# 平成22年度 情報工学コース卒業研究報告要旨

| 酒井 研究室 | 氏　　名 | 安 藤　聡 |
|---|---|---|
| 卒業研究題目 | 難解言語 Malbolge における加算の効率よい実現について | |

難解プログラミング言語は，プログラミングを困難にするのを目的として設計された言語である．このような言語でのプログラミング手法を確立することは，情報セキュリティにおいて知的財産の保護やシステムの安全性の確保に役立つと考えられている．Malbolge は，数ある難解言語の中でも特に悪辣として知られており，プログラムの解読や改竄だけでなく，プログラムの作成すら非常に困難である．しかし近年，飯澤らによって，それまで困難と考えられていたループプログラムの作成手法が提案された．提案手法では，Malbolge の演算命令を拡張した疑似命令でプログラムを考えて，それをループ処理が可能な低級アセンブリプログラムにコーディングし，飯澤らが開発した低級アセンブラを用いることで Malbolge コードが作成できる．低級アセンブラは，長坂らによって使いやすく改良されている．また，高級アセンブリ言語も飯澤らによって設計され，低級アセンブリ言語を用いた基本モジュール方式で実現されている．しかし，高級アセンブリ言語の算術命令にはインクリメントとデクリメントしかなく，加算を行うためにはインクリメントを繰り返し用いる必要があった．

本論文では，高級アセンブリ言語への加算命令の追加を実現するために，低級アセンブリ言語で加算モジュール作成し，それを評価する．また，高級アセンブリ言語への加算命令の組み込み方針の提案とそれに伴う課題を明らかにする．

低級アセンブリ言語では，Malbolge と同様に一語が十桁の三進数 (10trits) であるため，本研究で作成する加算モジュールも 10trits で表される二数の加算を行う．このモジュールを作成するために，(1) 加算モジュールの基となる加算アルゴリズムの提案，(2) 加算モジュールの疑似命令列での実現，(3) 低級アセンブリ言語による加算モジュールの実装を行う．(1) のアルゴリズムは，変数 $x$,$y$ に入力の二数を格納し，以下の (i) から (iii) を十回繰り返すと，$x$ に出力となる加算結果が格納される．

(i) $x$,$y$ に対して桁上げを考慮しない加算を行う
(ii) $x$,$y$ の加算の桁上げのみ計算し，結果は左に 1trit シフトし最下位 trit を 0 クリアする
(iii) 前述の二つの計算結果を $x$,$y$ にコピーする

(2) の疑似命令列の実現では，まずアルゴリズムで必要な各機能を疑似命令列で実現する．そのために，入力の二引数三値関数を実現する疑似命令列を探索するプログラムを作成する．このプログラムは深さ優先探索に従い探索を行う．ただし, ヒューリスティクスを導入して，枝刈りを行っている．探索プログラムやノウハウにより，必要な疑似命令列を発見し，それらを用いて加算モジュールを疑似命令列で実現する．(3) のモジュールの実装では，(2) で作成した疑似命令列を参考に低級アセンブリプログラムを作成する．この作成には，特定アドレスへの複数アクセス，フリップフロップのような役割を果たすフラグによる実行制御，各所で必要なアドレス計算などの困難を伴う．

次に加算モジュールの評価について述べる．理論的な評価では，従来のインクリメントを用いた加算と本研究で実現した加算で，それぞれに必要な演算命令数を比較した結果，本研究で実現した加算を用いた方が，インクリメントが 9 回以上必要な加算において効率よく計算を行えるという結果が得られた．また実験的な評価においても，従来の加算では計算に必要なインクリメントの数が増えるのに従い処理時間が増加するのに対し，本研究の加算は計算内容に関わらず一定の処理時間で済むという結果が得られた．