

リバースモデリングツール

RExSTM for C

要求仕様書

改訂履歴

版数	発行日	ツールバージョン	改定者	改定内容
第 1 版	2017.03.01	Ver1.32	若林丈紘 ¹	初版作成
第 2 版	2017.03.06	Ver1.32	若林丈紘 ¹	状態遷移表の書式における，要求にそぐわない項目の修正
第 3 版	2019.08.16	Ver2.1.1	清水貴裕 ¹	現状の要求仕様に則って軽微な修正

目次

1	本書の目的.....	3
2	作成される表の概要	3
2.1	条件処理表の構成要素	3
2.2	条件処理表の読み方	4
2.3	条件処理表の書式.....	4
2.4	状態遷移表の構成要素	6
2.5	状態遷移表の読み方	6
2.6	状態遷移表の書式.....	8
3	機能要求	10
4	動作環境	10

¹名古屋大学

1 本書の目的

本書は、C 言語のソースコードを対象に状態遷移表をリバース生成するためのソフトウェア要求を定義する。なお、ソフトウェアの名称は「RExSTM for C」であり、以降、本書では特に指定がない限り本ツールと呼ぶ。

本書の提供先と使用目的は、次のとおりである。

- 本ツールの発注者
 - 本ツールの発注者は、本書により、本ツールの仕様をツール開発者と合意する。また、仕様の変更や追加の発注がある場合には、本書で定義された要求との整合性を確認する。
- 本ツールのアーキテクチャ設計者
 - 本ツールのアーキテクチャ設計者は、本書で定義された要求に基づき、本ツールのアーキテクチャ設計を行う。
- 統合テスト設計者
 - 統合テスト設計者は、本書で定義されたとおりにプログラムが作成されていることを確認するために、本ツールの統合テスト仕様を作成する。

2 作成される表の概要

本ツールは、1つのソースコードに対して以下2つの表を作成する。

- 1) 条件処理表
- 2) 状態遷移表

条件処理表とは、本ツールにおいて状態遷移表をリバース生成する過程で、プログラム構造の理解および状態変数の選択に関するユーザの補助を目的として生成される表を指す。

本節では、条件処理表と状態遷移表の構成要素・読み方・書式を説明する。

2.1 条件処理表の構成要素

条件処理表は以下2つの要素で構成される。

- A) 条件 実行される事柄を分岐させる事象
- B) 処理 ある事象に対して行う事柄

2.2 条件処理表の読み方

条件処理表は条件と処理の組合せにより、プログラム実行の流れを表す。

条件処理表に存在する条件の判定と処理は上から下に向かって順に実行されることを意味する。

ある条件が満たされるとき、その条件と同じ行に存在するセルの処理が上から下に向かって順に実行される。

ただし、条件処理表のある行において、条件が複数列にわたって記述されている場合、それらの条件がすべて満たされるときにのみ処理が実行される。

2.3 条件処理表の書式

条件処理表は、ソースファイルごとに 1 枚の Excel シートに作成される。

条件処理表は関数ごとに作成される。

シートの構成は以下のとおりである。ソースファイル中の関数 1 つにつき、例外時に警告文を表示する行、状態遷移表の行、空白行が用意される。

	B	C	D	E	F	...
3	B3		警告文			
4						
5			条件処理表			
6						
...			空白			

ソースコード 1 を対象に、作成されるシート例を、シート 1 に示す。以下(1)～(10) は、状態遷移表の書式のルールである。これらのルールはシート 1 における青色の 1 ～ 10 に対応する。

- (1) 条件処理表の文字と、警告文の文字は MS ゴシック，11pt で表記する。
- (2) 関数は型と引数も含めて表記する。
- (3) 関数・条件・処理は左揃えで表記する。
- (4) ある条件において処理が存在しない場合，条件も処理も表記しない。
- (5) 同列の上下に同じ関数・条件のセルが隣接する場合，セル同士を結合してひとつにまとめる。
- (6) 条件処理表ごとに，その外枠を太線で囲む。
- (7) 警告文は文字色赤で，シート B 列 3 行目以降に表記する。
- (8) 条件処理表の各列の幅は 20 ポイントとする。セル内の文字列の長さが列の幅を超える場合，文字の折り返しはされずに表記される。
- (9) 処理にネストを含む構文が存在する場合，ネストの内側にある処理はインデントして他の処理と区別する。
- (10) ある処理における条件列の数が，条件処理表の最大条件列数に満たない場合は，該当セルに「無条件」と表記する。

シート 1

	A	B	C	D	E	F
1						
2						
3		WARNING: [関数 α] 警告が発生した場合，この行に表示される				
4		WARNING: [関数 β] 警告が発生した場合，この行に表示される				
5		関数 α	if (条件 C-1)	if (条件 C-11)	for (;;) {	
6					処理 P-1;	
7					}	
8						
9		関数 β	if (条件 C-2)	if (条件 C-21)	処理 P-2;	
10				else if (条件 C-22)		
11				無条件	処理 P-3;	
12						

ソースコード 1

```

1 関数 α {
2    if (条件 C-1) {
3        if (条件 C-11) {
4            for (;;) {
5                処理 P-1;
6            }
7        }
8    }
9 }

10 関数 β {
11    if (条件 C-2) {
12        if (条件 C-21) {
13            処理 P-2;
14        }
15        else if (条件 C-22) {
16        }
17        処理 P-3;
18    }
19 }

```

2.4 状態遷移表の構成要素

本ツールにおいてソースコードからリバース生成される状態遷移表（以降，本書では特に指定がない限り状態遷移表と呼ぶ）は，以下の 5 つの要素で構成される．

- A) 状態 事象に対して反応を変化させる状況
- B) イベント 対象としているシステムへの入力
- C) 処理 ある状況における事象に対して行う事柄
- D) 遷移 次の状況へと移行する事柄
- E) 状態変数 事象に対して状況が変化する対象

2.5 状態遷移表の読み方

状態遷移表は状態とイベント，交差するセルに存在する処理と遷移の組合せにより，プログラム実行の流れを表す．

状態遷移表に存在するイベント，処理，状態変数の更新は上から下に向かって順に実行されることを意味する．

また，状態の遷移は左から右に向かって順に実行されることを意味する．

イベントが発生し，状態変数がある状態をとるとき，その状態列とイベント行が交差するセルの処理，あるいは状態変数の更新が上から下に向かって順に実行される．

ただし，状態列とイベント行が交差するセルに状態変数の更新が存在する場合，そのセルにある処理と状態変数の更新がすべて実行された時点で状態変数の状態が遷移する．

以下，表 1 の状態遷移表を例にして読み方を示す．

なお，表 1 はソースコード 2 から作成された，状態変数 X に関する状態遷移表である．

- イベント E が発生し，状態変数 X が状態 S-1 であると，処理 P-11，処理 P-12，処理 P-13，遷移 T(S-2)が順に実行される．
- 遷移 T(S-2)まで実行されたあと，状態変数の状態 S-1 が状態 S-2 に遷移する．
- イベント E が発生し，状態変数 X が状態 S-2 であると，処理 P-21，遷移 T(S-1)，処理 P-22，処理 P-23 が順に実行される．
- 処理 P-23 まで実行されたあと，状態変数の状態 S-2 が状態 S-1 に遷移する．

表 1

イベント E	状態変数 X	
	状態 S-1	状態 S-2
	処理 P-11	処理 P-21
	処理 P-12	遷移 T(S-1)
	処理 P-13	処理 P-22
	遷移 T(S-2)	処理 P-23

ソースコード 2

```
1 if (イベント E) {  
2   if (状態変数 X==状態 S-1) {  
3     処理 P-11;  
4     処理 P-12;  
5     処理 P-13;  
6     遷移 T(S-2);  
7   }  
8   else if (状態変数 X==状態 S-2) {  
9     処理 P-21;  
10    遷移 T(S-1);  
11    処理 P-22;  
12    処理 P-23;  
13  }  
14 }
```

2.6 状態遷移表の書式

状態遷移表は、ユーザが選択した状態変数につき 1 枚の Excel シートに描画される。
状態遷移表は関数ごとに 1 つ作成される。

シートの構成は以下のとおりである。状態変数を含む関数 1 つにつき、例外時に警告文を表示する行、状態遷移表の行、空白行が用意される。

	B	C	D	E	F	...
3	B3		警告文			
4						
5			状態遷移表			
6						
...			空白			

ソースコード 3 を対象に、状態変数 X（グローバル変数）を選択した場合に作成されるシート例を、シート 2 に示す。以下(1)～(13) は、状態遷移表の書式のルールである。これらのルールはシート 2 における青色の 1 ～ 13 に対応する。

- (1) 状態遷移表の文字と、警告文の文字は MS ゴシック、11pt で表記する。
- (2) 関数は型と引数も含めて表記する。
- (3) 関数・状態・イベント・状態変数・処理・遷移はすべて左揃えで表記する。
- (4) ある状態・イベントにおいて処理・遷移が存在しない場合、処理に「N/A」と表記する。
- (5) 同列の上下に同じ関数・イベントのセルが隣接する場合、セル同士を結合してひとつにまとめる。
- (6) 状態遷移表ごとに、その外枠を太線で囲む。
- (7) 警告文は文字色赤で、シート B 列 3 行目以降に表記する。
- (8) 遷移のセルは背景色白、文字色赤で表記する。
- (9) 状態遷移表の各列の幅は 20 ポイントとする。セル内の文字列の長さが列の幅を超える場合、文字の折り返しはされずに表記される。（if 構文のブロックのセルのみ、セル内の改行を反映させるため、文字の折り返しをする）
- (10) 処理にネストを含む構文が存在する場合、ネストの内側にある処理はインデントして他の処理と区別する。
- (11) イベント、状態に else 文が含まれる場合、表の該当するセルには「else」と表記する。
- (12) ある処理・状態変数の更新に関するイベント列の数が、状態遷移表の最大イベント列数に満たない場合は、該当セルに「無条件」と表記する。

	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								

シート 2

ソースコード 3

```

1 関数 α {
2    if (イベント E-1) {
3      if (イベント E-11) {
4        if (状態変数 X ==状態 S-1) {
5          処理 P-1 ;
6          遷移 T(S-2) ;
7        }
8        else if (状態変数 X ==状態 S-2){
9          処理 P-2 ;
10         遷移 T(S-1) ;
11        }
12        else if (状態変数 X ==状態 S-3) {
13          }
14        }
15      else {
16      }
17    }
18    if (イベント E-2) {
19      処理 P-3 ;
20    }
21 関数 β {
22    if (イベント E-3) {
23      if (状態変数 X ==状態 S-1) {
24        for (;;) {
25          処理 P-4 ;
26        }
27      }
28      else {
29        処理 P-5;
30      }
31    }
32  }

```

3 機能要求

条件処理表を作成する機能，状態遷移表を作成する機能の仕様は「別紙 A.要求事項一覧」に記載する（ここに記載されている仕様は，要求仕様書作成時におけるワーキンググループ内での議論の結果であり，記載されている要求事項のすべてに対応しているわけではない）。

4 動作環境

本ツールの動作環境の仕様は「ユーザマニュアル.pdf」に記載する。