

J-02 CASL アセンブラの製作

依田 亨一

指導教員 石館 勝好

1. 研究概要

情報処理技術者試験の基本情報技術者試験に出題されるプログラム言語 CASL をアセンブルし、オブジェクトコードを生成するアセンブラを製作した。使用言語は、Java SE6 である。

なお、この出力したオブジェクトコードを読み込んで実行するのが水谷が製作するトレサである。

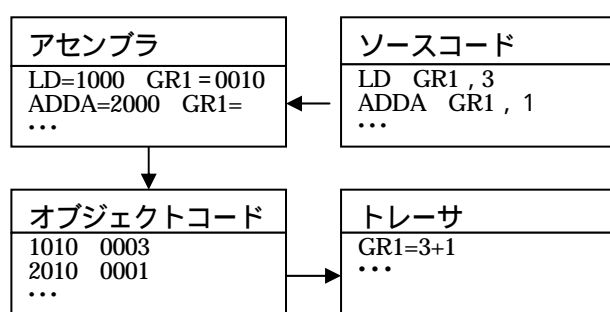


図1 アセンブラとトレサの関係

2. CASL の構文

CASL で使うプログラムは、ラベル、命令、オペランド、コメントの4つの要素によって書かれている。

<u>LB.1</u>	<u>LD</u>	<u>GR1.1</u>	<u>;LD 命令LB.1.1</u>
ラベル	命令	オペランド	コメント

図2 CASL2 の基本的な構文

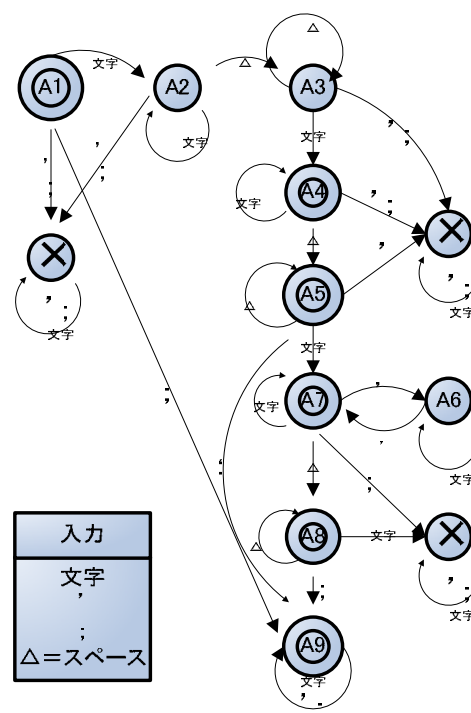
この4つの要素はスペースで区切られ、オペランドはコンマで区切られている。また、ラベル、コメント、オペランドは命令や場合によっては記述されないことがある。

3. 構文の解析

当初は、読み込んだプログラムを区切り文字のスペースやコンマで分割しようとしていた。

しかし、これら区切り文字は、単なる文字列として扱われる場合もあり、上手くいかなかった。

そこで、言語仕様からBNFで定義し直し、これを解釈できる有限オートマトンを考えた(図3)。



これらの処理を実装するために表 1 のクラスを作成した。

表 1 作成したクラス

クラス名	機能
Sentence	プログラムの 1 行分を格納する。Table を用いて、ラベル部、命令部、オペランド部に振り分ける。
Source	Sentence オブジェクト及びラベル部、命令部、オペランド部、メッセージをプログラム行数分だけ格納する。
Operand	カンマで区切られたオペランド部の各部を表す。
Table	有限オートマトンから作った状態遷移表である。入力文字を Sentence クラスのラベル部、命令部、オペランド部に振り分ける。
BinCode	アセンブルして成功した場合の、オブジェクトコードを格納する。
Check	ラベル部、命令部、オペランド部それぞれをチェックする。チェックする部分に応じて、違うメソッドを用意している。

Source クラスを使った処理イメージを説明する。
入力されたソースは有限オートマトン (Table クラス) を使って、ラベル部、命令部、オペランド部に分ける (図 5)。

正誤	ラベル部	命令部	オペランド部	メッセージ部
true	MON1	START		
true		LD	GR1,A	
false		LD	B,GR2	オペランド部にミスがあります
true		RET		
true	A	DC	100	
false	1B	DC	200	プログラムの書き方にミスがあります
true		END		

図 5 エラーのあるプログラム

ラベル部 命令部 オペランド部を順に見ていき、

エラーがある行は正誤を false に、メッセージ部にエラーメッセージを書く。

エラーがなければ図 6 の状態になる。この時は正誤が全て true であり、この状態のときのみ、オブジェクトコードが生成できる。

正誤	ラベル部	命令部	オペランド部	コメント部
True	MON1	START		
True		LD	GR1,A	
True		LD	GR2,B	
True		RET		
true	A	DC	100	
true	B	DC	200	
true		END		

図 6 エラーのあるプログラム

最終的に BinCode クラスのオブジェクトコードに機械語が格納され、番地にプログラムの開始番地が格納される。

番地	オブジェクトコード 1	オブジェクトコード 2
0000	1010	0005
0002	1020	0006
0004	8100	
0005	0100	
0006	0200	

図 7 アセンブル結果

4 まとめ

どんなプログラムでも動作はするが、存在しないラベルのチェック、リテラルの処理、水谷のトレーサに渡すためのフォーマットが定義できていないので残りの時間で作成する。

5 参考文献

- [1] 東田幸樹, 山本芳人, 広瀬啓雄『アセンブラ言語 CASL』工学図書株式会社
- [2] 白鳥則朗, 高橋薫, 神長裕明『ソフトウェア工学の基礎知識』