CS 132 Compiler Construction, Fall 2020
Instructor: Jens Palsberg
Multiple Choice Exam, Nov 16, 2020

ID

Name

This exam consists of 56 questions. Each question has four options, exactly one of which is correct, while the other three options are incorrect. For each question, you can check multiple options.

I will grade each question in the following way.

- If you check *none* of the options, you get 0 points.

- If you check all *four* options, you get 0 points.

- **Check one option.** If you check *one* option, and that option is correct, you get 2 points. If you check *one* option, and that option is wrong, you get –0.667 points (yes, negative!).

- **Check two options.** If you check *two* options, and one of those options is correct, you get 1 point. If you check *two* options, and both of them are wrong, you get –1 point (yes, negative!).

- **Check three options.** If you check *three* options, and one of those options is correct, you get 0.415 points. If you check *three* options, and all three of them are wrong, you get –1.245 points (yes, negative!).

The maximum point total is $56 \times 2 = 112$ points. I will calculate a percentage based on the points in the following way:

$$\frac{\max(0, \text{point total})}{112} \times 100$$

Notice that if your point total is negative, you will get 0 percent. Submit on CCLE (under Midterm exam) by uploading a text file with your answers. CCLE gives detailed instructions about the format of the text file.

Example
Consider the grammar

$$
\begin{array}{lcl}
A & ::= & w\ C \quad | \quad z\ B \\
B & ::= & x\ y\ C \quad | \quad \epsilon \\
C & ::= & B\ A \quad | \quad y
\end{array}
$$

where $\{A,B,C\}$ is the set of nonterminal symbols, A is the start symbol, $\{w,x,y,z\}$ is the set of terminal symbols, and ϵ denotes the empty string. The grammar is LL(1). Assume that a recursive-descent parser for the above grammer declares a variable `next` of type `token`, and that the program has three procedures `A()`, `B()`, `C()`, and the following main part:

```
void main() { next = getnexttoken(); A(); eat(EOF); }
```

The procedure `getnexttoken()` gets the next token from an input file. The procedure `eat()` is here written in pseudo-code:

```
void eat(token t) { if (t == next) { next = getnexttoken(); } else { error(); }
```

Question 1
The procedure `A()` contains the snippet:

```
if (next == x) { ???? }
```

What is "????"?
a ☐ `eat(x); eat(y); C();`
b ☐ `eat(z); B();`
c ☐ `error();`
d ☐ `eat(w); C();`

Question 2
The procedure `C()` contains the snippet:

```
if (next == x) { ???? }
```

What is "????"?
a ☐ `error();`
b ☐ `eat(y);`
c ☐ `B(); A();`
d ☐ `eat(w); C();`

Question 3
The procedure `C()` contains the snippet:

```
if (next == w) { ???? }
```

What is "????"?
a ☐ `eat(z); B();`
b ☐ `B(); A();`
c ☐ `eat(y);`
d ☐ `error();`

Question 4

The procedure B() contains the snippet:

```
if (next == z) { ???? }
```

What is "????"?

a ☐ error();

b ☐ /* do nothing */

c ☐ eat(w); C();

d ☐ eat(x); eat(y); C();

Question 5

The procedure A() contains the snippet:

```
if (next == z) { ???? }
```

What is "????"?

a ☐ eat(w); C();

b ☐ eat(z); B();

c ☐ error();

d ☐ eat(y);

Question 6

The procedure B() contains the snippet:

```
if (next == x) { ???? }
```

What is "????"?

a ☐ error();

b ☐ eat(x); eat(y); C();

c ☐ /* do nothing */

d ☐ eat(z); B();

Question 7

The procedure B() contains the snippet:

```
if (next == w) { ???? }
```

What is "????"?

a ☐ eat(x); eat(y); C();

b ☐ /* do nothing */

c ☐ error();

d ☐ eat(y);

Question 8

The procedure `B()` contains the snippet:

```
if (next == y) { ???? }
```

What is "????"?

a ☐ `eat(y);`
b ☐ `/* do nothing */`
c ☐ `eat(x); eat(y); C();`
d ☐ `error();`

Question 9

The procedure `C()` contains the snippet:

```
if (next == y) { ???? }
```

What is "????"?

a ☐ `B(); A();`
b ☐ `eat(y);`
c ☐ `eat(z); B();`
d ☐ `error();`

Question 10

The procedure `C()` contains the snippet:

```
if (next == z) { ???? }
```

What is "????"?

a ☐ `eat(y);`
b ☐ `error();`
c ☐ `B(); A();`
d ☐ `eat(w); C();`

Question 11

The procedure `A()` contains the snippet:

```
if (next == w) { ???? }
```

What is "????"?

a ☐ `eat(w); C();`
b ☐ `error();`
c ☐ `eat(z); B();`
d ☐ `eat(y);`

Question 12

The procedure `A()` contains the snippet:

```
if (next == y) { ???? }
```

What is "????"?
a☐ `eat(w); C();`
b☐ `error();`
c☐ `eat(z); B();`
d☐ `B(); A();`

Example

Consider the grammar

$$
\begin{aligned}
A &::= B\ A\ x\ \mid\ \epsilon\ \mid\ x\ z\ z \\
B &::= x\ y\ z\ \mid\ B\ z\ C \\
C &::= z\ B\ w
\end{aligned}
$$

where {A,B,C} is the set of nonterminal symbols, A is the start symbol, {w,x,y,z} is the set of terminal symbols, and ϵ denotes the empty string.

Question 13

Which nonterminals are nullable?
a☐ $\{B, C\}$
b☐ $\{B\}$
c☐ $\{A\}$
d☐ $\{\}$

Question 14

What is Follow(B)?
a☐ $\{\$, w, z\}$
b☐ $\{\$, x\}$
c☐ $\{w, x\}$
d☐ $\{w, x, z\}$

Question 15

What is Follow(C)?
a☐ $\{x\}$
b☐ $\{\}$
c☐ $\{w, x, z\}$
d☐ $\{\$, w, x\}$

Question 16

Is the grammar LL(1)?
a☐ Yes
b☐ The question cannot be answered with the information provided.
c☐ No
d☐ The LL(1)-checker would go into an infinite loop.

Question 17

What is First(A)?

a ☐ $\{z\}$
b ☐ $\{w, x, z\}$
c ☐ $\{x\}$
d ☐ $\{\}$

Question 18

What is First(B)?

a ☐ $\{x, y, z\}$
b ☐ $\{x, z\}$
c ☐ $\{\}$
d ☐ $\{x\}$

Question 19

What is First(C)?

a ☐ $\{z\}$
b ☐ $\{x, y\}$
c ☐ $\{\}$
d ☐ $\{w, x, z\}$

Question 20

What is Follow(A)?

a ☐ $\{x\}$
b ☐ $\{\$, x\}$
c ☐ $\{x, y\}$
d ☐ $\{w, y, z\}$

Example

Consider the grammar

$$
\begin{aligned}
A &::= \ z\ B \ \ | \ \ y\ w \\
B &::= \ A\ z\ A \ \ | \ \ \epsilon \ \ | \ \ w\ B\ C \\
C &::= \ A\ B \ \ | \ \ z\ A\ y
\end{aligned}
$$

where $\{A,B,C\}$ is the set of nonterminal symbols, A is the start symbol, $\{w,y,z\}$ is the set of terminal symbols, and ϵ denotes the empty string.

Question 21

Is the grammar LL(1)?

a ☐ The LL(1)-checker would go into an infinite loop.
b ☐ Yes
c ☐ The question cannot be answered with the information provided.
d ☐ No

Question 22
What is First(C)?
a ☐ $\{y, z\}$
b ☐ $\{z\}$
c ☐ $\{w\}$
d ☐ $\{\}$

Question 23
Which nonterminals are nullable?
a ☐ $\{A, C\}$
b ☐ $\{\}$
c ☐ $\{B\}$
d ☐ $\{A\}$

Question 24
What is First(B)?
a ☐ $\{z\}$
b ☐ $\{y, z\}$
c ☐ $\{w, y\}$
d ☐ $\{w, y, z\}$

Question 25
What is First(A)?
a ☐ $\{y, z\}$
b ☐ $\{w\}$
c ☐ $\{w, y\}$
d ☐ $\{y\}$

Question 26
What is Follow(A)?
a ☐ $\{z\}$
b ☐ $\{y\}$
c ☐ $\{\$, w, y\}$
d ☐ $\{\$, w, y, z\}$

Question 27
What is Follow(B)?
a ☐ $\{\$, w, y, z\}$
b ☐ $\{w, y, z\}$
c ☐ $\{y, z\}$
d ☐ $\{w, y\}$

Question 28

What is Follow(C)?

a ☐ $\{w, y, z\}$
b ☐ $\{w, y\}$
c ☐ $\{\$, w, y, z\}$
d ☐ $\{y\}$

Example

Consider the grammar

$$
\begin{aligned}
A & ::= \quad \epsilon \quad | \quad w \quad | \quad z\ C\ B \\
B & ::= \quad z \quad | \quad \epsilon \quad | \quad w\ C\ B \\
C & ::= \quad x\ x\ z \quad | \quad z\ z\ z
\end{aligned}
$$

where {A,B,C} is the set of nonterminal symbols, A is the start symbol, {w,x,z} is the set of terminal symbols, and ϵ denotes the empty string.

Question 29

Is the grammar LL(1)?

a ☐ The question cannot be answered with the information provided.
b ☐ Yes
c ☐ The LL(1)-checker would go into an infinite loop.
d ☐ No

Question 30

What is First(C)?

a ☐ $\{w, z\}$
b ☐ $\{x, z\}$
c ☐ $\{w, x\}$
d ☐ $\{x\}$

Question 31

What is First(B)?

a ☐ $\{\}$
b ☐ $\{x\}$
c ☐ $\{w, z\}$
d ☐ $\{z\}$

Question 32

What is Follow(A)?

a ☐ $\{w\}$
b ☐ $\{\$, w\}$
c ☐ $\{\$\}$
d ☐ $\{w, x, z\}$

Question 33
What is Follow(B)?
a ☐ $\{\$, x\}$
b ☐ $\{\$\}$
c ☐ $\{z\}$
d ☐ $\{\}$

Question 34
What is Follow(C)?
a ☐ $\{\$, w, z\}$
b ☐ $\{\}$
c ☐ $\{\$, w, x\}$
d ☐ $\{w\}$

Question 35
Which nonterminals are nullable?
a ☐ $\{B, C\}$
b ☐ $\{A, B\}$
c ☐ $\{A\}$
d ☐ $\{\}$

Question 36
What is First(A)?
a ☐ $\{z\}$
b ☐ $\{w, z\}$
c ☐ $\{x, z\}$
d ☐ $\{x\}$

Example
Consider the grammar

$$
\begin{aligned}
A &::= \quad w\ C \quad | \quad z\ B \\
B &::= \quad x\ y\ C \quad | \quad \epsilon \\
C &::= \quad B\ A \quad | \quad y
\end{aligned}
$$

where $\{A,B,C\}$ is the set of nonterminal symbols, A is the start symbol, $\{w,x,y,z\}$ is the set of terminal symbols, and ϵ denotes the empty string. The grammar is LL(1). The predictive parsing table is a two-dimensional table called *table*.

Question 37
What does table(B,x) contain?
a ☐ ϵ
b ☐ $x\ y\ C$
c ☐ y
d ☐   error

Question 38

What does table(C,y) contain?

a ☐ y

b ☐   error

c ☐ B A

d ☐ x y C

Question 39

What does table(C,w) contain?

a ☐ B A

b ☐ x y C

c ☐ y

d ☐   error

Question 40

What does table(B,y) contain?

a ☐ ϵ

b ☐ w C

c ☐   error

d ☐ x y C

Question 41

What does table(A,y) contain?

a ☐ w C

b ☐ z B

c ☐   error

d ☐ B A

Question 42

What does table(C,x) contain?

a ☐ B A

b ☐ y

c ☐   error

d ☐ x y C

Question 43

What does table(A,z) contain?

a ☐ z B

b ☐ w C

c ☐ x y C

d ☐   error

Question 44
What does table(B,w) contain?
a ☐ ϵ
b ☐ $x \ y \ C$
c ☐ error
d ☐ $w \ C$

Question 45
What does table(A,x) contain?
a ☐ $z \ B$
b ☐ $B \ A$
c ☐ error
d ☐ $w \ C$

Question 46
What does table(A,w) contain?
a ☐ error
b ☐ $z \ B$
c ☐ $x \ y \ C$
d ☐ $w \ C$

Question 47
What does table(C,z) contain?
a ☐ $B \ A$
b ☐ y
c ☐ $x \ y \ C$
d ☐ error

Question 48
What does table(B,z) contain?
a ☐ $x \ y \ C$
b ☐ $B \ A$
c ☐ error
d ☐ ϵ

Example
Consider the grammar

$$A \ ::= \ y \ B \ x \ \mid \ w \ y \ \mid \ C \ A$$
$$B \ ::= \ \epsilon \ \mid \ x$$
$$C \ ::= \ \epsilon \ \mid \ w \ z \ z \ \mid \ x$$

where $\{A,B,C\}$ is the set of nonterminal symbols, A is the start symbol, $\{w,x,y,z\}$ is the set of terminal symbols, and ϵ denotes the empty string.

Question 49
Which nonterminals are nullable?

a ☐ $\{B, C\}$
b ☐ $\{A\}$
c ☐ $\{C\}$
d ☐ $\{A, C\}$

Question 50
What is Follow(B)?

a ☐ $\{\}$
b ☐ $\{y, z\}$
c ☐ $\{\$, w, x\}$
d ☐ $\{x\}$

Question 51
What is First(B)?

a ☐ $\{x\}$
b ☐ $\{y, z\}$
c ☐ $\{w, x\}$
d ☐ $\{w, x, z\}$

Question 52
What is Follow(A)?

a ☐ $\{\$\}$
b ☐ $\{\$, w, y, z\}$
c ☐ $\{\$, x\}$
d ☐ $\{y\}$

Question 53
Is the grammar LL(1)?

a ☐ No
b ☐ The LL(1)-checker would go into an infinite loop.
c ☐ The question cannot be answered with the information provided.
d ☐ Yes

Question 54
What is First(C)?

a ☐ $\{w, x\}$
b ☐ $\{\}$
c ☐ $\{x, y\}$
d ☐ $\{y\}$

Question 55

What is First(A)?

a ☐ $\{x, z\}$

b ☐ $\{\}$

c ☐ $\{w, y\}$

d ☐ $\{w, x, y\}$

Question 56

What is Follow(C)?

a ☐ $\{y, z\}$

b ☐ $\{w, x, y\}$

c ☐ $\{\$, x, y\}$

d ☐ $\{\$, w, x, z\}$