

## Kuis 2:Teori Bahasa dan Automata (Kelas A-C)

23 Mei 2022 (15.00 – 16.30)

Dari Kelas C

1. Jika  $L$  adalah suatu bahasa DCFL maka bisa jadi ada FSM yang dapat menerimanya.

**Benar.** Bahasa DCFL adalah superset dari bahasa regular. Jadi setiap bahasa regular adalah DCFL. Sudah jelas setiap bahasa regule dapat diterima suatu FSM.

2. Jika  $L_1$  dan  $L_2$  adalah dua bahasa DCFL maka konkatenasi kedua bahasa itu (atau  $L_1L_2$ ) adalah juga DCFL.

**Salah.** Counter example dengan dua bahasa ini:  $\{a^nb^m : m \neq n\}$  dan  $\{b^pa^q : p \neq q\}$  adalah DCFL, tapi jika diconcate menjadi bahasa  $a^nb^ra^q : m, r, q \geq 0$  yang bahkan merupakan inherently ambiguous, misalnya abbba yang bisa merupakan concate dari ab dan bba atau abb dan ba, sehingga parse-tree string tsb lebih dari satu.

3. Jika  $L_1$  adalah bahasa regular dan  $L_2$  adalah bahasa CFL, maka konkatenasi kedua bahasa itu (atau  $L_1L_2$ ) adalah juga CFL.

**Benar.** Setiap regular adalah CFL sementara menurut sifatnya CFL closure terhadap operasi konkatenasi.

4. Jika  $L_1$  adalah bahasa D dan  $L_2$  adalah bahasa SD, maka mungkin saja konkatenasi kedua bahasa itu (atau  $L_1L_2$ ) adalah juga D.

**Benar.** Jika ternyata  $L_2$  adalah juga D maka  $L_1L_2$  adalah D.

5. Jika  $L_1$  adalah bahasa D dan  $L_2$  adalah bahasa SD, sementara  $L_2 \subset L_1$ , maka konkatenasi kedua bahasa itu (atau  $L_1L_2$ ) adalah juga D.

**Salah.** Misalnya mesin Turing  $M_1$  untuk  $L_1$  dan  $M_2$  untuk  $L_2$ . Jika kedua mesin dirangkaikan menjadi  $M_1M_2$  untuk mengenali  $L_1L_2$  maka saat memeriksa string  $w_1w_2$ , jika  $w_1$  bukan anggota  $L_1$  secara keseluruhan proses bisa tidak dapat masuk (karena infinite loop) ke bagian  $M_2$  untuk memeriksa  $w_2$ . Dengan demikian, belum tentu D (tapi sudah pasti SD).

6. Pernyataan “setiap bahasa yang tidak decidable disebut non-decidable” menyatakan komplemen dari suatu kelas bahasa yang decidable adalah non-decidable (atau undecidable).

**Benar.** Jika tidak decidable berarti adanya string  $w$  mungkin anggota  $L$  atau tidak yang dapat membuat mesin Turing infinite-loop. (Note: jika hanya  $w$  yang bukan anggota  $L$  saja yang menyebabkan infinite-loop sementara jika anggota  $L$  selalu halt disebut semi-decidable).

7. Bahasa  $\{a^nb^{2n}c^{3n} : n \text{ bilangan bulat}\}$  dapat diterima oleh suatu PDA yang memiliki stack ganda}.

**Benar.** PDA dengan stack ganda (atau double stack) setara dengan Mesin Turing dan Mesin Turing dapat menerima bahasa tersebut.

8. Jika  $L$  adalah bahasa semidecidable (SD) tapi terdapat mesin Turing yang dapat mengenumerasikan anggota-anggota bahasa itu secara proper order, maka komplemennya pasti decidable (D).

Benar. Kemampuan melakukan enumerasi secara proper-order menunjukkan mesin bisa selalu halt-yes atau halt-no untuk setiap kemungkinan input (halt-yes jika anggota  $L$  dan halt-no jika bukan).

Dari Kelas A & C

11. Jika  $L$  adalah suatu bahasa non-inherently ambiguous CFL maka bisa jadi ada FSM yang dapat menerimanya.

**Benar.** Non-inherently ambiguous adalah superset dari DCFL dan DCFL superset dari bahasa regular sehingga bisa saja  $L$  mrpk bahasa regular.

12. Jika  $L_1$  dan  $L_2$  adalah dua bahasa DCFL maka union kedua bahasa itu (atau  $L_1 \cup L_2$ ) adalah juga DCFL.

**Salah.** DCFL tidak closure pada operasi union.

13. Jika  $L_1$  adalah bahasa regular dan  $L_2$  adalah bahasa CFL, maka union kedua bahasa itu (atau  $L_1 \cup L_2$ ) adalah juga CFL.

**Benar.**  $L_1$  sebagai bahasa regular juga dalah bahasa CFL, dan CFL closure pada operasi union.

14. Jika  $L_1$  adalah bahasa D dan  $L_2$  adalah bahasa SD, maka mungkin saja union kedua bahasa itu (atau  $L_1 \cup L_2$ ) adalah juga D.

**Benar.** (Perhatikan kata “mungkin saja” bukannya “pasti selalu”) Kalau  $L_2$  ternyata D (sebagai SD bisa juga D), maka union keduanya bisa D.

15. Jika  $L_1$  adalah bahasa D dan  $L_2$  adalah bahasa SD, sementara  $L_2 \subset L_1$ , maka union kedua bahasa itu (atau  $L_1 \cup L_2$ ) adalah juga D.

**Benar.** Kalau  $L_2 \subset L_1$  maka  $L_1 \cup L_2 = L_1$ . Dengan sendirinya karena  $L_1 \cup L_2 = L_1$  adalah D.

16. Jika untuk suatu bahasa  $L$ , terdapat mesin Turing yang dapat meng-enumerasi string-string anggota bahasa tersebut, maka mesin itu akan selalu halt untuk setiap string apapun yang diberikan.

**Salah.** Enumerasi tersebut tidak disebutkan bersifat “proper-order”. Jadi kalau hanya enumerasi saja aka bahasa SD memenuhi hal ini. Sementara, string-string yang tidak masuk

dalam enumerasi (artinya bukan string anggota  $L$ ) bisa jadi membawa mesin menjadi infinite-loop (tidak halt).

17. Mesin Turing single-tape hanya bisa menerima bahasa yang diterima oleh Mesin Turing double-tape saja, sementara mesin Turing dengan jumlah tape 10 memerlukan tape yang jumlahnya lebih dari 1 untuk mensimulasikan bekerjanya kesepuluh tape itu.

**Salah.** Berapa banyak tape yang digunakan tetap bisa diemulasi (ditiru) oleh mesin Turing 1 tape (bukan hanya 2 tape!).

18. Jika  $L$  adalah bahasa non-decidable ( $\neg D$ ) maka  $\neg L$  (komplemen dari  $L$  tersebut) adalah juga bahasa non-decidable ( $\neg D$ ).

**Salah.** Bahasa-bahasa  $D$  bersifat closure pada operasi komplemen.