Theory exam test, December 17, 2020

Due No due date **Points** 20 **Questions** 20
Available Dec 17 at 10am - Dec 17 at 11am about 1 hour **Time Limit** 45 Minutes

Instructions

Instructions for the quiz

Your quiz activity is being logged, please do not change to other windows while filling the quiz. Upon suspicious activities in your log, we may decide to ignore some of your answers.

You are authenticating yourself for the exam by logging into Canvas with your own credentials. Please note that the exam is closed-book, using external material during the test is strictly forbidden. By submitting the quiz, you declare that you worked on your own, nobody helped you and you have not used any external material while filling the quiz.

The theory test consists of **20 multiple-choice questions** which you need to answer in **45 minutes**. You have to solve them in order, one at a time. Please note that you cannot postpone questions or navigate back to previous questions, so do select one of the options for each question. The questions as well as the order of the possible answers are randomized.

Grading scale

Percentage	Points	Grade
90-100	18-20	5
75-89	15-17	4
60-74	12-14	3
45-59	9-11	2
0-44	0-8	1

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	42 minutes	14 out of 20

ⓘ Correct answers are hidden.

Score for this quiz: **14** out of 20
Submitted Dec 17 at 10:51am
This attempt took 42 minutes.

Incorrect

Question 1 0 / 1 pts

Simply speaking, programming languages can be either interpreted (run directly without translation) or compiled (translated to machine code). Which of the following statements is true?

- By definition, compilation has to emit machine code.
- Some languages are compiled to byte-code which is a portable format interpreted by virtual machines.
- Programming languages are specifically designed to be either only interpreted or only compiled, so these techniques cannot be mixed.
- Typically, compilation and interpretation are used as synonyms.

Question 2

1 / 1 pts

Under what restrictions can Chomsky-2 grammars express regular languages?

- The regular language has to be deterministic.
- The regular language has to be finite.
- Under no restrictions can Chomsky-2 grammars define regular languages.
- No restrictions are needed, any regular language can be defined by means of context-free grammars.

Question 3

1 / 1 pts

Which of the following is a subset of the regular language defined by the regular expression **a|ab***?

- {abab}
- {a, aa, aaa, aaaa}
- {a, ab, abbb}
- {a, ab, abab}

Question 4

1 / 1 pts

Which of the following regular expressions matches the string "cba"?

- a*b*c*
- (a|b|c)*
- All of them.
- a*|b*|c*

Incorrect

Question 5

0 / 1 pts

Which of the following regular expression defines a finite language?

- a*b*
- a+b+
- None of them.
- a?b?

Question 6

1 / 1 pts

The lexical analysis follows two basic principles, one of those is that the scanner...

- Matches no whitespaces.
- Matches the simplest lexeme.
- Matches according to token priority.
- Matches no comments.

Incorrect

Question 7

0 / 1 pts

Which of the following is a subset of the regular language defined by the regular expression: a*b*c*

- {a, ab, abc}
- {abc, abcabc}
- {a, ab, abab}
- None of them.

Incorrect

Question 8

0 / 1 pts

Suppose that a grammar contains production rules **S** → **a S b** and **S** → **ε**. Which of the following statements is definitely true?

- **ab** cannot be derived with this grammar.
- The empty string cannot be derived with this grammar.
- The generated language is infinite.
- The generated language is finite.

Question 9

1 / 1 pts

Syntactic analyzers (parsers) produce the syntax tree for programs based on a Chomsky-2 grammar
defining the language syntax. What is the input of parsing?

- A set of tokens.
- A list of characters.
- A set of characters.
- A list of tokens.

Question 10

1 / 1 pts

Bottom-up parsing is based on the idea of finding handles, which are...

- The leftmost simple phrase of the sentential form.
- The shortest simple phrase of the sentential form.
- The longest simple phrase of the sentential form.
- The rightmost simple phrase of the sentential form.

Question 11

1 / 1 pts

Syntactic analyzers (parsers) produce the syntax tree for programs based on a Chomsky-2 grammar defining the language syntax. What is the input of parsing?

- A set of tokens.

- [x] A list of tokens.
- [ ] A set of characters.
- [ ] A list of characters.

Question 12

1 / 1 pts

Parsing yields a syntax tree. Which of the following is true regarding this tree?

- [ ] The leaves are labeled with non-terminal symbols.
- [ ] None of the other answers are true.
- [x] The root is labeled with a non-terminal symbol.
- [ ] The syntax tree is independent of the syntax grammar.

Question 13

1 / 1 pts

How do we store and share the symbol table in BisonC++ semantics analysers?

- [ ] In a synthesised attribute.
- [ ] In a local variable.
- [x] In a global variable.
- [ ] In an inherited attribute.

Incorrect

Question 14

0 / 1 pts

In BisonC++ semantic routines, **$$** refers to the semantic attribute of...

- [x] The last symbol in the production rule.
- [ ] The root node in the syntax tree.
- [ ] The most recently scanned terminal symbol.
- [ ] The root node in the currently built syntax subtree.

Question 15

1 / 1 pts

Semantic analysis checks programs for static semantic errors, such as...

- [ ] Misspelled keywords.
- [ ] All of these.
- [ ] Improper indentation.
- [x] Undeclared variables.

Question 16

1 / 1 pts

Which of the following assembly instructions writes the instruction pointer (IP)?

- [ ] mov
- [x] jmp
- [ ] add
- [ ] push

Question 17

1 / 1 pts

Which of the following registers is changed by the NASM instruction **jz eax**?

- esp
- eax
- ebp
- eip

Question 18

1 / 1 pts

How does **call eax** affect the stack?

- It pushes the method arguments onto the stack.
- It reserves space for the return value.
- It does not affect the stack as it is an invalid instruction.
- It pushes the address of the next instruction onto the stack.

Incorrect

Question 19

0 / 1 pts

Which of the following is true regarding code synthesis?

- Optimisation may change the execution order of high-level instructions.
- Optimisation does not affect target code size.
- Code generation rules are expected to result in optimal code.
- None of the answers are fully correct.

Question 20

1 / 1 pts

Which of the following instructions is guaranteed to zero the value of the EAX register?

- add eax, eax
- xor eax, eax
- and eax, eax
- or eax, eax

Quiz Score: **14** out of 20