

# 仕様書

24G1007 網中洲

2024 年 07 月 21 日

## 1 プログラムの目的と概要

このプログラムは,pattern 配列に格納された複数のパターンを指定された文字で表示する. また, このプログラムの目的は, 指定された文字でパターンを視覚的に表示することで, パターンのデザインや配置の確認ができるようにすることである. 表 1 に関数リストを示す.

表 1 関数

関数名	仕様
main	pattern 配列に格納されたパターンを初期化し, 指定されたインデックスのパターンを順に表示する
set_pattern	与えられた二次元配列に特定のパターンを定義する
printx	指定されたピクセルとマスクを使用して, 指定された文字を表示する
printx1	指定されたパターン番号に対応するパターンを, 指定された文字で表示する

### 1.1 main 関数

図 1 に main 関数のフローチャートを示す. 次に関数の説明を行う.

関数名:main

役割: まず,pattern 配列の宣言と初期化を行う. 次に,pattern 配列の宣言と初期化を行い, パターンインデックス配列 str を宣言し, その内容に基づいてパターンを表示する.for ループを用いて str 配列の各要素を順に処理し, printx1 関数でパターンを表示する. 最後に,return 0; でプログラムを正常に終了する.

パラメータ: パラメータを受け取らない形式で, 内部で定義された配列 str に基づいてパターンを表示する.

返回值: 常に 0

### 1.2 set\_pattern 関数

図 2 に set\_pattern 関数のフローチャートを示す. 次に関数の説明を行う. これは, パターンデータを格納するために提供された 2 次元配列 pattern を初期化し, 各パターンは 8 バイトのデータとして表す. 関数名:set\_pattern

役割: 配列にパターンデータを設定する. 関数内で, この配列に適切なパターンが初期化する.

パラメータ 1:unsigned char 型 pattern[][8] という配列に設定する.

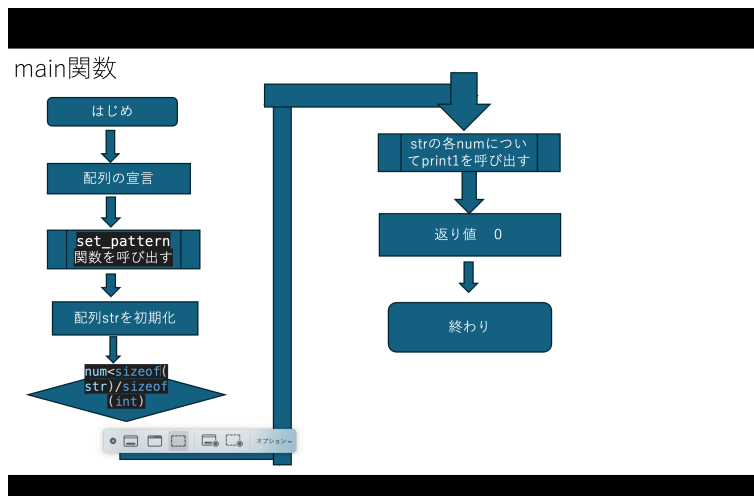


図1 main 関数

返回值: なし

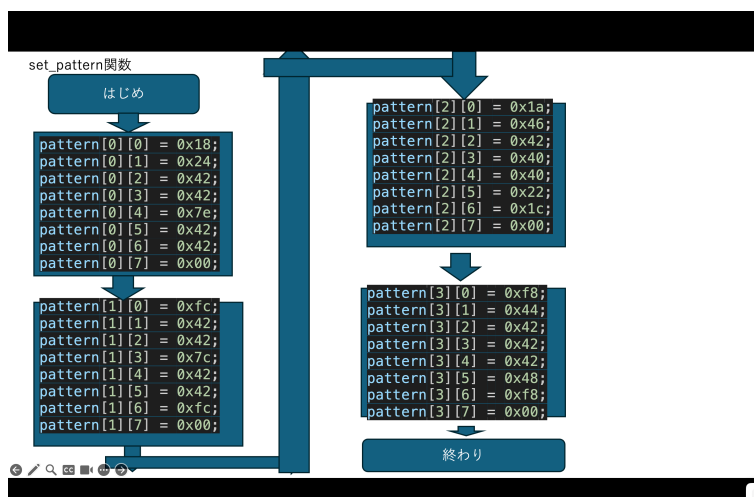


図2 set\_pattern 関数

### 1.3 printx 関数

図3にprintx関数のフローチャートを示す. 次に関数の説明を行う. ピクセルデータをマスクして, 指定された文字で表示する.

役割: 指定された pixel と mask を使用して, 指定された文字を表示する.

パラメータ 1: unsigned char pixel. これは, パターンデータの 1 バイトを表し, パターンの各ピクセルを表示する.

パラメータ 2: unsigned char mask. これは, パターンデータに適用するマスクであり, ピクセルデータをマスクするために使用され, 特定のビットをチェックして, どの部分が描画されるべきかを決定する.

パラメータ 3:unsigned char character. これは, 表示に使用する文字である. パターンを表示する際に使用する文字を指定し, この文字でピクセルが描画される.

返回值: なし

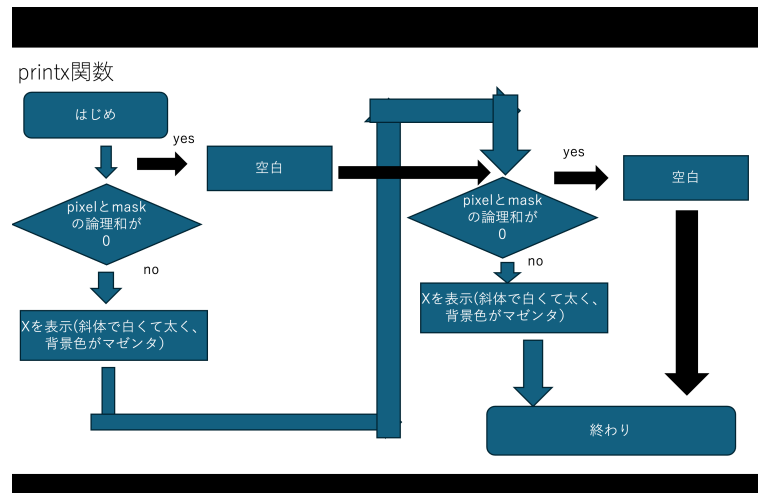


図3 printx 関数

## 1.4 printx1 関数

図4にprintx1関数のフローチャートを示す. 次に関数の説明を行う. 指定されたインデックス'number'のパターンを, 指定された文字'character'で表示する.

役割: 指定されたパターン番号に対応するパターンを, 指定された文字で表示する.

パラメータ 1:unsigned char pattern[][8]. これは, パターンデータを格納する2次元配列. 指定された number のパターンデータを取得するために使用する.

パラメータ 2:int number. これは, 表示するパターンのインデックス.pattern配列から取得するパターンデータのインデックスを指定する.

パラメータ 3:unsigned char character. これは, 表示に使用する文字である. パターンを表示する際に使用する文字を指定し, この文字でピクセルが描画される.

返回值: なし

## 2 分割コンパイル方法

まず, 各ソースファイルをオブジェクトファイルにコンパイルする. これにより,'reportpattern.o'と'pattern-lib.o'というオブジェクトファイルが生成される. 次に, 生成されたオブジェクトファイルをリンクして実行ファイルを生成する. これにより,beast.out という実行ファイルが生成される. 最後に, 生成された実行ファイルを実行する (./beast.out).

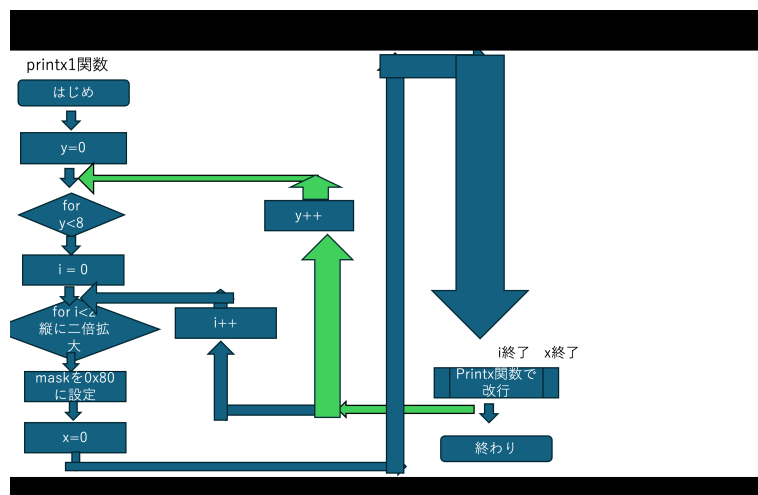


図 4 printx1 関数