

2025 年度 プログラミング III 第 1 回 レポート

学籍番号 36714029
遠藤 裕人

2025 年 10 月 2 日

1 はじめに

演習課題 1 の実行結果について報告します。

2 演習課題

2.1 2.1 課題 8

2.1.1 題名：何を作ったか

呪われた Todo リスト管理システムを作成した。このプログラムは通常のタスク管理機能に加えて、ランダムで発生する恐怖演出（文字化け、赤色表示、不気味なメッセージ）を含む、ホラー要素を持った Todo リストアプリケーションである。

2.1.2 どのような技術を使ったか

- 制御構文の利用: if 文、while 文、switch 文、for 文を使用
- 関数（自作）の利用: 20 個以上の関数を定義・利用
- ポインタの利用: 構造体ポインタを使用してタスク情報を操作
- 構造体の利用: Task、Date、TodoList の 3 つの構造体を定義
- ファイル操作: バイナリファイルによるデータの永続化
- 250 行以上: 530 行のコードで実装

2.1.3 実行例

2.1.4 プログラムの解説

1. タスク登録機能: タスクの内容、優先度 (Low/Medium/High)、期限を入力して登録
2. タスク編集機能: 登録済みのタスクの内容を編集、または削除
3. 月次タスク表示: 今月のタスクを表形式で一覧表示
4. 今日のタスク表示: 本日期限のタスクを目次形式で表示
5. ファイル永続化: タスクデータを task8.dat ファイルに保存・読み込み

恐怖演出として以下の機能を実装:

- タスク登録時に 10% の確率でタスク内容が文字化け
- 月次タスク表示時に 12.5% の確率で全ての線が赤色に変化
- 月次タスク表示時に 16.7% の確率で不気味なタスクが追加表示
- メニュー操作後に 20% の確率で恐怖メッセージを表示

2.1.5 コンパイル方法

以下のコマンドでコンパイル可能：

```
gcc task8.c -o task8
```

使用ライブラリ：

- stdio.h: 標準入出力
- stdlib.h: 標準ライブラリ (rand 関数等)
- string.h: 文字列操作 (strlen、strcspn 等)
- time.h: 日時取得 (time、localtime 等)

2.1.6 プログラムの動作 (自作関数の説明)

主要な自作関数の説明：

initialize_list(TodoList* list)

TodoList へのポインタ

出力：なし

動作：Todo リストを初期化し、全タスクを非アクティブ状態にする

add_task(TodoList* list)

TodoList へのポインタ

出力：なし

動作：ユーザーからタスク情報（内容、優先度、期限）を入力し、リストに追加する。10%の確率で corrupt_text 関数を呼び出し、タスク内容を文字化けさせる

edit_task(TodoList* list)

TodoList へのポインタ

出力：なし

動作：タスク ID を指定して、タスクの内容・優先度・期限を編集、または削除する

display_monthly_tasks(TodoList* list)

TodoList へのポインタ

出力：なし

動作：現在の月と同じ月のタスクを表示する。12.5%の確率で赤色表示になり、16.7% の確率で不気味なタスクが追加される

save_to_file(TodoList* list)

TodoList へのポインタ

出力：なし

動作：タスク情報をバイナリファイル (task8.dat) に保存する

load_from_file(TodoList* list)

TodoList へのポインタ

出力 : なし

動作 : task8.dat からタスク情報を読み込み、リストに復元する

corrupt_text(char* text)

文字列へのポインタ

出力 : なし

動作 : 文字列の各文字を 33% の確率でランダムな文字に置換し、文字化けを演出する

is_valid_date(Date* date)

Date へのポインタ

出力 : 有効な日付なら 1、無効なら 0

動作 : 年・月・日の妥当性をチェックし、うるう年も考慮する

2.1.7 感想

今回のプログラム作成で苦労した点と学んだ点 :

- 構造体とポインタを組み合わせた複雑なデータ構造の設計が難しかった。特に TodoList 構造体内に Task 配列を持たせ、それをポインタで操作する部分で理解が深まった。
- ファイル操作では、バイナリモードでの読み書きを使用することで、構造体をそのまま保存・読み込みできることを学んだ。
- 亂数を使った確率的な動作の実装が面白かった。rand 関数と srand 関数の使い方、特に time をシード値にすることで毎回異なる乱数を生成できることを理解した。
- 日付の妥当性チェックでは、うるう年の判定ロジックを実装し、暦の仕組みについても理解を深めることができた。月の日数を配列で管理する方法も学んだ。
- ANSI エスケープコードを使った色付き出力により、コンソールでも視覚的に面白い演出ができるることを知った。

学籍番号:36714029 氏名: 遠藤裕人