第 1 章 正则语言

1. 有穷自动机DFA是一个 5 元组$(Q, \Sigma, \delta, q_0, F)$；
2. 若A是机器M接受的全部字符串集，则称A是机器M的语言，记作$L(M) = A$；又称M识别A或M接受A；
3. 如果机器不接受任何字符串，那么它识别空语言Φ；
4. 如果一个语言被一台有穷自动机识别，则称它是正则语言；
5. 正则语言类在并、连接、星号运算下封闭（P36）；
6. NFA中如果子过程中至少有一个接受，那么整个计算接受；
7. 非确定型有穷自动机NFA是一个 5 元组$(Q, \Sigma, \delta, q_0, F)$；
8. 每一台NFA都等价于某一台DFA（P33）；
9. 一个语言是正则的，当且仅当有一台NFA识别它；
10. 把空集连接到任何集合上得到空集：$1^*\Phi = \Phi,\ \Phi^* = \{\varepsilon\}$；
11. 一个语言是正则的，当且仅当可以用正则表达式描述它（P40）；
12. 广义非确定性有穷自动机$GNFA$是一个 5 元组$Q, \Sigma, \delta, q_{start}, q_{accept}$；
13. 泵引理：若A是一个正则语言，则存在一个数p（泵长度）使得，如果s是A中任一长度不小于p的字符串，那么s可以被分成 3 段，$s = xyz$，满足下述条件：
    1) 对每一个$i \geq 0$，$xy^iz \in A$；
    2) $|y| > 0$；
    3) $|xy| < p$；
14. 正则语言类在交、补、反转运算下封闭；
15. 设x和y是两个字符串，L是一个语言。如果存在字符串z，使得xz和yz中恰好有一个是L的成员，则称x和y是用L可区分的；否则，对每一个字符串z，xz和yz要么都是、要么都不是L的成员，则称x和y是用L不可区分的，记作$x \equiv_L y$；
16. Myhill-Nerode 定理：设L是一个语言，X是一个字符串集合。如果X中的任意两个不同字符串都是用L可区分的，则称X是用L两两可区分的；定义L的指数为用L两两可区分的集合中的元素个数的最大值；L的指数可能是有穷的或无穷的；
    L是正则的当且仅当它有有穷的指数，它的指数是识别它的最小的 DFA 的大小。

第 2 章 上下文无关文法

17. 上下文无关文法CFG是一个 4 元组(V, Σ, R, S)；
18. 如果字符串w在上下文无关文法G中有两个或两个以上不同的最左派生，则称G歧义地产生字符串w，如果文法G歧义地产生某个字符串，则称G是歧义的；
19. 称一个上下文无关文法为乔姆斯基范式，如果它的每一个规则具有如下形式：

$$A \to BC$$

$$A \to a$$

其中，a是任意的终结符，A、B和C是任意的变元，且B和C不能是起始变元。此外，允许规则$S \to \varepsilon$，其中S是起始变元；

20. 任一上下文无关语言都可以用一个乔姆斯基范式的上下文无关文法产生（P67）；
21. 确定型下推自动机与非确定型下推自动机在语言识别能力上不相同；
22. 非确定型下推自动机等价于上下文无关文法；
23. 下推自动机PDA是 6 元组$(Q, \Sigma, \Gamma, \delta, q_0, F)$；
24. 一个语言是上下文无关的，当且仅当存在一台下推自动机识别它（P72）；
25. 每一个正则语言都是上下文无关的；
26. 关于上下文无关语言的泵引理：如果A是上下文无关语言，则存在数p（泵长度），使得A中任何一个长度不小于p的字符串s都能被划分成 5 段$s = uvxyz$且满足下述条件：
    1) 对于每一个$i \geq 0$，$uv^i xy^i z \in A$；
    2) $|vy| > 0$；
    3) $|vxy| \leq p$；
27. $D = \{ww | w \in \{0,1\}^*\}$不是$CFL$；
28. 确定型下推自动机（P79，略）；
29. 上下文无关语言类在并、连接、星号运算下封闭，在交、补运算下不封闭；

第 3 章 丘奇-图灵论题

30. 图灵机进入拒绝和接受状态将立即停机；
31. 图灵机TM是一个 7 元组$(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$；
32. 当前状态、当前带子内容和读写头当前位置组合在一起称为图灵机的格局；
33. M接受的字符串的集合被称为M的语言，或被M识别的语言，记为$L(M)$；
34. 如果一个语言能被某一图灵机识别，则称该语言是图灵可识别的；

35. 对所有输入都停机的图灵机被称为判定器；对于可以识别某个语言的判定器，称其判定该语言；
36. 如果一个语言能被某一图灵机判定，则称它是图灵可判定的，简称可判定的；
37. 为证明两个模型是等价的，只要证明它们能相互模拟即可；
38. 每个多带图灵机等价于某一个单带图灵机（P111）；
39. 每个非确定型图灵机都等价于某一个确定型图灵机（P112）；
40. 如果对所有输入所有分支都停机，则称这个非确定型图灵机是一个判定器；
41. 一个语言是图灵可识别的，当且仅当存在枚举器枚举它（P113）；
42. 不能在带子的输入区域写的单带图灵机只能识别正则语言；
43. 每一个无穷图灵可识别语言都有一个无穷可判定子集；
44. 一个语言是可判定的，当且仅当有枚举器以标准字符串顺序枚举这个语言；
45. 图灵可识别语言类在并、连接、星号、交、同态运算下封闭；
46. 可判定语言类在并、连接、星号、补、交运算下封闭；

第 4 章 可判定性

47. A_{DFA}是可判定的（直接模拟）；
48. A_{NFA}是可判定的（把NFA转换成DFA）；
49. A_{REX}是可判定的（把R转换成NFA）；
50. E_{DFA}是可判定的（从起始状态开始标记，检查有无接受状态被标记）；
51. EQ_{DFA}是可判定的（对称差，$(A \cap \bar{B}) \cup (\bar{A} \cap B) = \Phi$）；
52. A_{CFG}是可判定的（转换成乔姆斯基范式，在$2n-1$步内模拟）；
53. E_{CFG}是可判定的（从终结符开始标记，检查起始变元是否被标记）；
54. EQ_{CFG}是不可判定的（利用ALL_{CFG}是不可判定的）；
55. 每个上下文无关语言都是可判定的（利用A_{CFG}的判定器）；
56. A_{TM}是不可判定的（对角化方法，P130），是图灵可识别的（直接模拟）；
57. 一个语言是可判定的，当且仅当它既是图灵可识别的，也是补图灵可识别的（交替模拟）；
58. ALL_{DFA}是可判定的（利用EQ_{DFA}的判定器）；
59. E_{TM}是补图灵可识别的（在所有字符串上运行M）；

第 5 章 可归约性

60. $HALT_{TM}$是不可判定的（A_{TM}可规约到$HALT_{TM}$）；
61. E_{TM}是不可判定的（A_{TM}可规约到E_{TM}：对于输入$< M, w >$，构造机器M_1拒绝所有不是w的输入，在输入w上运行M；在$< M_1 >$上运行E_{TM}）；
62. $REGULAR_{TM}$是不可判定的（同上规约）；
63. Rice's theorem：测定语言的任何一个性质是否可由图灵机识别都是不可判定的；
64. EQ_{TM}是不可判定的（E_{TM}可规约到EQ_{TM}）；
65. 图灵机在输入上的计算历史就是当这个图灵机处理此输入时所经过的格局序列；如果M在w上不停机，则M在w上既没有接受也没有拒绝计算历史存在；
66. 线性界限自动机LBA是一种受到限制的图灵机，它不允许其读写头离开包含输入的带子区域；
67. LBA使用一个比输入字母表要大一些的带子字母表，就能使得可用存储增加到常数倍；即LBA对于长度为n的输入，可用存储量关于n是线性的；
68. A_{DFA}、A_{CFG}、E_{DFA}、E_{CFG}的判定器都是LBA；每个CFL都可由一个LBA来判定；
69. A_{LBA}是可判定的（利用格局判定是否停机，M的格局数为qng^n个，在M上模拟qng^n步）；
70. E_{LBA}是不可判定的（A_{TM}可利用计算历史规约到E_{LBA}，构造LBA识别TM的接受计算历史，P141）；
71. ALL_{CFG}是不可判定的（A_{TM}可利用计算历史规约到ALL_{CFG}，构造CFG派生所有不是TM的接受计算历史的串，P143）；
72. 函数$f: \Sigma^* \to \Sigma^*$是一个可计算函数，如果有某个图灵机M，使得在每个输入w上M停机，且这时只有$f(w)$出现在带子上；
73. 语言A是映射可归约到语言B的，如果存在可计算函数$f: \Sigma^* \to \Sigma^*$使得对每个w，

$$w \in A \Leftrightarrow f(w) \in B$$

记作$A \leq_m B$；称函数f为从A到B的规约；
74. 如果$A \leq_m B$且B是可判定的，则A也是可判定的；
75. 如果$A \leq_m B$且A是不可判定的，则B也是不可判定的；
76. 不存在从A_{TM}到E_{TM}的映射规约（因为 79，且E_{TM}是补图灵可识别的）；
77. 如果$A \leq_m B$且B图灵可识别的，则A也是图灵可识别的；
78. 如果$A \leq_m B$且A不是图灵可识别的，则B也不是图灵可识别的；

79. $A \leq_m B$和$\bar{A} \leq_m \bar{B}$具有相同的含义；
80. EQ_{TM}既不是图灵可识别的，也不是补图灵可识别的（A_{TM}可规约到EQ_{TM}和$\overline{EQ_{TM}}$）；
81. EQ_{CFG}是不可判定的（利用ALL_{CFG}是不可判定的），是补图灵可识别的（转换成乔姆斯基范式，在所有字符串上在$2n-1$步内模拟）；
82. 如果$A \leq_m B$且B是一个正则语言，A不一定也是一个正则语言；
83. 如果A是图灵可识别的，且$A \leq_m \bar{A}$，则A是可判定的；

第 6 章 可计算性理论的高级专题

84. 存在可计算函数$q: \Sigma^* \to \Sigma^*$，对任意串w，$q(w)$是图灵机P_w的描述，P_w打印出w，然后停机（P155）；
85. 递归定理：设T是计算函数$t: \Sigma^* \times \Sigma^* \to \Sigma^*$的一个图灵机，则存在计算函数$r: \Sigma^* \to \Sigma^*$的一个图灵机$R$，使得对每一个$w$，有（P157）：

$$R(w) = t(< R >, w)$$

86. A_{TM}是不可判定的（递归定理，P158）；
87. MIN_{TM}不是图灵可识别的（用枚举器枚举一个比自己的描述更长的机器并模拟它）；
88. 设$t: \Sigma^* \to \Sigma^*$是一个可计算函数，则存在一个图灵机F，使得$t(< F >)$描述一个与F等价的图灵机（递归定理，P159）；
89. 逻辑理论的可判定性：$Th(\boldsymbol{N}, +)$是可判定的；$Th(\boldsymbol{N}, +, \times)$是不可判定的（略）；
90. 语言B的一个谕示是一个能够报告某个串w是否为B的成员的外部装置；一个谕示图灵机是一种修改过的图灵机，它有询问一个谕示的额外能力；记M^B为对语言B有谕示的谕示图灵机；
91. 语言A图灵可归约到B，如果A相对于B是可判定的，记作$A \leq_T B$；
92. 如果$A \leq_T B$且B是可判定的，则A也是可判定的；
93. 极小长度的描述、不可压缩的串（P165，略）；
94. MIN_{TM}的任何无限子集都不是图灵可识别的（同 87）；

第 7 章 时间复杂性

95. 令M是一个在所有输入上都停机的确定型图灵机；M的运行时间或者时间复杂度是一个函数$f: \boldsymbol{N} \to \boldsymbol{N}$，其中$\boldsymbol{N}$是非负整数集合，$f(n)$是$M$的所有长度为$n$的输入上运行时经过的

最大步数；若$f(n)$是M的运行时间，则称M在时间$f(n)$内运行，M是$f(n)$时间图灵机；通常使用n表示输入的长度；

96. 设f和g是两个函数$f,g:\boldsymbol{N}\to\boldsymbol{R}^+$；称$f(n)=O(g(n))$，若存在正整数$c$和$n_0$，使得对所有$n\geq n_0$有

$$f(n)\leq cg(n)$$

当$f(n)=O(g(n))$时，称$g(n)$是$f(n)$的上界，或更准确地说，$g(n)$*是*$f(n)$的渐进上界，以强调没有考虑常数因子。

97. 设f和g是两个函数$f,g:\boldsymbol{N}\to\boldsymbol{R}^+$；如果

$$\lim_{n\to\infty}\frac{f(n)}{g(n)}=0$$

则称$f(n)=o\big(g(n)\big)$；换言之，$f(n)=o\big(g(n)\big)$意味着对于任何实数$c>0$，存在一个数n_0，使得对所有$n\geq n_0$，$f(n)<cg(n)$；

98. 令$t:\boldsymbol{N}\to\boldsymbol{R}^+$是一个函数；定义时间复杂性类$TIME(t(n))$*为由*$O(t(n))$时间的图灵机判定的所有语言的集合；

99. 单带图灵机在$o(nlogn)$时间内判定的语言都是正则语言；

100. 设$t(n)$是一个函数，$t(n)\geq n$；则每一个$t(n)$时间的多带图灵机都和某一个$O(t^2(n))$时间的单带图灵机等价（单带模拟多带，P178）；

101. 设N是一个非确定型图灵机，并且是个判定机；N的运行时间是函数$f:\boldsymbol{N}\to\boldsymbol{N}$，其中$f(n)$是在任何长度为$n$的输入上所有计算分支中的最大步数；

102. 设$t(n)$是一个函数，$t(n)\geq n$；则每一个$t(n)$时间的非确定型单带图灵机都与某一个$2^{O(t(n))}$时间的确定型单带图灵机等价（搜索计算树模拟，P179）；

103. P是确定型单带图灵机在多项式时间内可判定的语言类；换言之，

$$P=\bigcup_k TIME(n^k)$$

104. $PATH=\{<G,s,t>\mid G$是具有从s到t的有向路径的有向图$\}$，$PATH\in P$（BFS）；

105. $RELPRIME=\{<x,y>\mid x$与y互素$\}$，$RELPRIME\in P$（欧几里德算法）；

106. 每一个上下文无关语言都是P的成员（动态规划，P184）；

107. 语言A的验证机是一个算法V，这里

$$A=\{w\mid \text{对某个字符串}c,\ V\text{接受}<w,c>\}$$

因为只根据w的长度来度量验证机的时间，所以多项式时间验证机在w的长度的多项式

时间内运行；若语言A有一个多项式时间验证机，则称它为多项式可验证的；

108. 对于多项式验证机，证书具有多项式的长度（w的长度），因为这是该验证机在它的时间界限内所能访问的全部信息长度；

109. NP是具有多项式时间验证机的语言类；

110. $HAMPATH$和$COMPOSITES$都是NP的成员；$COMPOSITES$也是P的成员；

111. 一个语言在NP中，当且仅当它能被某个非确定型多项式时间图灵机判定（P186）；

112. $NTIME(t(n)) = \{ L \mid L$是一个被$O(t(n))$时间的非确定型图灵机判定的语言$\}$；

113. $NP = \cup_k NTIME(n^k)$；

114. $LIQUE = \{ < G, k > \mid G$是包含$k$团的无向图$\}$属于$NP$（证书）；

115. $SUBSET - SUM = \{ < s, t > \mid s = \{x_1, \cdots, x_k\} \}$，且存在（证书）

$$\{y_1, \cdots, y_l\} \subseteq \{x_1, \cdots, x_k\}$$使得$\sum y_i = t$

116. $coNP$包括NP中的语言的补语言，还不知道$coNP$是否与NP不同；

117. $NP \subseteq EXPTIME = \cup_k TIME(2^{n^k})$，但不知道$NP$是否包含在某个更小的确定型时间复杂性类中；

118. $SAT = \{ < \phi > \mid \phi$ 是可满足的布尔公式 $\}$；$SAT \in P$，当且仅当$P = NP$；

119. 若存在多项式时间图灵机M，使得在任何输入w上，M停机时$f(w)$恰好在带子上，则称函数$f: \Sigma^* \to \Sigma^*$为多项式时间可计算函数；

120. 语言A称为多项式时间映射可归约到语言B，或简称为多项式时间可归约到B，记为$A \leq_P B$，若存在多项式时间可计算函数$f: \Sigma^* \to \Sigma^*$，对于每一个w，有

$$w \in A \Leftrightarrow f(w) \in B$$

函数f称为A到B的多项式时间归约；

121. 若$A \leq_P B$且$B \in P$，则$A \in P$；

122. $3SAT$多项式时间可归约到$CLIQUE$（P191）；

123. 如果语言B满足下面两个条件，就称为NP完全的：

1) B属于NP；
2) NP中的每个A都多项式时间可归约到B；

124. 若上述的B是NP完全的，且$B \in P$，则$P = NP$；

125. 若上述的B是NP完全的，且$B \leq_P C$，C属于NP，则C是NP完全的；

126. 库克-列文定理：SAT是NP完全的（P192，略）；

127. $SAT \leq_P 3SAT$；

128. $3SAT$多项式时间可归约到$VERTEX - COVER$（P196）；

129. $3SAT$多项式时间可归约到$HAMPATH$（P198）；

130. $HAMPATH$多项式时间可归约到$UHAMPATH$（P200）；

131. $3SAT$多项式时间可归约到$SUBSET - SUM$（P201）；

132. P在并、连接和补运算下封闭；

133. NP在并和连接运算下封闭；

134. $CONNECT = \{ < G > \mid G$是连通的无向图$\} \in P$；

135. $TRIANGLE = \{ < G > \mid G$包含一个三角形$\} \in P$；

136. $ALL_{DFA} \in P$；

137. $EQ_{DFA} \in P$；不知道EQ_{NFA}是否属于P；

138. $SET - SPLITTING$是NPC的（P204）；

139. $VERTEX - COVER$多项式时间可归约到$DOMINATING - SET$（P205）；

140. $PUZZLE$是NPC的（P205）；

第 8 章 空间复杂性

141. 令M是一个在所有输入上都停机的确定型图灵机；M的空间复杂度是一个函数$f: \boldsymbol{N} \to \boldsymbol{N}$，其中$f(n)$是$M$在任何长度为$n$的输入上扫描带子方格的最大数；若$M$的空间复杂度是$f(n)$，则称$M$在空间$f(n)$内运行；

142. 如果M是对所有输入在所有分支上都停机的非确定型图灵机，则将它的空间复杂度$f(n)$定义为M对任何长为n的输入，在任何计算分支上所扫描的带子方格的最大数。

143. 令$f: \boldsymbol{N} \to \boldsymbol{R}^+$是一个函数。空间复杂性类$SPACE(f(n))$和$NSPACE(f(n))$定义如下：

$$SPACE(f(n)) = \{ L \mid L$是被$O(f(n))$空间的确定型图灵机判定的语言$\}$$

$$NSPACE(f(n)) = \{ L \mid L$是被$O(f(n))$空间的非确定型图灵机判定的语言$\}$$

144. 萨维奇定理：对于任何函数$f: \boldsymbol{N} \to \boldsymbol{R}^+$，其中$f(n) \geq log n$，（可产生性问题，P209）

$$NSPACE(f(n)) \subseteq SPACE(f^2(n))$$

145. $PSPACE$是在确定型图灵机上、在多项式空间内可判定的语言类；换言之，

$$PSPACE = \bigcup_k SPACE(n^k)$$

146. 根据萨维奇定理，$NPSPACE = PSPACE$；

147. 一个消耗$f(n)$空间的图灵机至多有$f(n)2^{O(f(n))}$个不同的格局；

$$P \subseteq NP \subseteq PSPACE = NPSPACE \subseteq EXPTIME = \bigcup_k TIME(2^{n^k})$$

148. 若语言B满足下面两个条件，则称它是$PSPACE$完全的：

1) B属于$PSPACE$；
2) $PSPACE$中的每一个语言A多项式时间可归约到B；

若B只满足条件 2，则称它为$PSPACE$难的；

149. $TQBF = \{ < \phi > | \phi$是真的全量词化的布尔公式$\}$是$PSPACE$完全的（P212）；

150. 设A是一个由图灵机M在n^k空间内判定的语言，将A多项式规约到$TQBF$的公式长度为n^{2k}；

151. $FORMULA - GAME = \{ < \phi > |$ 在与ϕ相关联的公式博弈中选手E有必胜策略$\}$ 是$PSPACE$完全的（P215）；

152. $GG = \{ < G, b > |$ 在图G上以结点b起始的广义地理学游戏中，选手Ⅰ有必胜策略$\}$ 是$PSPACE$完全的，$FORMULA - GAME$多项式时间可规约到GG（P217）；

153. L是确定型图灵机在对数空间内可判定的语言类；换言之，

$$L = SPACE(logn)$$

NL是非确定型图灵机在对数空间内可判定的语言类；换言之，

$$NL = NSPACE(logn)$$

154. $PATH \in NL$（工作带上只记录每一步当前结点的位置，P219）；

155. 若M是一个有单独的只读输入带的机器，w是输入，则M在w上的格局包含状态、工作带和两个读写头位置；输入w不作为M在w上的格局的一部分；

156. 对数空间转换器是有一条只读输入带、一条只写输出带和一条读/写工作带的图灵机；输出带的头部不能向左移动，因此它不能读已写内容；工作带可以包含$O(log(n))$个符号；对数空间转换器M计算一个函数$f: \Sigma^* \to \Sigma^*$，其中$f(w)$是把w放在M的输入带上启动M运行到M停机时输出带上存放的字符串，称f为对数空间可计算函数；如果语言A通过对数空间可计算函数f映射可归约到语言B，则称A对数空间可归约到B，记为$A \leq_L B$；

157. 语言B是NL完全的，如果

1) $B \in NL$；

2) NL中的每个A对数空间可归约到B；

158. $A \leq_L B$且$B \in L$，则$A \in L$（P221）；

159. $PATH$是NL完全的（构造格局图G，P221）；

160. $NL \subseteq P$（消耗空间$f(n)$的图灵机在时间$n^{2O(f(n))}$内运行，P222）；

161. $NL = coNL$（$\overline{PATH} \in NL$，P222）；

162. $L \subseteq NL = coNL \subseteq P \subseteq NP \subseteq PSPACE$；

163. $PSPACE$在并、补和星号运算下封闭；

164. $A_{DFA} \in L$；

165. $PSPACE$难的语言也是NP难的；

166. NL在并、连接和星号运算下封闭；

167. $BIPARTITE = \{ < G > \mid G$是二部图$\} \in NL$；

168. $STRONGLY - CONNECTED = \{ < G > \mid G$是强连通图$\}$是$NL$完全的；

169. $BOTH_{NFA} = \{ < M_1, M_2 > \mid M_1$和$M_2$是$NFA$，$L(M_1) \cap L(M_2) \neq \Phi \}$是$NL$完全的；

170. A_{NFA}是NL完全的；

171. E_{DFA}是NL完全的；

172. $2SAT$是NL完全的；

173. $CYCLE = \{ < G > \mid G$是包含一个有向回路的有向图$\}$ 是NL完全的；

174. $EQ_{REX} = \{ < R, S > \mid R$和$S$是等价的正则表达式$\} \in PSPACE$；

175. A_{LBA}是$PSPACE$完全的；

第 9 章 难解性

176. 对于函数$f: \boldsymbol{N} \to \boldsymbol{N}$，其中$f(n)$至少为$O(logn)$，如果函数$f$把$1^n$映射为$f(n)$的二进制表示，并且该函数在空间$O(f(n))$内是可计算的，则称该函数为空间可构造的；

177. 空间层次定理：对于任何空间可构造函数$f: \boldsymbol{N} \to \boldsymbol{N}$，存在语言$A$，在空间$O(f(n))$内可判定，但不能在空间$o(f(n))$内判定（对角线方法，P229）；

178. 对于任意两个函数$f_1, f_2: \boldsymbol{N} \to \boldsymbol{N}$，其中$f_1(n)$等于$o(f_2(n))$，$f_2$是空间可构造的，有$SPACE\big(f_1(n)\big) \subset SPACE(f_2(n))$；

179. 对于任意两个实数$0 \le \varepsilon_1 < \varepsilon_2$，有

$$SPACE(n^{\varepsilon_1}) \subset SPACE(n^{\varepsilon_2})$$

180. $NL \subset PSPACE \subset EXPSPACE$；

181. 对于函数$t: \boldsymbol{N} \to \boldsymbol{N}$，其中$t(n)$至少为$O(nlogn)$，如果函数$t$把$1^n$映射为$t(n)$的二进制表示，并且该函数在空间$O(t(n))$内是可计算的，则称该函数为时间可构造的；

182. 时间层次定理：对于任何时间可构造函数 t: $\boldsymbol{N} \to \boldsymbol{N}$，存在语言$A$，在时间$O(t(n))$内可判定，但在时间$o(t(n)/logt(n))$内不可判定（对角线方法，P231）；

183. 对于任意两个函数$t_1, t_2: \boldsymbol{N} \to \boldsymbol{N}$，其中$t_1(n)$等于$o(t_2(n)/logt_2(n))$，而且$t_2$是空间可构造的，有$TIME\big(f_1(n)\big) \subset TIME(f_2(n))$;

184. 对于任意两个实数$1 \le \varepsilon_1 < \varepsilon_2$，有

$$TIME(n^{\varepsilon_1}) \subset TIME(n^{\varepsilon_2})$$

185. $P \subset EXPTIME$；

186. 语言B是$EXPSPACE$完全的，如果

1) $B \in EXPSPACE$；
2) $EXPSPACE$中的每个A都多项式时间可归约到B；

187. $EQ_{REX\uparrow} = \{ < Q, R > \mid Q$和$R$是等价的带指数运算的正则表达式 $\}$ 是$EXPSPACE$完全的（把输入w映射为一对表达式R_1和R_2，R_1产生所有字符串，R_2产生不代表M在w上的拒绝计算历史的所有字符串，它们等价当且仅当M接受w，P234）；

188. P^A是采用谕示A的多项式时间谕示图灵机可判定的语言类；NP^A是采用谕示A的多项式时间非确定型谕示图灵机可判定的语言类；

189. $NP \subseteq P^{SAT}$，$coNP \subseteq P^{SAT}$（P^{SAT}是一个确定型复杂性类，在补运算下封闭）；

190. 不太可能用对角化方法解决P与NP的问题：

存在谕示A使得$P^A \neq NP^A$（P238）；

存在谕示B使得$P^B = NP^B$（$NP^{TQBF} \subseteq NPSPACE \subseteq PSPACE \subseteq P^{TQBF}$）；

191. 一个电路族C是电路的一个无穷列表$(C_0, C_1, C_2, \cdots)$，其中C_n有n个输入变量，称C在$\{0,1\}$上判定语言A，如果对于每个字符串$w(w \in A)$当且仅当$C_n(w) = 1$，其中n是w的长度；

192. 电路的规模是它所包含的门的数目；一个电路族$(C_0, C_1, C_2, \cdots)$的规模复杂度是一个函数

$f: \boldsymbol{N} \to \boldsymbol{N}$，其中$f(n)$是$C_n$的规模；

193. 电路的深度是从输入变量到输出门的最长路径的长度；一个电路族$(C_0, C_1, C_2, \cdots)$的深度复杂度是一个函数$f: \boldsymbol{N} \to \boldsymbol{N}$，其中$f(n)$是$C_n$的深度；

194. 语言的电路复杂度是该语言的极小电路族的规模复杂度，语言的电路深度复杂度是该语言的深度极小电路族的深度复杂度；

195. 设$t: \boldsymbol{N} \to \boldsymbol{N}$是一个函数，$t(n) \geq n$；若$A \in TIME(t(n))$，则$A$的电路复杂度为$O(t^2(n))$（按画面构造电路，P240）；

196. 电路可满足性问题$CIRCUIT - SAT$是NP完全的（模拟验证机V）；

197. $CIRCUIT - SAT$可多项式时间规约到$3SAT$（用$3SAT$模拟电路，P243）；

198. 若$NP = P^{SAT}$，则$NP = coNP$；

第 10 章 复杂性理论高级专题

199. $MIN - VERTEX - COVER$和$MAX - CUT$的近似算法（P247）；

200. 概率图灵机$PTM\ M$是一种非确定性图灵机，它的每一非确定性步称作掷硬币步，并且有两个合法的下步动作；按照下述方式把概率赋给M对输入w的每一个计算分支b；定义分支b的概率为

$$Pr[b] = 2^{-k}$$

其中，k是在分支b中出现的掷硬币步的步数，定义M接受w的概率为

$$Pr[M\text{接受}w] = \sum_{b\text{是接受分支}} Pr[b]$$

201. 对于某个正数$\varepsilon(0 \leq \varepsilon < 1/2)$，如果

1) $w \in A$蕴含$Pr[M\text{接受}w] \geq 1 - \varepsilon$；

2) $w \notin A$蕴含$Pr[M\text{拒绝}w] \geq 1 - \varepsilon$；

则称 M 以错误概率ε判定语言A；

202. BPP是多项式时间的概率图灵机以错误概率$1/3$判定的语言类；

203. 加强引理：设ε是一给定的常数，且$0 \leq \varepsilon < 1/2$；又设M_1是一台错误概率为ε的多项式时间概率图灵机，则对于任意给定的多项式$p(n)$，存在与M_1等价的错误概率为$2^{-p(n)}$的多项式时间概率图灵机M_2（运行k次，P250）；

204. 素数性（P250，略）；

205. $PRIMES \in BPP$；

206. RP是多项式时间概率图灵机识别的语言类，在这里，在语言中的输入以不小于$1/2$的概率被接受，不在语言中的输入以概率1被拒绝；

207. $COMPOSITES \in RP$；

208. 分支程序BP是一个有向无环图；分支程序与L类的关系类似于布尔电路与P类的关系；

209. EQ_{BP}是$coNP$完全的；

210. 只读一次的分支程序$ROBP$是这样的一种分支程序，从它的起始顶点到输出顶点的每一条有向路径上，每一个变量至多能被查询一次；

211. $EQ_{ROBP} \in BPP$（化为多项式赋值，P255）；

212. 交错式图灵机ATM是一种具有特殊功能的非确定型图灵机；除q_{accept}和q_{reject}外，它的状态分为全称状态和存在状态；当对输入串运行交错式图灵机时，根据对应的格局是包含全称状态还是包含存在状态，用∧或∨标记它的非确定型计算树的每一个顶点；如果一个顶点标记∧且它的儿子都接受，或者标记∨且它的儿子中有一个接受，则指定这个顶点接受；如果起始顶点被指定为接受，则接受输入；

213. 交错式图灵机的时间复杂性和空间复杂性是每个计算分支所用的时间和空间的最大值，交错式时间复杂性类和空间复杂性类的定义如下：

$$ATIME\big(t(n)\big) = \{\, L \mid L\text{是被一台}O\big(t(n)\big)\text{时间的交错式图灵机判定的语言}\,\}$$

$$ASPACE\big(t(n)\big) = \{\, L \mid L\text{是被一台}O\big(t(n)\big)\text{空间的交错式图灵机判定的语言}\,\}$$

214. AP、$APSPACE$和AL分别是多项式时间、多项式空间和对数空间的交错式图灵机判定的语言类；

215. $TAUT = \{ < \phi > \mid \phi\text{是一个永真式} \} \in AP$；

216. $NP \in AP$，$coNP \in AP$；

217. 对于$f(n) \geq n$，有（深度优先搜索、可产生性递归，P259）

$$ATIME\big(f(n)\big) \subseteq SPACE\big(f(n)\big) \subseteq ATIME(f^2(n))$$

对于$f(n) \geq logn$，有（P259）

$$ASPACE\big(f(n)\big) = TIME(2^{O(f(n))})$$

218. 多项式时间层次（P260，略）；

219. 交互式证明系统（P261，略）；

220. $IP = PSPACE$（P263，略）；

221. 布尔电路的处理器复杂度定义为它的规模，布尔电路的并行时间复杂度定义为它的深度；

222. 设$(C_0, C_1, C_2, \cdots)$是一族电路，如果存在对数空间转换器T，当T的输入为1^n时，T输出$< C_n >$，则称该电路族是一致的；

223. 如果存在规模复杂度为$f(n)$和深度复杂度为$g(n)$的一致电路族识别某个语言，则称这个语言的规模-深度联合电路复杂度不超过$(f(n), g(n))$；

224. 对于$i \geq 1$，令NC^i是能够用多项式规模和$O(log^i n)$深度的一致电路族识别的语言类；NC是所有在某个NC^i中的语言组成的语言类；用这种电路族计算的函数分别叫做NC^i可计算的和NC可计算的；

225. $NC^1 \subseteq L$（从输出门开始深度优先搜索电路的值，P272）；

226. $NL \subseteq NC^2$（计算NL机格局图的传递闭包，P272）；

227. $NC \subseteq P$（直接模拟）；

228. 语言B是P完全的，如果

    1) $B \in P$；

    2) P中每一个A对数空间可归约到B；

229. 如果$A \leq_L B$且B在NC中，则A在NC中；

230. $CIRCUIT - VALUE$是P完全的；

231. 密码学（P273，略）；

232. $BPP \subseteq PSPACE$；

233. $BPL \subseteq P$；

234. ZPP –机器M是一台概率图灵机，它的每个分支有三种输出：接受、拒绝和？；如果M对每个输入串w都输出正确答案的概率大于等于$2/3$而且M从不回答错误，则M判定语言A；对每个输入，M输出？的概率至多为$1/3$；更进一步，对输入w，M所有分支上的平均运行时间一定限定在输入串w长度的多项式时间；令ZPP是ZPP –机器识别语言的集合；

235. $ZPP = RP \cap coRP$；