

問1 次の(1)~(5)の問いに答えなさい。

(1) 下の(a)~(e)に示すコンパイラの処理過程を、実行順（論理的な意味での）に並べなさい。

(a) オブジェクトコード生成 (b) 構文解析 (c) コード最適化 (d) 字句解析 (e) 中間コード生成

(2) コード最適化で何を行うかを簡単に説明しなさい。

(3) 中間コードを用いることの利点を一つ述べなさい。

(4) マルチパスコンパイラと比較したワンパスコンパイラの利点と欠点をひとつずつあげなさい。

(5) コンパイラとインタプリタの違いを簡単に説明しなさい。

問2 次の(1)~(7)の問いに答えなさい。

(1) 下の(a)~(c)のような文法定義の記述（規則）を何と呼ぶか。

(2) (a)~(c)のような記述の集まりによって言語の文法を定義する記法を何と呼ぶか。

(3) この記法を文法定義に利用した最初の言語の名前を答えなさい。

(4) (a)に現れる構文変数、終端記号、メタ記号をすべて示しなさい。

(5) (b)を[]を使わず、かわりに|を用いる形に書き改めなさい。

(6) (c)を、[]を利用することで左再帰性のない式に書き改めなさい。

(7) (a)~(c)にならってC言語のfor文を定義する規則を示しなさい。

(a)  $\langle \text{while 文} \rangle \rightarrow \text{while} ( \langle \text{式} \rangle ) \langle \text{文} \rangle$

(b)  $\langle \text{if 文} \rangle \rightarrow \text{if} ( \langle \text{式} \rangle ) \langle \text{文} \rangle [ \text{else} \langle \text{文} \rangle ]$

(c)  $\langle \text{文リスト} \rangle \rightarrow \langle \text{文} \rangle | \langle \text{文リスト} \rangle \langle \text{文} \rangle$

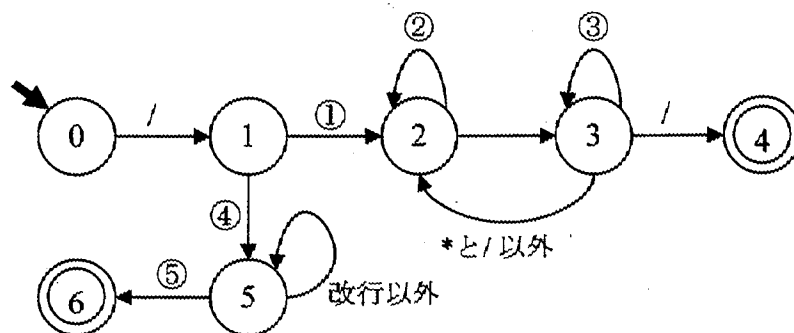
問3 次の(1)~(3)の問いに答えなさい。

(1) C言語のコメントを受理する有限オートマトンを作成したい。下図の①~⑤の入力記号として適切なものを(a)~(f)から選びなさい。

(2) 状態0のような状態をなんと呼ぶか。

(3) 状態4, 6のような状態をなんと呼ぶか。

(4) C言語の識別子は「英字または'\_'で始まり、任意個の英字または数字または'\_'が続く文字列」である。このルールを正規表現で表しなさい。



(a) / (b) \* (c) 改行 (d) / 以外 (e) \* 以外 (f) 改行以外

問4 次の(1)~(3)の問いに答えなさい。

(1) 次のC言語のプログラムの字句の区切りを示しなさい。

```
for(i = 0; i <= 10; i+=2){ func(i); }
```

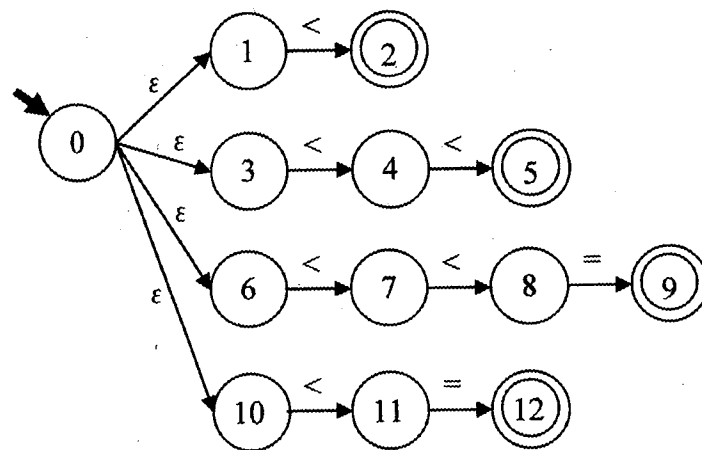
(2) 字句解析における「最長一致方式」を、具体例を用いて説明しなさい。

問5 次の(1)~(3)の問いに答えなさい。

(1) 図のような有限オートマトンを省略語でNFAと呼ぶが、日本語ではなんと呼ばれるか。

(2) NFAとはどのような有限オートマトンであるか。簡単に説明しなさい。

(3) 図のNFAと等価なDFAを示しなさい。



学籍番号 J

問1

(1) 処理順に d b e c a

1 (2) 例えば, while文などで、値の変わらない変数をwhile文の外に出したりする。

-5 (3) 処理が速くなる。

(4) 利点 処理速度が速い。

1 欠点 複雑なプログラムに適さない。

(5) インタプリタは行単位で処理を行う。

-2 コンパイラは一括して処理を行う。

問2

-12 (1) 生成規則 (2) BNF (3) yacc

(4) 構文変数 <while文>, while 終端記号 ), [, ]

メタ記号 →

(5) <if文> → if(<式>){<文>} | if(<式>){<文>} else<文>}

(6) <文リスト> → <文> [[文]]

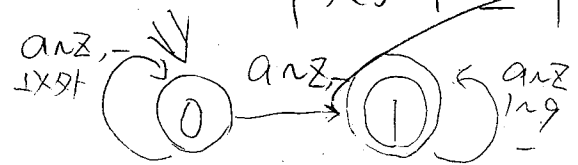
(7) <for文> → for(<式>;<式>;<式>){<文>}

問3

-6 (1) ① b ② e ③ d ④ a ⑤ c

(2) 初期状態 (3) 受理状態

(4) {英字 | ' '}, {英字 | 数字 | ' '}



学籍番号 J.

問 4

(1) (字句ごとに下線で区切る)

for (i=0;i<=10;i+=2){ func(i); }

(2) 例えば、 $i+=2$  という処理があった場合、  
 $i$  の後に  $+$  を読まず、 $=$  を先に読んでから処理を行う方式。

問 5

(1) 決定性オートマトン

(2) 末尾がきちんと定まっている

(3)

