

## Recomendações/observações aos grupos

### Grupo 1

- Interface gráfica avançada (+ 0.6 da nota extra)
- Mensagens de erro bem detalhadas (+ 0.4 da nota extra)

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Há caso em que é gerada uma cascata de erros muito extensa (tokens de sincronização)

### Grupo 2

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos

### Grupo 3

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Em alguns casos, identifica o erro corretamente mas a mensagem indica um token esperado errado
- 3- Para alguns casos o compilador quebra

### Grupo 4

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Grupo poderia melhorar comentário internos no código → -0.1

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Em alguns casos, identifica o erro corretamente mas a mensagem indica um token esperado errado

### Grupo 5

Makefile → +0.2 da Nota Extra

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Reporta alguns erros sintáticos inexistentes, e é omissa para alguns erros (verificar as produções)

### Grupo 6

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Makefile → +0.2 da Nota Extra

Programa não exibe mensagem quando a cadeia é toda reconhecida -0.2 da usabilidade

Erros nos Casos de Teste:

- 1- Informa erro sintático inexistente após identificar erro léxico
- 2- Em alguns casos, identifica o erro corretamente mas a mensagem indica um token esperado errado

### Grupo 7

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Grupo poderia melhorar comentário internos no código → -0.1

Makefile → +0.2 da Nota Extra

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Reporta alguns erros sintáticos inexistentes, e é omissa para alguns erros (verificar as produções)
- 3- Em alguns casos é dada a mensagem genérica 'compilation error'

Não reporta a linha onde foi encontrado erro → -0.3 da usabilidade

### Grupo 8

Relatório bem completo e com os grafos desenhados → +0.3 da nota extra

- 1- Problema na identificação de alguns erros léxicos
- 2- Reporta alguns erros sintáticos inexistentes, e é omissa para alguns erros (verificar as produções)

### **Grupo 9**

Makefile → +0.2 da Nota Extra

Mensagens de erro bem detalhadas (+ 0.4 da nota extra)

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Reporta alguns erros sintáticos inexistentes, e é omissa para alguns erros (verificar as produções)

### **Grupo 11**

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Grupo não faz documentação interna do código → -0.5

Erros nos Casos de Teste:

- 1- Reporta alguns erros sintáticos inexistentes, e é omissa para alguns erros (verificar as produções)
- 2- Problema na identificação de alguns erros léxicos

### **Grupo 12**

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Dá exceção em um teste

### **Grupo 13**

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Reporta alguns erros sintáticos inexistentes (verificar as produções)
- 3- Em alguns casos, identifica o erro corretamente mas a mensagem indica um token esperado errado

Usabilidade pode ser melhorada (explicar como usar quando não são passados os parâmetros ou redirecionado stdin) → -0.

Makefile → +0.2 da Nota Extra

### **Grupo 14**

Erros nos Casos de Teste:

- 1- Problema na identificação de alguns erros léxicos
- 2- Reporta alguns erros sintáticos inexistentes, e é omissa para alguns erros (verificar as produções)
- 3- Sem regra para FOR
- 4- Não identifica erro léxico para comentário mal formado

Sem relatório → -0.5

### **Grupo 15**

Erros nos Casos de Teste:

- 1- Informa erro sintático inexistente após identificar erro léxico
- 2- Em alguns casos, identifica o erro corretamente mas a mensagem indica um token esperado errado
- 3- Para alguns casos em que ocorre erro léxico, o compilador quebra
- 4- Não identifica erro léxico para comentário mal formado
- 5- Em alguns casos é dada a mensagem genérica 'syntax error'
- 6- Em alguns casos mostra na mensagem de erro apenas um possível token, mas vários poderiam aparecer corretamente

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Não reporta a linha onde foi encontrado erro → -0.3 da usabilidade

### **Grupo 16**

Erros nos Casos de Teste:

Há caso em que a compilação quebra ao encontrar erro sintático

Há caso em que é gerada uma cascata de erros muito extensa (tokens de sincronização)

Resultado da compilação detalha os tokens de sincronização (modo pânico) → +0.3 da nota extra

### **Grupo 17**

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Grupo poderia melhorar comentário internos no código → -0.1

Código não compilou. Foi necessário mudar a biblioteca .h no arquivo do lex para funcionar. (-0.5 da modularidade)

Erros nos Casos de Teste:

1- Não reporta alguns erros léxicos

2 – Reporta erros sináticos inconsistentes

### **Grupo 18**

Erros nos Casos de Teste:

1- Em alguns casos, identifica o erro corretamente mas a mensagem indica um token esperado errado

2- Não identifica erro léxico para comentário mal formado

3- Código numérico para token encontrado nas mensagens de erro (-0,2 da usabilidade)

4- Não reporta a linha onde foi encontrado erro (-0,2 da usabilidade)

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

### **Grupo 19**

Erros nos Casos de Teste:

1- Indica erro sintático inexistente após encontrar erro léxico, em alguns casos

2- Há caso em que é gerada uma cascata de erros muito extensa (tokens de sincronização)

3- Não identifica erro léxico para comentário mal formado

Sem relatório → -0.5

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1

Mensagens de erro bem detalhadas (+ 0.4 da nota extra)

### **Grupo 20**

Erros nos Casos de Teste:

1- Compilação quebra ao encontrar erro léxico, em alguns casos

2- Indentifica erro sintático inexistente em alguns casos

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros); e não amarrar entrada ao código

### **Grupo 21**

Erros nos Casos de Teste:

1- Compilação quebra ao encontrar erro léxico, em alguns casos

2- Compilação quebra ao encontrar erro sintático, em alguns casos

3- Indentifica erro sintático inexistente em alguns casos

4- Há caso em que é gerada uma cascata de erros muito extensa (tokens de sincronização)

5- Compilação quebra após reportar erro inexistente

Sem relatório → -0.5

### **Grupo 22**

Erros nos Casos de Teste:

1- Compilação pára ao encontrar o primeiro erro léxico ou sintático (especificado no relatório, porém não é o comportamento

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros)

Mensagens de erro bem detalhadas (+ 0.4 da nota extra)

### **Grupo 23**

Erros nos Casos de Teste:

1- Indentifica erro sintático inexistente em alguns casos

2- Há caso em que é gerada uma cascata de erros muito extensa (tokens de sincronização)

3- Sem regra para FOR

4- Loop infinito em caso de erro léxico de comentário mal formado

Usabilidade pode ser melhorada (explicar como usar quando não são passados parâmetros) → -0.1