Kush Shah
Acknowledgment: Keith DeSantis,
Sam Parks

b) $a^nb^mc^md^{2n}$ with n, m any natural numbers

S→ aSdd | F | λ
F→ bFc | λ

Explanation: aSdd ensures that there are twice as many d as a and bFc ensures that the number of b and c is the same.

c) a^nb^m with 0 <= n <= m <= 2n

S→ aSb | aSbb | λ

Explanation: aSbb ensures that the maximum number of b is always twice the number of a but aSb allows for any number of b less than 2n.

d)$a^mb^nc^k$ where (m=n) or (m=k)

S → T | U
T → aTbC | λ
C→ cC | λ
U → aUc | B | λ
B → bB | λ

Explanation: This grammar is the Union of a^mb^m and a^mc^m.

e) $\{a^ib^ic^kd^k \mid i,k >= 0\}$

S→ TU | λ
T→ aTb
U→ cUb

Explanation: This grammar is the concatenation of a^ib^i and c^kd^k .

f) $\{a^ib^jc^kd^m \mid i,j,k,m >= 0$, and $(i = j$ or $k = m)\}$

S→ F | Q

F→ TU
T→ aT | Tb | λ
U→ cUd | λ

Q→ NM
N→ aNb | λ
M→cM | Md | λ

Explanation: This grammar is the concatenation $a^ib^j c^kd^m$ where i could equal j or k = m.

199. ArithAmb1

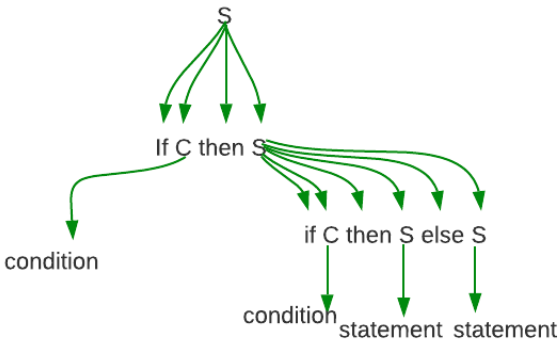
a)
E → E + T | T
T → T * I | I
I → a | b | c

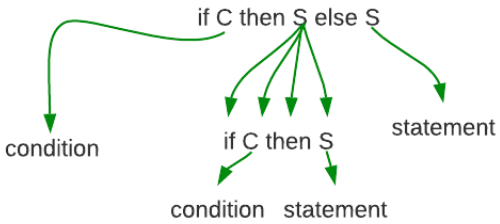
b)
E → E + T | E – T |T
T → T * I | I
I → a | b | c

c)
E → E + T | E – T |T
T → T * U | U
U → I ↑ U | I
I → a | b | c
d)
C → C = E | C < E | E
E → E + T | E – T |T
T → T * U | U
U → I ↑ U | I
I → a | b | c

203. PLAmb

1)





Both trees represent the string “if condition then if condition then statement else statement”

2) if (1 > x) {
    If (2 < x) {
        print(“in second if”);
    } else {
        print(“In first if”);

```
        }
        OR

If (1 > x){
        If(2<x){
        print("in second if");
        }
} else{
        print("in neither if");
}
```

3)
S → if C then S | if C then T else S | statement
T → if C then T else T | statement
C → condition

The above grammar resolves the issue of dangling else.

209. ElimUseless

S → dS | aS | a

217. GNFPractice

E → 0*T + E | 1*T + E | 0 + E | 1 + E |
T → 0*T | 1*T | 0 | 1