(解答) (a) 次で定義される文脈自由文法 $G_0 = (V_0, T, P_0, S)$ は $\{0,1\}$ 上の正則表現を定義する。

$$V_0 = \{E\} \qquad T = \{0, 1, (,), +, *, \phi\} \qquad S = E$$
$$P_0 : E \rightarrow 0|1|\phi|(E+E)|(EE)|(E^*)$$

(b)G に対し、新たな文法 $G = (V, T, P, S)$ を次のように定義する。
生成規則中の $0, 1, \phi$ 以外の全ての終端記号 a に対してそれらを変数 C_a に置き換え、生成規則 $C_a \rightarrow a$ を導入する。さらに、生成規則 $A \rightarrow B_1 \cdots B_n$($B_i$ は変数) の代わりに、新たな変数 $D_1 \cdots D_{n-2}$ と以下の生成規則

$$\begin{cases} A \rightarrow & B_1 D_1 \\ D_1 \rightarrow & B_2 D_2 \\ \vdots & \\ D_{n-3} \rightarrow & B_{n-2} D_{n-2} \\ D_{n-2} \rightarrow & B_{n-1} B_n \end{cases}$$

を導入する。
このように定義された文法 $G = (V, T, P, S)$ に対して $L(G_0) = L(G)$ であり、さらに G は Chomsky 標準形である。
(b) さて、記号列 $e = x_1 x_2 \cdots x_n$ が $\{0,1\}$ 上の正規表現であるかどうかは、「記号列 e が G から生成されるか」という文脈自由文法の所属性判定問題として定式化でき、これは次のようなアルゴリズムで判定することができる。

begin
  for $i \leftarrow 1$ **to** n **do**
    $V_{i1} \leftarrow \{A | A \rightarrow a \in P, x_i = a\}$;
  for $j \leftarrow 2$ **to** n **do**
    for $i \leftarrow 1$ **to** n **do**
      begin
        $V_{ij} \leftarrow \phi$;
        for $k \leftarrow 1$ **to** $j-1$ **do**
          $V_{ij} \leftarrow V_{ij} \cup \{A | A \rightarrow BC \in P, B \in V_{ik}, C \in V_{i+k, j-k}\}$
      end;
  if $S \in V_{1n}$ **then** $w \in L(G)$ **else** $w \in L(G)$
end

(c) さて、上のアルゴリズムにより入力の e が正規表現でないと判断されれば、題意のアルゴリズムは”no”を出力すればよい。e が正規表現であるならば、e が表現する言語を受理する ε 動作付非決定性有限オートマトン（以下 ε-NFA と呼ぶ）M_e を以下の定義に従って構成できる。

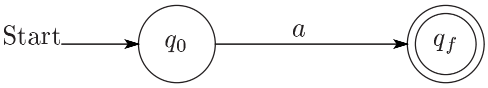
ある正規表現 ψ を受理する ε-NFA を、ψ の構成に関して以下のように帰納的に定義する。
ここで、以下に従って構成される各 ε-NFA には、初期状態へ入る状態遷移がなく、（唯一の）受理状態から出る状態遷移も無いことに注意する。
各有限オートマトンはダイアグラムで表現するが、q_0 が初期状態、q_f が受理状態を表すものとする。
(1)$\psi = \phi$ のとき
次の ε-NFAM_1 はどんな記号列も受理せず、$L(M_1) = \phi$ である。

M_1

(2)$\psi = a \quad (a \in \{0,1\})$ のとき

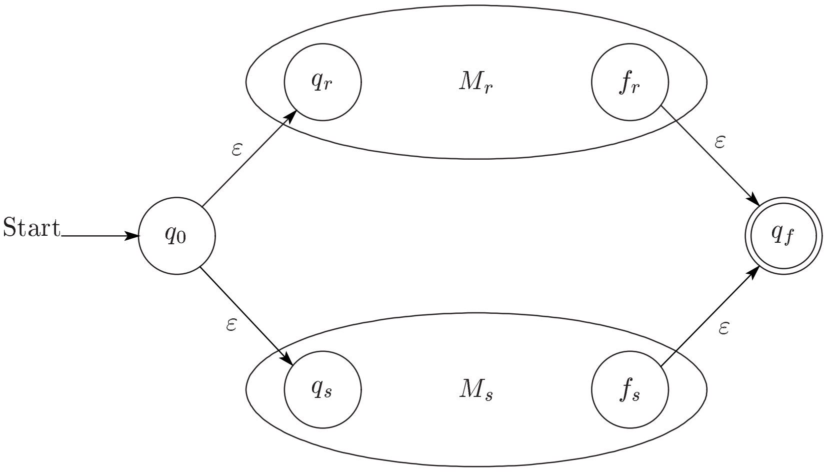
次の ε-NFAM_2 は記号列 a のみ受理し、$L(M_2) = a$ である。

M_2



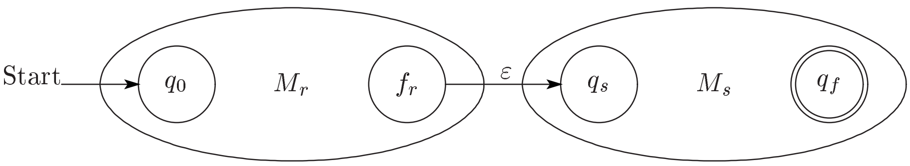
(3)$\psi = r + s \quad (r, s$ は正規表現) のとき
次に示す ε-NFA M_3 を考える。
つまり、r, s を受理する ε-NFA M_r, M_s に新たな初期状態 q_0 と受理状態 q_f を導入し、
q_0 から M_r, M_s の初期状態への ε 動作と、M_r, M_s の受理状態から q_f への ε 動作を付け加える。
$x \in r$ ならば、最初に ε 動作で M_r の初期状態に行き、最後に M_r の受理状態から q_f に ε 動作で行けばよい。$x \in s$ も同様である。逆に、$x \notin r$ かつ $x \notin s$ ならば受理されないので、$L(M_3) = r + s$ である。

M_3



(4)$\psi = rs \quad (r, s$ は正規表現) のとき
次に示す ε-NFA M_4 を考える。
つまり、r, s を受理する ε-NFA M_r, M_s を考え、M_r の初期状態を M_4 の初期状態に、M_s の受理状態を M_4 の受理状態とする。そして、M_r の受理状態から M_s の初期状態への ε 動作を付け加える。
この M_4 は $L(M_r)L(M_s)$ を受理するので、$L(M_4) = rs$ である。

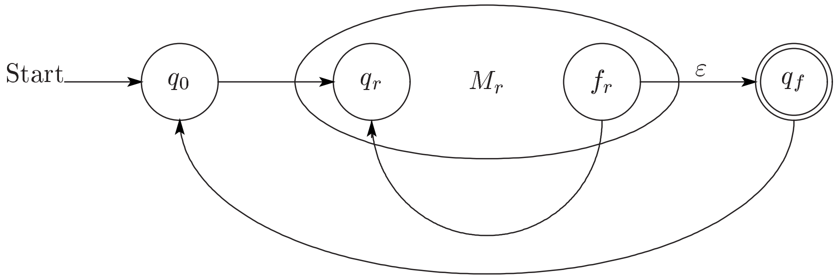
M_4



(5)$\psi = r^* \quad (r$ は正規表現) のとき
次に示す ε-NFA M_5 を考える。
つまり、r を受理する ε-NFA M_r に新たな初期状態 q_0 と受理状態 q_f を導入し、
q_0 から M_r の初期状態への ε 動作、M_r の受理状態から q_f への ε 動作、M_r の初期状態から q_f への ε 動作、そして M_r の受理状態から M_r の初期状態への ε 動作を付け加える。
$x = \phi$ なら最初に ε 動作で受理状態に行けばよく、$x = w^n (w \in r, n = 1, 2, \cdots)$ なら M_r に行き、w を n 回読んだ後に受

理状態に行けばよい。

M_5



さて、以上の方法を e に適用して得られた ε-NFA $M = (Q, \Sigma, \delta, q_0, F)$ に対し、決定性有限オートマトン $\widehat{M} = (\widehat{Q}, \Sigma, \widehat{\delta}, \widehat{q_0}, \widehat{F})$ を

$$
\begin{aligned}
\widehat{Q} &= \{S | S \subseteq Q\} \\
\widehat{q_0} &= \{q_0\text{から}\varepsilon\text{動作だけで到達可能な状態全体}\} \\
\widehat{F} &= \{S \subseteq Q | S \cap F \neq \phi\} \\
\widehat{\delta}\text{は}S &\in \widehat{Q}, a \in \Sigma\text{に対し、} \\
S(a) &= \{q \in Q | \exists p \in S, (p, a, q) \in \delta\}\ \text{とおいて} \\
\widehat{\delta}(S, a) &= \{q \in S(a)\ \text{から}\varepsilon\text{動作だけで到達可能な状態全体}\}
\end{aligned}
$$

で定義すると、$L(M) = L(\widehat{M})$ となる。

次に、このオートマトンの状態数を次のようにして最小化する。

まず、

$$Marked = \{(p, q) | p \in F, q \in Q - F\}, UnMarked = F \times F \cup Q - F \times Q - F$$

とし、「(☆) ある $a \in \Sigma$ とある $(p, q) \in UnMarked$ が存在して $(\delta(p, a), \delta(q, a)) \in Marked$ となる」ならば、

$$Marked = Marked \cup \{(p, q)\}, UnMarked = UnMarked - \{(p, q)\}$$

とする。(☆) が成り立たなくまでこれを繰り返し、最終的に $UnMarked$ に残っている状態をまとめて 1 つの状態にすることで得られたオートマトンは状態数が最小のオートマトンとなっている。これを $\widetilde{M}$ とする。

(d)(2008) 言語 $\{0, 1\}^*$ を受理する最小オートマトンは、状態数が 1 つでそれが初期状態かつ受理状態になっているものである。

よって、$\widetilde{M}$ の状態が 1 つでそれが受理状態であれば"yes"を、そうでなければ"no"を出力する。

(d)(2007) 言語 ϕ を受理する最小オートマトンは、状態が 1 つの初期状態のみで、受理状態をもたないものである。

よって、$\widetilde{M}$ の状態数が 1 つでそれが受理状態でなければ"yes"を、そうでなければ"no"を出力する。