

情報科学実験レポート ocaml 基礎 1

6321120

横溝 尚也

提出日：5月 10日（火）

実行すると以下のような結果になる関数があります。この関数は、どのような動作をしているか考えよ。

1 課題1 : dellt

```
# dellt;;
- : int -> 'a list -> 'a list = <fun>
# dellt 0 [1;2;3;4];;
- : int list = [1; 2; 3; 4]
# dellt 1 [1;2;3;4];;
- : int list = [2; 3; 4]
# dellt 2 [1;2;3;4];;
- : int list = [3; 4]
# dellt 3 [1;2;3;4];;
- : int list = [4]
# dellt 5 [1;2;3;4];;
- : int list = []
# dellt 3 ["A"; "B"; "C"; "D"; "E"; "F"];;
- : string list = ["D"; "E"; "F"]
# dellt(-2) [1; 2];;
- : Exception: Failure "Error"
```

2 このプログラムの目的

この関数 `dellt` は、指定された任意の整数を受け取り、その整数値に対応する要素数を抜き取るプログラムである。このプログラムによって先頭からしか要素を抜き取ることができないという使いにくさもあるが、数を引数として指定するだけでその分のリストの要素を抜き取ることができるという優れたプログラムである。

3 課題2 : dellt2

```
#dellt2;;  
- : int -> ' a list -> ' a list = <fun>  
# dellt2 1 ["A"; "B"; "C"; "D"; "E"; "F"];;  
- : string list = ["B"; "C"; "D"; "E"; "F"]  
# dellt2 3 ["A"; "B"; "C"; "D"; "E"; "F"];;  
- : string list = ["A"; "B"; "D"; "E"; "F"]  
# dellt2 3 [1; 2; 3; 4; 5; 6];;  
- : int list = [1; 2; 4; 5; 6]
```

4 このプログラムの目的

関数 dellt は指定された要素数分抜き取るものであったが、今回の関数 dellt2 は、引数 n によって指定された要素 n 番目のみを抜き取るプログラムである。このプログラムは先頭からしか抜き出せなかったものがこのプログラムによって先頭以外の要素も抜き出すことが可能になった。

5 課題3 : find

```
# find;;  
- : ' a -> ' a list -> bool = <fun>  
# find 3 [1; 2; 3; 4; 5; 6];;  
- : bool = true  
# find 1 [2; 3; 4; 5];;  
- : bool = false
```

6 このプログラムの目的

関数 find はあるリストの中に指定した文字（数値も可能）が要素として含まれているのか判別するプログラムである。またこのプログラムは最終的な型が bool 型であり、正しいか正しくないかを判別するものである。このプログラムを応用すれば全生徒のテストの点数などを格納しておき、ある点数を取った人がいるのかどうか引数を指定するだけで判別が可能である。

7 課題4 : posl

```
# posl;;  
- : int -> ' a list -> ' a = <fun>  
# posl 3 ["AB"; "C"; "DEF"; "G"; "H"; "IJ"];;  
- : string = "DEF"  
# posl 2 [ 1; 2; 3; 4; 5];;  
- : int = 2  
# posl 0 [ 1; 2; 3; 4; 5];;  
Exception: Failure "Not Exist...".
```

8 このプログラムの目的

関数 posl はあるリストの中から指定された要素を抽出するものである。関数 dellt では指定された要素をリストから抜き取りリストを出力するものであったが、今回は指定された要素のみを出力するものである。このプログラムによってたくさんのデータの中から一つの絞られたデータのみを抽出することが可能になった。

9 課題5 : divlist

```
# divlist;;  
- : int list -> int list -> int list = <fun>  
# divlist [12; 8; 9; 7] [2; 4; 3; 7];;  
- : int list = [6; 2; 3; 1]
```

10 このプログラムの目的

関数 divlist は二つのリストの対応する要素の商を計算し、その結果を新たなリストとして出力するプログラムである。

11 課題6 : mul2list

```
# mul2list;;  
- : int list -> int list = <fun>  
# mul2list [1; 2];;  
- : int list = [2]  
# mul2list [1; 2; 3];;  
- : int list = [2; 6]  
# mul2list [1; 2; 3; 4; 5];;  
- : int list = [2; 6; 12; 20]
```

12 このプログラムの目的

関数 `mul2list` はあるリストに格納されている隣り合わせの要素の積を新たなリストとして出力するプログラムである。このプログラムを複数回使用することでリストに含まれているすべての要素の積も表すことができおる。

13 課題 7 : chglist

```
# chglist;;  
- : ' a * ' a -> ' a list -> ' a list = <fun>  
# chglist ("A", "*") ["1"; "A"; "2"; "B"; "A"; "A"; "3"; "4"];;  
- : string list = ["1"; "*"; "2"; "B"; "*"; "*"; "3"; "4"]
```

14 このプログラムの目的

関数 chglist はあるリストの中から指定した文字（数値も可）を別の文字にすべて置き換え、そのリストを出力するプログラムである。このプログラムを利用することである項目のみ非表示にした時などもその項目を指定し、変換することで変換することもできる。

15 課題8 : inslist

```
# inslist;;
- : int -> 'a -> 'a list -> 'a list = <fun>
# inslist 2 "*" ["A"; "B"; "C"; "D"; "E"];;
- : string list = ["A"; "*"; "B"; "C"; "D"; "E"]
# inslist 6 "*" ["A"; "B"; "C"; "D"; "E"];;
- : string list = ["A"; "B"; "C"; "D"; "E"; "*"]
# inslist 1 "+" [];;
- : string list = ["+"]
# inslist 1 "+" ["A"];;
- : string list = ["+"; "A"]
# inslist 0 "+" ["A"];;
Exception: Failure "Error".
```

16 このプログラムの目的

関数 `inslist` はあるリストの中へと指定した文字を好きな位置に要素として追加することができる。

17 課題9：replicate

```
# replicate;;  
- : int -> 'a -> 'a list = <fun>  
# replicate 3 ["A"];;  
- : string list list = [["A"]; ["A"]; ["A"]]  
# replicate 5 "A";;  
- : string list = ["A"; "A"; "A"; "A"; "A"]  
# replicate 3 ["1"; "#"];;  
- : string list list = [["1"; "#"]; ["1"; "#"]; ["1"; "#"]]
```

18 このプログラムの目的

関数 replicate はあるリストを一つの要素として指定された要素数分同じ要素をを持つリストを出力するプログラムである。

19 課題 10 : merge

```
# merge;;  
- : ' a list -> ' a list -> ' a list = <fun>  
# merge [1; 2; 3] [4; 5; 6];;  
- : int list = [1; 4; 2; 5; 3; 6]  
# merge ["A"; "B"] [ "C"; "D"; "EF"; "GH"];;  
- : string list = ["A"; "C"; "B"; "D"; "EF"; "GH"]
```

20 このプログラムの目的

関数 merge は二つのリストから左からひとつずつ順番に新たなリストの要素として格納していくプログラムである。

21 課題 1.1 : *insidelength*

```
# insidelength;;  
- : 'a list list -> int = <fun>  
# insidelength[[1; 2; 3]; [4; 5]; [6]; [7; 8; 9; 10]];;  
- : int = 10  
# insidelength[["A"; "B"]; [ "C"; "D"]; ["EF"; "GH"]];;  
- : int = 6
```

22 このプログラムの目的

関数 *insidelength* はリストの中のリストの要素数をすべて数え上げるプログラムである授業内ではリストの要素数を数え上げるプログラムは学習済みであるが、今回の課題ではリストの中にリストが含まれているときの要素数を数え上げることができる。

23 課題 1 2 : concat

```
# concat;;  
- : 'a list list -> 'a list = <fun>  
# concat [[0; 3; 4]; [2]; [0]; [5; 0]];;  
- : int list = [0; 3; 4; 2; 0; 5; 0]
```

24 このプログラムの目的

関数 `concat` はリストの中にリストが含まれている要素をすべて結合して一つのリストにするプログラムである。このプログラムによってリスト構造が一つなくなり簡潔に表すことができる。

25 課題 13 : assoc

```
# assoc;;
- : 'a -> ('a * 'a) list -> 'a = <fun>
# assoc 33 [(3,4); (33,5); (11,2); (55,1)];;
- : int = 5
# assoc 2 [(3,4); (33,5); (11,2); (55,1)];;
- : int = 11
# assoc "03" [("Kyoto", "075"); ("Osaka", "06"); ("Tokyo", "03")];;
- : string = "Tokyo"
# assoc "Kyoto" [("Kyoto", "075"); ("Osaka", "06"); ("Tokyo", "03")];;
- : string = "075"
# assoc 6 [(3,4); (33,5); (11,2); (55,1)];;
Exception: Failure "Not found...".
```

26 このプログラムの目的

関数 `assoc` は二つの要素を持つ組を要素に持つリストに対して指定された文字（数値も可）が組の中に含まれている場合、その組のもう片方の要素を抽出するプログラムである。

27 課題 1 4 : maximum

```
# maximum;;  
- : 'a list -> 'a = <fun>  
# maximum [3; 2; 5; 1];;  
- : int = 5  
# maximum ["abc"; "sdf"];;  
- : string = "sdf"  
# maximum [];;  
Exception: Failure "Error".
```

28 このプログラムの目的

関数 maximum はリストの要素の中で最大のものを抽出するプログラムである。また文字列の場合はプログラムによって一度 ASCII コードに変換され数の大小を比較していると考えられる。

29 課題 15 : index

```
# index;;  
- : ' a list -> ' a -> int = <fun>  
# index [21; 2; 31; 1] 21;;  
- : int = 0  
# index ['a'; '3'; 'b'; 'z'; '1'] 'z';;  
- : int = 3
```

30 このプログラムの目的

関数 index はあるリストに対して指定された要素が何番目に位置しているかを出力するプログラムである。

31 課題 16：集合の計算

リストを集合と見なして，以下の集合計算をする関数を定義してください．

- 1．二つの集合の積（共通要素）を返す関数: `inter`
 - 2．二つの集合の和を返す関数: `union`
 - 3．二つの集合の差を返す関数: `diff`
- ただし各集合は同じ要素を持たないこととします．

32 プログラムファイル名

`oc_report1.ml`