1 CFLs

note 1: $\Sigma = \{a, b\}$
note 2: $n_x(a)$ means the number of a's in the string x.

- $\{x|n_x(a) = n_x(b)\}$
  corresponding CFG :

$$S \to aSbS|bSaS|\epsilon$$

- $\{x|n_x(a) \geq n_x(b)\}$
  corresponding CFG:

$$S \to SS|aSb|bSa|a|\epsilon$$

  not surely right version :

$$\begin{aligned} S &\to SaS|T \\ T &\to aTbT|bTaT|\epsilon \end{aligned}$$

- $\{x|n_x(a) = 2n_x(b)\}$
  corresponding CFG:

$$S \to SaSaSbS|SaSbSaS|SbSaSaS|\epsilon$$

  using similar method, we can easily generalize this into k.
  important lemma needed in the proof: substring $a^x ba^y$, $(x + y = k)$ can always be found in any non-empty string s in the language.

- complement of $a^n b^n$
  corresponding CFG:

$$\begin{aligned} S &\to aSb|bY|Ya \\ Y &\to bY|aY|\epsilon \end{aligned}$$

- $\left\{a^i b^j | i \neq j \wedge i \neq 2j\right\}$
  corresponding CFG(not surely right):

$$\begin{aligned} S &\to A|B|N \\ E &\to aEb|\epsilon \\ A &\to Eb|aAb|Ab \\ B &\to aaabb|aBb|aaBb \\ M &\to aaMb|\epsilon \\ N &\to aM|aNb|aN \end{aligned}$$

- $\left\{a^i b^j | i \leq j \leq 2i\right\}$
  With the idea of previous problem, it's easy to construt a CFG to recognize this language.

  But it's also elegant to use PDA to solve this problem.

  While PDF read 'a', it randomly push one 'a' or two 'a's into stack and pop two 'a's while reading 'b'.And PDA accepts while stack is empty.

  Using this idea, it's also easy to show $\{x|n_x(a) \leq n_x(b) \leq 2n_x(a)\}$ is CFL.

- $\{xyz|x,\ z \in \Sigma^\star,\ y \in \Sigma^\star b \Sigma^\star \ and \ |x| = |z| \geq |y|\}$
  corresponding CFG (not surely right):

$$\begin{aligned} S &\to A|B|C \\ A &\to EEEbEE|EAE|EEAE \\ B &\to EEbEEE|EBE|EBEE \\ C &\to EbE|ECE \\ E &\to a|b \end{aligned}$$

- convert NFA/DFA into CFG
  let D = $(Q, \Sigma, \delta, q_0, F)$ be a NFA, and WLOG, $F = \{p\}$.

  let C = (V, T, R, S) be a CFL, and

  - $V = Q$
  - $T = \Sigma$
  - $S = q_0$
  - $\forall x, y \in Q, c \in \Sigma, y \in \delta(x, c)$, insert "$x \to cy$" into R
  - insert "$p \to \epsilon$" into R

  Also, one can construt D = (V', T', R', S') be a CFL, and

  - $V' = Q'$
  - $T' = \Sigma$
  - $S' = p$
  - $\forall x, y \in Q, c \in \Sigma, y \in \delta(x, c)$, insert "$y \to xc$" into R'
  - insert "$q_0 \to \epsilon$" into R'

  Using similar idea, it's obvious that if R in arbitrary CFG contains only transitions of type $A \to cB$ where $A, B \in V, c \in T^*$or only transitions of type $A \to Bc$, this CFG generates regular language.

  But if R contains both type $A \to cB$ and type $A \to Bc$, thus this CFG might generate CFL.

  For example, consider this CFG

  $$S \to Bb|\epsilon$$
  $$B \to aS$$

  This CFG generates CFL(hint: use pumping lemma of regular language).
  But I(yyx) haven't figured out whether CFGs contains only these two types transitions can generate all CFLs or not. I(yyx) guess the answer is NO.