

# COMP LabBook 2024 1 - E2

Lucas M. Schnorr

April 15, 2024

## Contents

<b>1</b>	<b>Logs de testes, tabelas de decisão e entradas de teste</b>	
<b>2</b>	<b>Para todos: Entradas corretas identificada como erradas</b>	
<b>3</b>	<b>Para todos: Entradas incorretas mas aceitas</b>	
<b>4</b>	<b>Comentários</b>	
4.1	Gerais . . . . .	
4.2	Comentários por Grupo . . . . .	
4.2.1	GrupoA . . . . .	
4.2.2	GrupoB . . . . .	
4.2.3	GrupoC . . . . .	
4.2.4	GrupoD . . . . .	
4.2.5	GrupoE . . . . .	
4.2.6	GrupoF . . . . .	
4.2.7	GrupoG . . . . .	
4.2.8	GrupoH . . . . .	
4.2.9	GrupoJ . . . . .	
4.2.10	GrupoK . . . . .	
4.2.11	GrupoL . . . . .	
4.2.12	GrupoP . . . . .	
4.2.13	GrupoS . . . . .	
4.2.14	GrupoV . . . . .	
4.2.15	GrupoW . . . . .	
4.2.16	GrupoX . . . . .	
4.2.17	GrupoY . . . . .	
4.2.18	GrupoZ . . . . .	
<b>5</b>	<b>Subjetivo</b>	
5.1	Tabela com valores dos critérios por grupo . . . . .	
5.2	Justificativas da tabela . . . . .	
5.2.1	Gramática (Sec 2.1) . . . . .	
5.2.2	Relatório de Erro (Sec 2.2) . . . . .	
5.2.3	Remoção de Conflitos (Sec 2.3) . . . . .	
5.2.4	Opinião . . . . .	
<b>6</b>	<b>Final</b>	
<b>7</b>	<b>Recuperação</b>	

## 1 Logs de testes, tabelas de decisão e entradas de teste

Os arquivos fornecidos no `e2_CSV_LOG.zip`.

- `e2_make.log`: o arquivo contém o log da compilação
- `e2.log`: o arquivo contém o log cru dos testes
- `e2.csv`: após parsing para obter apenas a saída, as colunas são as seguintes:
  - Grupo: identificador do grupo
  - Teste: identificador do teste
  - Result: resultado da solução do grupo
    - \* 0, aceitou a entrada
    - \* 1, refugou a entrada

- Razao
  - \* Vazio se aceitou a entrada
  - \* Erro sintático (possivelmente) informando o erro
- `e2_esperado.csv`: informada se a entrada é correta ou incorreta sintaticamente de acordo com a especificação da E2
  - Teste: identificador do teste
  - Esperado
    - \* 0, entrada sintaticamente correta de acordo com a E2
    - \* 1, entrada sintaticamente **incorreta** de acordo com a E2
- `e2_output_objetivo.csv`
  - E2.O: nota final do teste objetivo (sobre 10)
- `e2_output_quais_os_testes_mais_errados.csv`
  - Estatística que ilustra os testes que mais deram errado
- `e2_output_quem_errou_o_que.csv`
  - Indica apenas os testes que falharam, para cada grupo
  - Grupo, Test, Result, Esperado, Correto, Razao
    - \* Result: resultado da solução do grupo
    - \* Esperado: resultado esperado para aquele teste
  - Por exemplo:
    - \* Temos a linha `GrupoA,asl031,1,0,FALSE,line 4: syntax error`
      - Esse teste foi considerado errado pois o GrupoA informou erro sintático na entrada asl031 (1 na coluna Result) quando na realidade sintaticamente o teste está correto (0 na coluna Esperado).
    - \* Para fazer essas verificações, consulte o TGZ (listagem abaixo).

Os arquivos fornecidos no `E2.tgz`.

- Cada arquivo tem um comentário `//CORRECT` ou `//INCORRECT` para facilitar a identificação se o teste está correto ou incorreto do ponto de vista da E2. Estes comentários são utilizados para definir o conteúdo de `e2_esperado.csv`.

## 2 Para todos: Entradas corretas identificada como erradas

As entradas abaixo estão sintaticamente corretas (de acordo com a E2). Verifique porque sua gramática considera elas falhas e faça as alterações para aceitá-las como sintaticamente corretas.

Grupo	Teste
GrupoA	asl003
GrupoA	asl012
GrupoA	asl014
GrupoA	asl015
GrupoA	asl016
GrupoA	asl017
GrupoA	asl018
GrupoA	asl019
GrupoA	asl027
GrupoA	asl028
GrupoA	asl031
GrupoA	asl033
GrupoA	asl037
GrupoA	asl057
GrupoA	asl058
GrupoA	asl059
GrupoA	asl061
GrupoA	asl063
GrupoA	asl069
GrupoA	asl074
GrupoA	asl079
GrupoA	asl083
GrupoA	asl086
GrupoA	asl087
GrupoA	asl091

Continued on next page

Continued from previous page

Grupo	Teste
GrupoA	asl092
GrupoA	asl093
GrupoA	asl118
GrupoA	asl126
GrupoA	asl127
GrupoA	asl129
GrupoA	asl130
GrupoA	asl131
GrupoA	asl146
GrupoA	asl147
GrupoB	asl127
GrupoB	asl147
GrupoC	asl146
GrupoC	asl147
GrupoD	asl009
GrupoD	asl010
GrupoD	asl016
GrupoD	asl017
GrupoD	asl037
GrupoD	asl048
GrupoD	asl050
GrupoD	asl052
GrupoD	asl053
GrupoD	asl057
GrupoD	asl058
GrupoD	asl073
GrupoD	asl083
GrupoD	asl086
GrupoD	asl091
GrupoD	asl092
GrupoE	asl037
GrupoF	asl003
GrupoF	asl009
GrupoF	asl010
GrupoF	asl012
GrupoF	asl014
GrupoF	asl015
GrupoF	asl016
GrupoF	asl017
GrupoF	asl018
GrupoF	asl019
GrupoF	asl027
GrupoF	asl028
GrupoF	asl031
GrupoF	asl033
GrupoF	asl037
GrupoF	asl048
GrupoF	asl050
GrupoF	asl052
GrupoF	asl053
GrupoF	asl057
GrupoF	asl058
GrupoF	asl059
GrupoF	asl061
GrupoF	asl063
GrupoF	asl069
GrupoF	asl073
GrupoF	asl074
GrupoF	asl079
GrupoF	asl083
GrupoF	asl086
GrupoF	asl087
GrupoF	asl091
GrupoF	asl092
GrupoF	asl093
GrupoF	asl106

Continued on next page

Continued from previous page

Grupo	Teste
GrupoF	asl118
GrupoF	asl126
GrupoF	asl127
GrupoF	asl129
GrupoF	asl130
GrupoF	asl131
GrupoF	asl146
GrupoF	asl147
GrupoF	asl148
GrupoF	asl149
GrupoF	asl150
GrupoF	asl151
GrupoF	asl152
GrupoF	asl153
GrupoF	asl154
GrupoF	asl155
GrupoF	asl156
GrupoF	asl157
GrupoF	asl158
GrupoF	asl159
GrupoF	asl160
GrupoG	asl016
GrupoG	asl017
GrupoG	asl019
GrupoG	asl037
GrupoG	asl057
GrupoG	asl058
GrupoG	asl083
GrupoG	asl086
GrupoG	asl092
GrupoG	asl118
GrupoH	asl126
GrupoH	asl146
GrupoJ	asl146
GrupoJ	asl147
GrupoL	asl016
GrupoL	asl017
GrupoL	asl019
GrupoL	asl037
GrupoL	asl083
GrupoL	asl086
GrupoL	asl092
GrupoL	asl118
GrupoP	asl017
GrupoP	asl019
GrupoP	asl037
GrupoS	asl016
GrupoS	asl017
GrupoS	asl057
GrupoS	asl058
GrupoS	asl073
GrupoS	asl074
GrupoV	asl003
GrupoV	asl009
GrupoV	asl010
GrupoV	asl012
GrupoV	asl014
GrupoV	asl015
GrupoV	asl016
GrupoV	asl017
GrupoV	asl018
GrupoV	asl019
GrupoV	asl031
GrupoV	asl033
GrupoV	asl037
GrupoV	asl048

Continued on next page

Continued from previous page

Grupo	Teste
GrupoV	asl050
GrupoV	asl052
GrupoV	asl053
GrupoV	asl057
GrupoV	asl058
GrupoV	asl073
GrupoV	asl083
GrupoV	asl086
GrupoV	asl091
GrupoV	asl092
GrupoV	asl118
GrupoV	asl131
GrupoV	asl148
GrupoV	asl149
GrupoV	asl150
GrupoV	asl151
GrupoV	asl152
GrupoV	asl153
GrupoV	asl154
GrupoV	asl155
GrupoV	asl156
GrupoV	asl157
GrupoV	asl158
GrupoV	asl159
GrupoV	asl160
GrupoW	asl146
GrupoW	asl147
GrupoX	asl037
GrupoY	asl001
GrupoY	asl003
GrupoY	asl009
GrupoY	asl010
GrupoY	asl012
GrupoY	asl014
GrupoY	asl015
GrupoY	asl016
GrupoY	asl017
GrupoY	asl018
GrupoY	asl019
GrupoY	asl027
GrupoY	asl028
GrupoY	asl031
GrupoY	asl033
GrupoY	asl037
GrupoY	asl044
GrupoY	asl048
GrupoY	asl050
GrupoY	asl052
GrupoY	asl053
GrupoY	asl055
GrupoY	asl057
GrupoY	asl058
GrupoY	asl059
GrupoY	asl061
GrupoY	asl063
GrupoY	asl069
GrupoY	asl073
GrupoY	asl074
GrupoY	asl077
GrupoY	asl079
GrupoY	asl083
GrupoY	asl086
GrupoY	asl087
GrupoY	asl088
GrupoY	asl091
GrupoY	asl092

Continued on next page

Continued from previous page

Grupo	Teste
GrupoY	asl093
GrupoY	asl099
GrupoY	asl106
GrupoY	asl108
GrupoY	asl118
GrupoY	asl126
GrupoY	asl127
GrupoY	asl129
GrupoY	asl130
GrupoY	asl131
GrupoY	asl146
GrupoY	asl147
GrupoY	asl148
GrupoY	asl149
GrupoY	asl150
GrupoY	asl151
GrupoY	asl152
GrupoY	asl153
GrupoY	asl154
GrupoY	asl155
GrupoY	asl156
GrupoY	asl157
GrupoY	asl158
GrupoY	asl159
GrupoY	asl160
GrupoZ	asl146
GrupoZ	asl147

### 3 Para todos: Entradas incorretas mas aceitas

As entradas abaixo estão sintaticamente incorretas (de acordo com a E2). Verifique porque sua gramática considera elas corretas e faça as alterações para não aceitá-las como sintaticamente corretas.

Grupo	Teste
GrupoA	asl025
GrupoL	asl113
GrupoV	asl113

## 4 Comentários

### 4.1 Gerais

- Em todas as regras com recursão gramatical (listas de comandos, listas de funções, etc), dar preferência para a recursão à esquerda pois isso facilita as regras de associatividade (em expressões, por exemplo).
- Usar os contra-exemplos (rodar o `bison` com o parâmetro `-Wcounterexamples`) que demonstram as situações dos conflitos e auxiliam o grupo a entender como modificar a gramática para evitar aquele conflito.
  - Os contra-exemplos indicam os caminhos equivalentes para chegar no conflito. Deve-se editar a gramática então para que tenha um caminho único de derivações para chegar até a situação do contra-exemplo.

Os comentários abaixo focam na observação do código e a experiência do professor da execução dos testes com o código do grupo. Os comentários abaixo **não** mencionam os erros detectados através dos testes automáticos. Para estes, a sugestão é olhar o ZIP e o TGZ e a introdução deste relatório para decodificar os ajustes que devem ser feitos. Caso houverem dúvidas, entre em contato.

### 4.2 Comentários por Grupo

#### 4.2.1 GrupoA

- Associatividade e precedência não foram implementadas. Observa-se que as mesmas foram implementadas como listas (`expressionList`), algo que dificultará bastante a execução da próxima etapa.

#### 4.2.2 GrupoB

- Conforme estava especificado, recomendava-se evitar o uso de comandos do bison para resolver precedência (%precedence). Em princípio, o texto se corretamente interpretado e implementado, não carecia desse tipo de comandos pois não existem conflitos S/R.
- "unexpected invalid token" no teste asl127 e asl147. Isso indica algum problema no scanner.l

#### 4.2.3 GrupoC

- O arquivo parser.y poderia ter um pouco mais de comentários, por exemplo para categorizar o conjunto de regras gramaticais, facilitando o mapeamento com as seções da especificação.

#### 4.2.4 GrupoD

- Sem identificação de autoria no parser.y

#### 4.2.5 GrupoE

- No teste asl037, retorna um TK\_ERRO, algo incomum pois todos os tokens fornecidos em todos os testes são válidos. Isso indica algum problema no scanner.l

#### 4.2.6 GrupoF

- O arquivo poderia ter um pouco mais de comentários (além do relato sobre o qual já conversamos), por exemplo para categorizar o conjunto de regras gramaticais, facilitando o mapeamento com as seções da especificação.
- Na regra *cabecalho* faltou a / (por isso a minha insistência em revisar após a submissão precoce), fazendo com que a integralidade dos testes corretos seja considerado errada. A nota objetiva é ilusória pois o compilador sistematicamente retorna erro sintático e 63% dos testes são sabidamente incorretos sintaticamente. Recomenda-se fortemente a correção desse problema caso contrário nas próximas etapas todos os testes falharão por erro sintático.

#### 4.2.7 GrupoG

- Recomenda-se não colocar o arquivo parser.y no diretório *include* (que seria para colocar apenas os cabeçalhos - arquivos .h)

#### 4.2.8 GrupoH

#### 4.2.9 GrupoJ

#### 4.2.10 GrupoK

#### 4.2.11 GrupoL

#### 4.2.12 GrupoP

- No teste asl037, retorna um TK\_ERRO, algo incomum pois todos os tokens fornecidos em todos os testes são válidos. Isso indica algum problema no scanner.l

#### 4.2.13 GrupoS

#### 4.2.14 GrupoV

- Sem identificação de autoria no parser.y
- Sem comentários no arquivo parser.y, algo que seria útil - por exemplo - para categorizar o conjunto de regras gramaticais, facilitando o mapeamento com as seções da especificação.
- Na implementação das expressões, há um problema pois a associatividade não foi implementada. Faça o teste, por exemplo, do encadeamento de duas operações or (ou and, ou ...).

#### 4.2.15 GrupoW

#### 4.2.16 GrupoX

- Sem identificação de autoria no parser.y
- No teste asl037, retorna um TK\_ERRO, algo incomum pois todos os tokens fornecidos em todos os testes são válidos. Isso indica algum problema no scanner.l

#### 4.2.17 GrupoY

- Relatório de erro contém apenas "syntax error"
- Na compilação do bison, temos

```
parser.y: warning: 30 nonterminals useless in grammar [-Wother]
parser.y: warning: 69 rules useless in grammar [-Wother]
```

indicando que existem 30 não-terminais e 69 regras sem uso. Alguma ideia de porque não se teve o cuidado de resolver esses problemas?

- Bom, o fato de ter esquecido `program: TK_LIT_INT` realmente prejudica o grupo visto que este (`program`) é considerado o símbolo não-terminal inicial da gramática. O professor então resolveu esse problema substituindo `TK_LIT_INT` por `elements_list` mas daí o bison reporta um conflito R/R (indicando ambiguidade, que inexistia na especificação).

#### 4.2.18 GrupoZ

- Sem comentários no arquivo `parser.y`, algo que seria útil - por exemplo - para categorizar o conjunto de regras gramaticais, facilitando o mapeamento com as seções da especificação.

## 5 Subjetivo

### 5.1 Tabela com valores dos critérios por grupo

Grupo	Nota	Critério
GrupoA	7	Gramática
GrupoB	9	Gramática
GrupoC	9	Gramática
GrupoD	8	Gramática
GrupoE	9	Gramática
GrupoF	6	Gramática
GrupoG	9	Gramática
GrupoH	9	Gramática
GrupoJ	9	Gramática
GrupoK	10	Gramática
GrupoL	9	Gramática
GrupoP	9	Gramática
GrupoS	9	Gramática
GrupoV	7	Gramática
GrupoW	9	Gramática
GrupoX	9	Gramática
GrupoY	5	Gramática
GrupoZ	9	Gramática
GrupoA	10	Relatório de Erro
GrupoB	10	Relatório de Erro
GrupoC	10	Relatório de Erro
GrupoD	10	Relatório de Erro
GrupoE	10	Relatório de Erro
GrupoF	10	Relatório de Erro
GrupoG	10	Relatório de Erro
GrupoH	10	Relatório de Erro
GrupoJ	10	Relatório de Erro
GrupoK	10	Relatório de Erro
GrupoL	10	Relatório de Erro
GrupoP	10	Relatório de Erro
GrupoS	10	Relatório de Erro
GrupoV	10	Relatório de Erro
GrupoW	10	Relatório de Erro
GrupoX	10	Relatório de Erro
GrupoY	0	Relatório de Erro
GrupoZ	10	Relatório de Erro
GrupoA	10	Remoção de Conflitos
GrupoB	10	Remoção de Conflitos
GrupoC	10	Remoção de Conflitos
GrupoD	10	Remoção de Conflitos
GrupoE	10	Remoção de Conflitos

Continued on next page



Continued from previous page

Grupo	Nota	Critério
GrupoF	10	Remoção de Conflitos
GrupoG	10	Remoção de Conflitos
GrupoH	10	Remoção de Conflitos
GrupoJ	10	Remoção de Conflitos
GrupoK	10	Remoção de Conflitos
GrupoL	10	Remoção de Conflitos
GrupoP	10	Remoção de Conflitos
GrupoS	10	Remoção de Conflitos
GrupoV	10	Remoção de Conflitos
GrupoW	10	Remoção de Conflitos
GrupoX	10	Remoção de Conflitos
GrupoY	5	Remoção de Conflitos
GrupoZ	10	Remoção de Conflitos
GrupoA	8	Opinião
GrupoB	8.5	Opinião
GrupoC	9.5	Opinião
GrupoD	9.5	Opinião
GrupoE	9.5	Opinião
GrupoF	8	Opinião
GrupoG	9.5	Opinião
GrupoH	10	Opinião
GrupoJ	10	Opinião
GrupoK	10	Opinião
GrupoL	10	Opinião
GrupoP	9.5	Opinião
GrupoS	10	Opinião
GrupoV	8	Opinião
GrupoW	10	Opinião
GrupoX	9	Opinião
GrupoY	5	Opinião
GrupoZ	9.5	Opinião

## 5.2 Justificativas da tabela

### 5.2.1 Gramática (Sec 2.1)

Alinha-se com os resultados objetivos, com a função `trunc` (obtem o valor inteiro do ponto-flutuante).

### 5.2.2 Relatório de Erro (Sec 2.2)

- Nota 10: grupo apresentou solução com “mensagem de erro informando a linha do código da entrada que gerou o erro sintático e informações adicionais que auxiliem o programador que está utilizando o compilamensagem de erro informando a linha do código da”
- Nota 5: grupo apresentou solução que apenas identifica o número da linha onde aconteceu o erro sintático, sem informar qual foi o erro especificamente OU grupo apresentou solução que apenas identifica o erro, sem mencionar o número da linha
- Nota 0: grupo apresentou solução que apenas identifica que houve um erro, sem informar o número da linha ou qual foi o erro, ou não entregou.

Sugestão:

- Ative `%define parse.error verbose` no `parser.y`

### 5.2.3 Remoção de Conflitos (Sec 2.3)

- Nota 10: sem conflitos `shift/reduce` ou `reduce/reduce`
- Nota 5: apenas um conflito `shift/reduce` ou `reduce/reduce`
- Nota 0: com pelo menos um conflito acima ou `expect` não justificado

### 5.2.4 Opinião

Baseada na combinação da análise objetiva, com relatório de erro, remoção de conflitos e gramática, além da revisão das regras gramaticais, adequação às regras gerais, e demais comentários ressaltados acima.

## 6 Final

Grupo	E2.O	E2.S	E2.P	E2
GrupoA	7.65	8.3	1	7.98
GrupoB	9.87	9.1	1	9.48
GrupoC	9.87	9.5	1	9.68
GrupoD	8.95	9.2	1	9.07
GrupoE	9.93	9.5	1	9.71
GrupoF	6.34	8	1	7.17
GrupoG	9.35	9.5	1	9.43
GrupoH	9.87	9.7	1	9.79
GrupoJ	9.87	9.7	1	9.79
GrupoK	10	10	1	10
GrupoL	9.41	9.7	1	9.55
GrupoP	9.8	9.5	1	9.65
GrupoS	9.61	9.7	1	9.65
GrupoV	7.39	8.3	1	7.85
GrupoW	9.87	9.7	1	9.79
GrupoX	9.93	9.3	1	9.62
GrupoY	5.88	4	1	4.94
GrupoZ	9.87	9.5	1	9.68

## 7 Recuperação

Grupos em recuperação da E2:

- Em Moodle já configurado para tal, use o novo link "Recuperação E2"
- Política de recuperação ativada (vejam regras gerais)

GrupoY