontos e deve ser entregue até 08/07.
2. O código deve ser feito em Java, considerando as boas práticas de POO (como reuso de código, por exemplo, sempre que possível).
3. A atividade pode ser feita em grupos de até dois integrantes.
4. Atividades entregues após o prazo serão automaticamente rejeitados.
5. Atividades com erro de execução, com formato de saída incorreto, ou que não compilarem terão nota 0.
6. Atividades considerados plágio terão nota 0 para quem copiou e para quem forneceu o exercício, e serão enviados ao Conselho de Ética.
7. A Atividade deve ser enviado na sala da disciplina do AVA.

8. Em caso de dúvidas na especificação da atividade ou na própria atividade, contate-me em sala de aula.