

# プログラミング言語論レポート

---

2025年8月7日

学籍番号：35714121

名前：福富隆大

## レポート8-1 (ラムダ式)

### 課題内容

- 2つの整数を引数としてとり、その和を返すラムダ式を定義し、そのテストコードを書け
- $f(\text{int } n)$  はラムダ式を返す関数である。返ってくるラムダ式は "A" を  $n$  回表示するものである。関数  $f$  を実装し、そのテストコードを書け

### 実装

- ファイル名: `report8-1.cpp`
- 2つの整数の和を求めるラムダ式と、"A" を指定回数表示するラムダ式を返す関数を実装
- 複数のテストケースで動作を確認

## レポート8-2 (STL ライブラリ関数を使う)

### 課題内容

- `std::for_each` で何かプログラムを書いてみよ。ただし、ラムダ式を使う場合と使わない場合の両方を試すこと
- `std::find_if` で何かプログラムを書いてみよ。ただし、ラムダ式を使う場合と使わない場合の両方を試すこと

### 実装

- ファイル名: `report8-2.cpp`
- `std::for_each` でリスト要素の変換表示（通常関数 vs ラムダ式）
- `std::find_if` で条件に合う要素の検索（通常関数 vs ラムダ式）

## レポート8-3 (高階関数)

### 課題内容

$\text{int } n$  と関数  $\text{double } f(\text{double } a)$  を引数として取り、関数  $f(\dots f(f(f(\dots)))\dots)$  (ただし  $f$  の適用は  $n$  回) を返す関数とテストコードを書け

### 実装

- ファイル名: `report8-3.cpp`
- 関数合成を行う高階関数 `compose_n_times` を実装

- square、add\_one、multiply\_by\_two関数での動作テスト
- ラムダ式を使ったテストも含む

## レポート9-1 (Haskellインタラクティブシェル)

### 課題内容

下記計算をインタラクティブシェルで実行し、計算結果を確かめよ:

- 基本計算: `2+3*4`, `(2+3)*4`, `sqrt(3^2 + 4^2)`
- 関数確認: `reverse "cse"`, `length [1,2,3,4]`, `succ 8`, `min 5, 6`, `max 100.5, 200.2`, `div 92 10`, `92 div 10`

### 実装

- ファイル名: `report9-1-results.txt`
- 各計算の予想結果と説明を記載
- 実際の実行は環境構築完了後に行う

## レポート9-2 (Haskell階乗プログラム)

### 課題内容

次のプログラムを打ち込み、コンパイルして動作を確認せよ（階乗計算をforループ無しで行った例）

### 実装

- ファイル名: `report9-2.hs`
- 再帰を使った階乗計算関数
- `fact 5`の結果を表示

## レポート10-1 (再帰とパターンマッチング)

### 課題内容

- フィボナッチ数列を計算するプログラムを書け
- `length`と同じ動作をする関数`my_length`を再帰とパターンマッチングを利用して実装せよ
- `sum`関数も同様にして、再帰とパターンマッチングを利用して実装せよ
- `reverse`関数も同様にせよ

### 実装

- ファイル名: `report10-1.hs`
- `fibonacci`, `my_length`, `my_sum`, `my_reverse`関数を実装
- 各関数の動作テストを含む

## レポート10-2 (リストの内包的表記)

### 課題内容

- 1から100の間の3の倍数のみを含むリストを作成せよ（リストの内包的表記を利用のこと）

- $1^2+2^2+3^2+\dots+1000^2$ を求めよ（リストの内包的表記を利用のこと）

## 実装

- ファイル名: `report10-2.hs`
- `multiplesOfThree`: 3の倍数リスト生成
- `sumOfSquares`: 1から1000の平方和計算

## レポート10-3 (Haskellにおける高階関数)

### 課題内容

1. `map`関数を自分で実装した`my_map`を作成せよ
2. `filter`関数を自分で実装した`my_filter`を作成せよ

## 実装

- ファイル名: `report10-3.hs`
- `my_map`: 再帰とパターンマッチングによる`map`関数の実装
- `my_filter`: 再帰とパターンマッチングによる`filter`関数の実装
- 標準関数との比較テストを含む

## まとめ

本レポートでは、C++のラムダ式・STLアルゴリズム・高階関数と、Haskellの基本構文・再帰・パターンマッチング・内包的表記・高階関数について実装と動作確認を行いました。関数型プログラミングの概念を両言語で比較学習することができました。