Compilers [Spring 2024]
Test III

NAME: _____

Instructions:

1) This test is 12 pages in length.

2) You have 120 minutes to complete and turn in this test.

3) Short-answer and essay questions will be graded on how clearly you have communicated the necessary ideas. Respond in complete English sentences. Avoid using bullet points and enumerated lists. Respond at the level of detail discussed in class.

4) All grammars must contain (a) no circular rules of the form N::=N, (b) no syntactic categories lacking a base case (i.e., every syntactic category must generate at least one string), (c) no repeated rules, and (d) exactly one production for the starting nonterminal S having the form S::=N$.

5) Use the same notations, assumptions, invariants, etc. that we have been using in class.

6) This test is closed everything, including books, notes, papers, computers, phones, laptops, smartwatches, smartglasses, AI, LLMs, friends, neighbors, etc.

1. [4 points] [1 sentence] What are 8 standard code optimizations, as defined in class?

2. [8 points] [Short essay] Hitting all the main points discussed in class, and considering program memory, describe the standard memory segments and all the sorts of values that are stored in those segments.

3. [7 points] [Short essay] Discuss the "division of labor" during a method invocation, as implemented by djc, hitting all the main points discussed in class. Which parts of the target code have which responsibilities?

4. [3 points] [1-3 sentences] Compare and contrast the strong and weak generational hypotheses.

5. [5 points] [Short essay] Describe the mark-and-compact algorithm.

6. [3 points] Are computable languages closed under subset? Explain.

7. [10 points] Draw a vtable for the following DJ program.

```
class c1 extends c3 { nat m3(nat n){0;} nat m1(nat n){n;} nat m5(nat n){1;}}
class c2 extends c4 { nat m5(nat n){n;} nat m1(nat n){1;} nat m2(nat n){0;}}
class c3 extends Object { nat m1(nat n){1;} nat m2(nat n){0;} nat m3(nat n){n;}}
class c4 extends c1 { nat m4(nat n){0;} nat m2(nat n){n;} nat m3(nat n){1;}}     main{0;}
```

8. [8 points] Define an epsilon-free CFG such that its LR(0), SLR, LALR, and LR(1) tables are identical and conflict free, and prove it.

9. [8 points] Define an epsilon-free CFG such that its LR(0), SLR, LALR, and LR(1) tables are identical and contain a conflict, and prove it.

10. [10 points]
Define an algorithm to decide whether two input NFAs are equivalent. For 60% credit, you may define the algorithm we discussed for a similar problem on Test II. For full credit, define a different algorithm from the one we previously discussed for Test II—i.e., define an algorithm that avoids performing a state-by-state equivalence analysis.

11. [10 points]
Prove that the following language is undecidable. Do not use Rice's Theorem.
{ <M,s> | TM M, on input s, never writes a 0 on its tape}

12. [24 points] [Essay]
Suppose we wish to add repeat-until expressions to DJ. An example repeat-until expression is: `repeat {x=x+2; y=y+3;} until (x<y)`. Such expressions are similar to, but not the same as, do-while loops in languages like C and Java.

Describe in detail the modifications you would make to djc, to allow source programs to use repeat-until expressions. Maintain consistency with how DJ and djc handle similar constructs. The next page is blank, in case you would like to continue your essay there.

[This page provides optional additional space for Problem 12.]

Undergraduates stop here. The remaining problem is for graduate students.

13. [8 points]

Prove that the following language is undecidable. Do not use Rice's Theorem.

{ <M> | TM M diverges on all inputs}