

```

if (next == ' +') { /* rule  $F \rightarrow (+DN)$  */
    op = readnumber;
    if (next != ')') ERROR;
    next = getnext;
    return op;
}
else { /* rule  $F \rightarrow (-DN)$  */
    if (next == ' -') {
        next = getnext;
        op = readnumber;
        if (next != ')') ERROR;
        next = getnext;
        return -1 * op;
    }
    else { /* rule  $F \rightarrow (E)$  */
        op = E;
        if (next != ')') ERROR;
        next = getnext;
        return op;
    }
}
}
else ERROR;
}

```

5.10 Kontekstittomien kielten ominaisuuksia

5.10.1 Kontekstittomien kielten sulkeumaominaisuudet

Kontekstittomille kielille pätee joitain samantapaisia sulkeumaominaisuuksia kuin säännöllisille kielille.

Lause 5.5 Olkoon L_1 ja L_2 kontekstittomia kieliä. Tällöin myös

1. $L_1 \cup L_2$ (kielten yhdiste)
2. $L_1 L_2$ (kielten katenaatio)
3. $(L_1)^*$ (kielen sulkeuma)
4. $(L_1)^R$ (kielen käänteiskieli)

ovat kontekstittomia.

Huom! Kontekstittomat kielet eivät kuitenkaan ole suljettuja leikkauksen ja komplementin suhteen! Jos L_1 ja L_2 ovat kontekstittomia, ei $L_1 \cap L_2$ silti ole välttämättä kontekstiton! Samoin $\bar{L}_1 = \Sigma^* \setminus L_1$ (kielen komplementti) ei välttämättä ole kontekstiton.

Esim. kielet $L_1 = \{a^n b^n c^k | n, k = 0, 1, \dots\}$ ja $L_2 = \{a^k b^n c^n | k, n = 0, 1, \dots\}$ ovat kontekstittomia. Kuitenkaan kieli $L_1 \cap L_2 = \{a^n b^n c^n | n = 0, 1, \dots\}$ ei ole kontekstiton!

Toisaalta, jos L on kontekstiton kieli ja R on säännöllinen kieli, niin $L \cap R$ on kontekstiton. (Ks. esim. Hopcroft, Motwani, Ullman, teoreema 7.27.)

5.10.2 Ratkeavat ja ratkeamattomat ominaisuudet

Kontekstittomia kieliä itseään koskevat ongelmat eivät suinkaan kuulu kontekstittomien kielten luokkaan. Edellä olemme jo oppineet joitain kontekstittomien kielten ominaisuuksia, joita voidaan ratkaista tietokoneella. Tärkein tällainen ominaisuus koskee jäsentämistä: mille tahansa merkkijonolle x ja annetulle kontekstittomalle kieliopille G voimme ratkaista, kuuluuko x kieleen $L(G)$. Ts. ongelma $x \in L(G)$ on algoritmisesti ratkeava. Kulloinkin paras jäsennyksen menetelmä riippuu annetusta kielestä. Jos kielioppi voidaan esittää LL(1)-muodossa, tarjoaa tämä tehokkaan (ajassa $O(n)$ toimivan) ja helpon jäsennyksen menetelmän. Ohjelmointikielten kääntäjät puolestaan käyttävät LR(k)-jäsentäjiä, kuten Unixin yacc- ja bison-ohjelmien tuottamia jäsentäjiä. Mielivaltaisen kieliopin taas voimme aina muuntaa Chomskyn normaaliin muotoon ja jäsentää CYK-algoritmillä ajassa $O(n^3)$.

Eräät tärkeistä kontekstittomien kielten ominaisuuksista ovat kuitenkin ratkeamattomia ongelmia. Tällaisia ongelmia ovat:

- Kahden kieliopin ekvivalenssi. Ts. onko $L(G_1) = L(G_2)$, missä G_1 ja G_2 ovat kontekstittomia kielioppeja?