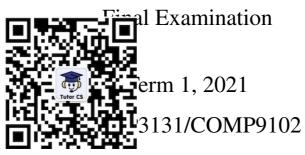
程序代写代做《South Wales辅导



Programming Languages and Compilers

WeChat: cstutorcs

- 1. **Time Allowed:** 3 hours (including 10 min **reading time**)
- 2. Total Number of Pages: 8 (including cover page and Appendix Alphan Help
- 3. Total Number of Questions: 5
- 4. Total Mark Available: 100 torcs@163.com
- 5. Marks available for each question are shown in the examination paper.
- 6. The questions are not of equal subject 76
- 7. Answer all questions.
- 8. Submit your answers via give or Webcms3 mttps://tutorcs.com
- 9. The answers to Q1 can be submitted as jpeg/gif/tiff/png/pdf files but the answers to Q2 Q5 must be submitted as ASCII text files:
 - Q1: *.suffix, where suffix is jpeg, gif, tiff, png, or pdf
 - Q2: *.txt
 - Q3: *.txt
 - Q4: *.txt
 - Q5: *.txt
- 10. No examination materials allowed.

Question 1. Regular Expressions to Finite Automa 程辅导[18 marks]

Consider the following regular expression:



 $(a|b)^*a(a|\epsilon)$

(a) Use **Truction** to convert this regular expression into an NF

[7 marks]

(b) Use the **subset construction** to convert the NFA of (a) into a DFA.

WeChat: cstutorcs

[7 marks]

(c) Convert the DFA of (b) into a minimal-state DFA.

Assignment Project Exam Help Assignment Project Exam Help

You are required to apply exactly Thompson's construction algorithm in (a) and the subset construction algorithm in (b) to solve those two problems.

QQ: 749389476

Accepted files to submit via give or Webcms3: *.jpeg, *.gif, *.tiff, *.png or *.pdf. https://tutorcs.com

Question 2. Context-Free Grammars 做 CS编程辅导[15 marks]

Assume that arithmetic expressions are built up from the following terminals:

- Binary , #, +, -
- Unary j
- Variabl
- Bracke

Operator \sim has the highest precedence, followed by @ and #, which have equal precedence. Operators + and - have the lowest precedence. Operators @, # and + are left-associative but In Instance associative. Brackets are used to group expressions in the usual manner.

Assignment Project Exam Help

Write a context-free grammar for this language.

Email: tutorcs@163.com

You are not allowed to use the regular operators, *, +, and ?, in your grammar.

QQ: 749389476

Accepted file to the file of t

Question 3. Recursive Descent LL(1) Parsing 编程辅导[20 marks]

Consider the following context-free grammar:



WeChat:scstutores

where the set of nonterminals is $\{S, A, B, C, D\}$ and the set of terminals is {a, c, x, f, gAssignment Project Exam Help

(a) Compute the FIRST sets for all nonterminal symbols.

Email: tutorcs@163.com

[4 marks]

(b) Compute the FOLLOW sets for all nonterminal symbols.

OO: 749389476

[6 marks]

(c) Construct the LL(1) parsing table for the grammar.

https://tutorcs.com

[4 marks]

(d) Is the grammar LL(1)? Justify your answer in a few sentences.

[2 marks]

(e) The string qx is NOT syntactically legal (since it is NOT in the language defined by the grammar). Explain concisely how this can be detected by an LL(1) table-driven parser for the language.

[4 marks]

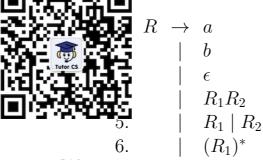
Note: In your answer, you can write E or e as an abbreviation for ϵ .

Accepted file to submit via give or Webcms3: *.txt.

Question 4. Attribute Frammars 代做 CS编程辅导[17 marks]

Consider the following context-free grammar that generates regular expres-

sions:



WeChat: cstut@rcs

(a) Define an attribute grammar that records the maximum number of **nested** Kleene Atar operators of a regular expression E in its attribute R depth. For example, ab has depth 0, a^* has depth 1 and $a^*|(b^*|a)^*$ has depth 2.

[14 marks]

Email: tutorcs@163.com

(b) Is R.depth inherited or synthesized? Explain your answer.

QQ: 749389476

[3 marks]

Note: In your answer, you can write R_1 , R_2 and * as R1, R2 and *, respectively.

Accepted file via **give**: *.txt.

- The ser ciated with a production are evaluated sequentially in the ner, and
 - "#" state to the state of the s

```
S \rightarrow \mathsf{while} \, B \, \mathsf{do} \, S
             S.begin := getNewLabel();
             B.true := getNewLabel();
             B.false := S.next:
S_1.next := V.em
             S_1.next := V 
S_1.next := V 
S_2.code := emit(S.begin' :') # B.code # emit(B.true' :') # S_1.code # emit('goto' S.begin')
S \to \mathbf{ID} = E
             S.code := E.code \#emit(ID.place' := 'E.place)
             E.place :=
             E.code := E_1.code \# E_2.code \# emit(E.place' := 'E_1.place' + 'E_2.place)
             E.place := p.place; // p.place is the lexement the E.code := malno.cold.generateCS of 163.com
B \rightarrow B_1 \&\& B_2
             B_1.true := getNewLabel();
             \begin{array}{l} B_1. false := gB. false; \\ B_2. true := B_1. false; \\ B_2. false := B_2. false; \\ B_2. false := B_1. code \# emit(B_1. true':') \# B_2. code \end{array}
             B_1.true := B.false;

B_1.false := B_1.true

B.code := B_1.code
                                             ://tutorcs.com
B \to \mathbf{ID}_1 > \mathbf{ID}_2
             B.code := emit('if' | \mathbf{ID}_1.place > \mathbf{ID}_2.place 'goto' B.true) \# emit('goto' B.false)
```

Note that this grammar is ambiguous but that does not affect the following questions.

Consider the following while loop:

while (!
$$(a > b \&\& x > y)$$
)
 $r = p + q$;

(a) Draw the AST (Abstract Syntax Tree) for the **while** loop. [5 marks]

Continued onto next page

Question 5 continued from Page 6 CS编程辅导

(b) Suppose that S nert = L666, getNewLabel() will return labels L1, L2, . . L666, getNewTemp() will return temporary variables the state of the state o

Give t f(a) of f(a).

.false attributes for all the B nodes in the AST

[7 marks]

(c) Give the S.code attribute for the root node S in the AST of (a). In other words, give the code generated for the while loop according to this attribute grammar.

[7 marks]

(d) The production B_1 B_2 B_3 B_4 B_5 B_5 B_5 B_5 B_5 B_5 B_6 B_6 B_6 B_7 B_8 so that conditional QR expressions are considered instead. Give the semantic rules for the new production to generate short-circuit code for conditional QR expressions.

QQ: 749389476

[5 marks]

(e) Give the semantic rules for the new production that defines do-while statements:

In a do-while statement, S_1 will be executed at least once.

[6 marks]

Note: In your answer, you can write S_1 , B_1 and B_2 as S_1 , B_1 and B_2 , respectively.

Accepted file to submit via give or Webcms3: *.txt.

```
Appendix A. Josmin assembly (i.e., IVM) instructions for Question 5. 在方式与代数器 推辑字
public final class JVM {
    // Arithmetic instructions
    IADD = "iadd"
    ISUB =
    IMUL =
    IDIV =
    // Load:
                           f Llocal variable into operand stack)
    ILOAD = | |
    ILOAD_0
    ILOAD_1
    ILOAD_2 = "iload_2",
    ILOAD_3 = "iload_3",
    // Store of store the Soph Offic Sn operand stack into a local variable)
    ISTORE = "istore",
    ISTORE 0 = "istore 0",
   ISTORE_1 = "istore_1", istore_2 A sister ment Project Exam Help ISTORE_3 = "istore_3",
    // Loads (for loading a constant into operand stack)
   ICONST_0 = "iconst_0", iconst_0",
    ICONST_1 = "iconst_1",
    ICONST_2 = "iconst_2"
    ICONST_4 (1):onst_4 9389476
    ICONST_5 = "iconst_5",
    // Contibute Store Com
    IFEQ = "ifeq",
   IFNE = "ifne",
   IFLE = "ifle",
   IFLT = "iflt",
   IFGE = "ifge",
   IFGT = "ifgt",
   IF_ICMPEQ = "if_icmpeq",
   IF_ICMPNE = "if_icmpne",
   IF_ICMPLE = "if_icmple",
    IF_ICMPLT = "if_icmplt",
    IF_ICMPGE = "if_icmpge",
    IF_ICMPGT = "if_icmpgt",
    // Operand stack management instructions
   DUP_X2 = "dup_x2",
   DUP = "dup",
   POP = "pop",
    . . .
}
```

Page 8 of 8

============= END OF PAPER ===============