

## Второй рубежный контроль по курсу «Конструирование компиляторов»

### Билет 1

**Вопрос.** Постановка задачи синтаксического анализа.

**Задача.** Дан язык арифметических выражений с операторами сравнения  $>$ ,  $<$ ,  $=$  и тернарным оператором  $\dots ? \dots : \dots$ . Корректные предложения языка:

- $n + n$
- $n / (n - n)$
- $n > n ? n * n : n + n$
- $n > n ? n + n : n < n ? n - n : n + n + n$

При этом тернарный оператор правоассоциативный: выражение  $e1 ? e2 : e3 ? e4 : e5$  трактуется как  $e1 ? e2 : (e3 ? e4 : e5)$ , а не как  $(e1 ? e2 : e3) ? e4 : e5$ .

Операторы сравнения неассоциативны, т.е. выражение вида  $n > n > n$  некорректно.

Нужно записать грамматику этого языка.

### Билет 2

**Вопрос.** Алгоритм предсказывающего синтаксического разбора.

**Задача.** Дан язык арифметических выражений с вызовами функций и условным оператором **if** ... **then** ... **else**. Корректные предложения этого языка:

- $n + n$
- $id * n$
- $id(n, id) / id()$
- **if**  $id(id, n)$  **then** **if**  $id(id, id, id / (id - n))$  **then**  $n$  **else**  $n$  **else**  $id$

Здесь  $id$  соответствует лексеме домена идентификаторов,  $n$  — лексеме домена чисел.

Нужно записать LL(1)-грамматику этого языка.

### Билет 3

**Вопрос.** Множества FIRST и FOLLOW.

**Задача.** Дан императивный язык с операторами **begin ... end, if ... then ..., if ... then ... else ...** и точкой с запятой в качестве разделителя операторов.

При этом **else** связывается с ближайшим оператором **if**: в конструкции

**if E1 then S1 if E2 then S2 else S3**

ветка **else S3** привязывается к **if E2**, а не **if E1**.

Корректные предложения для языка:

- **if E then if E then S**
- **begin if E then S; if E then S end**
- **if E then begin if E then S; if E then S end**
- **begin S; S; S; end**

Буква E означает некоторое выражение, буква S — примитивный оператор (вроде присваивания, вызова подпрограммы, return). Следует считать их лексемами.

Нужно записать грамматику и построить таблицу предсказывающего анализа.

### Билет 4

**Вопрос.** LL(1)-грамматики.

**Задача.** Дано подмножество XML:

```
<tag attr="string" attr=true>
  <tag/>
  <tag>
    <tag attr=10/>
    <tag attr=10/>
  </tag>
  <tag attr = "string" attr=10/>
</tag>
```

У открывающих и непарных тегов может быть ноль или более атрибутов, у закрывающих тегов атрибутов быть не может. Текста между тегами быть не может.

Запишите грамматику этого языка.

### Билет 5

**Вопрос.** Построение таблиц предсказывающего анализатора.

**Задача.** Дан язык арифметических выражений с операторами сравнения  $>$ ,  $<$ ,  $=$  и тернарным оператором  $\dots ? \dots : \dots$ . Корректные предложения языка:

- $n + n$
- $n / (n - n)$
- $n > n ? n * n : n + n$
- $n > n ? n + n : n < n ? n - n : n + n + n$

При этом тернарный оператор правоассоциативный: выражение  $e1 ? e2 : e3 ? e4 : e5$  трактуется как  $e1 ? e2 : (e3 ? e4 : e5)$ , а не как  $(e1 ? e2 : e3) ? e4 : e5$ .

Операторы сравнения неассоциативны, т.е. выражение вида  $n > n > n$  некорректно.

Нужно записать грамматику этого языка.

### Билет 6

**Вопрос.** Рекурсивный спуск.

**Задача.** Дан язык арифметических выражений с вызовами функций и условным оператором **if** ... **then** ... **else**. Корректные предложения этого языка:

- $n + n$
- $id * n$
- $id(n, id) / id()$
- **if**  $id(id, n)$  **then** **if**  $id(id, id, id / (id - n))$  **then**  $n$  **else**  $n$  **else**  $id$

Здесь  $id$  соответствует лексеме домена идентификаторов,  $n$  — лексеме домена чисел.

Нужно записать LL(1)-грамматику этого языка.

### Билет 7

**Вопрос.** Алгоритм Эрли.

**Задача.** Дан императивный язык с операторами **begin ... end, if ... then ..., if ... then ... else ...** и точкой с запятой в качестве разделителя операторов.

При этом **else** связывается с ближайшим оператором **if**: в конструкции

**if E1 then S1 if E2 then S2 else S3**

ветка **else S3** привязывается к **if E2**, а не **if E1**.

Корректные предложения для языка:

- **if E then if E then S**
- **begin if E then S; if E then S end**
- **if E then begin if E then S; if E then S end**
- **begin S; S; S; end**

Буква E означает некоторое выражение, буква S — примитивный оператор (вроде присваивания, вызова подпрограммы, return). Следует считать их лексемами.

Нужно записать грамматику и построить таблицу предсказывающего анализа.

### Билет 8

**Вопрос.** Расширенная БНФ. Построение множества FIRST для РБНФ.

**Задача.** Дано подмножество XML:

```
<tag attr="string" attr=true>
  <tag/>
  <tag>
    <tag attr=10/>
    <tag attr=10/>
  </tag>
  <tag attr = "string" attr=10/>
</tag>
```

У открывающих и непарных тегов может быть ноль или более атрибутов, у закрывающих тегов атрибутов быть не может. Текста между тегами быть не может.

Запишите грамматику этого языка.

### Билет 9

**Вопрос.** Синтаксический разбор типа «перенос-свёртка».

**Задача.** Дан язык арифметических выражений с операторами сравнения  $>$ ,  $<$ ,  $=$  и тернарным оператором  $\dots ? \dots : \dots$ . Корректные предложения языка:

- $n + n$
- $n / (n - n)$
- $n > n ? n * n : n + n$
- $n > n ? n + n : n < n ? n - n : n + n + n$

При этом тернарный оператор правоассоциативный: выражение  $e1 ? e2 : e3 ? e4 : e5$  трактуется как  $e1 ? e2 : (e3 ? e4 : e5)$ , а не как  $(e1 ? e2 : e3) ? e4 : e5$ .

Операторы сравнения неассоциативны, т.е. выражение вида  $n > n > n$  некорректно.

Нужно записать грамматику этого языка.

### Билет 10

**Вопрос.** Недетерминированные SLR-распознаватели.

**Задача.** Дан язык арифметических выражений с вызовами функций и условным оператором **if** ... **then** ... **else**. Корректные предложения этого языка:

- $n + n$
- $id * n$
- $id(n, id) / id()$
- **if**  $id(id, n)$  **then** **if**  $id(id, id, id / (id - n))$  **then**  $n$  **else**  $n$  **else**  $id$

Здесь  $id$  соответствует лексеме домена идентификаторов,  $n$  — лексеме домена чисел.

Нужно записать LL(1)-грамматику этого языка.

### Билет 11

**Вопрос.** Детерминированные SLR-распознаватели.

**Задача.** Дан императивный язык с операторами **begin ... end, if ... then ..., if ... then ... else ...** и точкой с запятой в качестве разделителя операторов.

При этом **else** связывается с ближайшим оператором **if**: в конструкции

**if E1 then S1 if E2 then S2 else S3**

ветка **else S3** привязывается к **if E2**, а не **if E1**.

Корректные предложения для языка:

- **if E then if E then S**
- **begin if E then S; if E then S end**
- **if E then begin if E then S; if E then S end**
- **begin S; S; S; end**

Буква E означает некоторое выражение, буква S — примитивный оператор (вроде присваивания, вызова подпрограммы, return). Следует считать их лексемами.

Нужно записать грамматику и построить таблицу предсказывающего анализа.

### Билет 12

**Вопрос.** Алгоритм LR(1)-анализа.

**Задача.** Дано подмножество XML:

```
<tag attr="string" attr=true>
  <tag/>
  <tag>
    <tag attr=10/>
    <tag attr=10/>
  </tag>
  <tag attr = "string" attr=10/>
</tag>
```

У открывающих и непарных тегов может быть ноль или более атрибутов, у закрывающих тегов атрибутов быть не может. Текста между тегами быть не может.

Запишите грамматику этого языка.