

平成 28 年度 情報工学コース卒業研究報告要旨

酒井 研究室	氏 名	坂 梨 元 軌
卒業研究題目	C 言語サブセットから Malbolge 難読コードへのコンパイラの設計と実装	
<p>難解プログラミング言語は意図的に読解が困難になるように設計されたプログラミング言語であり、その解析困難性により知的財産権の保護などに役立つ。この目的として難解性が高い Malbolge は適しているが、そのプログラムを作成することは困難である。</p> <p>Malbolge プログラムを生成する低級アセンブラおよび低級アセンブリプログラムを生成する高級アセンブラが飯澤らによって開発され、Malbolge のプログラミング環境が整いつつある。しかしこの高級アセンブラはインタプリタ形式のプログラムを生成するため、生成された Malbolge コードの実行中にメモリ上に出現する数値列から元の高級アセンブリプログラムが復元できるという問題がある。この問題の解決のために河邊らによって低級アセンブリ言語へコンパイラ方式で変換可能な制御付き疑似命令列が開発された。この制御付き疑似命令列は、低級アセンブリ言語と比較するとプログラミング容易な言語であるが、演算や制御文が特殊であるなど C 言語などと比較すると依然としてプログラミング困難な言語である。また高級アセンブリ言語も同様に C 言語などと比較するとプログラミング困難な言語である。そこで、C 言語に近い文法を持つ言語から制御付き疑似命令列へのコンパイラの作成が望まれている。</p> <p>本研究は、C 言語の文法に制限を設けた高級言語 (C 言語サブセット) から Malbolge コードへのコンパイラを実現することを目的とする。そのために (1) 関数構文の導入による制御付き疑似命令列の拡張、(2) 高級言語から拡張した制御付き疑似命令列への変換系の実装、を行い、先行研究と併せて目的とするコンパイラを実現する。</p> <p>(1) 関数構文の実現においては、関数のエントリーポイントへのジャンプと関数処理後に実行する呼び出し元へのジャンプの 2 種類のジャンプが必要である。前者は関数ごとにエントリーポイントが一意であるため、低級アセンブリ言語の IF_BRANCH 命令を利用して実現できる。しかしながら後者は、戻り先のアドレスが一般には複数あるためその実現には呼び出し元のアドレスが格納された変数の値を参照してジャンプするような動的ジャンプ機能が必要であるが、既存の低級アセンブリ言語にはこの機能がない。そこで動的ジャンプ機能を Malbolge 命令で直接記述し、低級アセンブリ言語の命令定義機能を利用してこれを新たな命令として定義した。この命令を利用することで関数の実現が可能となった。なお低級アセンブリ言語への変換の複雑さを考慮し、ここで拡張した関数構文では再帰呼び出しには対応しない。</p> <p>(2) 高級言語から拡張制御付き疑似命令列への変換においては、(a) 関数の再帰呼び出し、(b) 比較演算、(c) 論理演算、(d) 入出力命令、(e) if 文、(f) while 文の実現方法を提案する。(a) の関数の再帰呼び出しは、制御付き疑似命令列上の配列操作命令を利用したコールスタックにより実現する。(b)、(c) 及び (d) については、ROT 命令及び OPR 命令を組み合わせて実現する。(e) 及び (f) の制御文については、条件式として与えられた式を (b) 比較演算及び (c) 論理演算を利用して評価することで実現する。これらに加えて既に提案されている代入演算・インクリメント演算・デクリメント演算・加算演算の実現方法を利用することで高級言語から拡張した制御付き疑似命令列への変換を実現する。また加算演算について、既に提案されている実現方法を改良し高速化を行う。</p>		