苏州城市学院 计算机组成与结构课程 期末试卷 A

Suzhou City University
IOT Program

CS 210 Programming Languages (Spring 2022)

Format: 110 minutes, 100 points, closed notes, closed book, no calculator permitted.
Please show all the work and procedures to receive partial credit!

Name: 李名佳 Student ID: 2017494095

1. (10 points) Please briefly describe the differences between coercion and overloading. Are they both considered as kinds of polymorphism?

difference: a coercion is an implicit type conversion. supplied automatically and coercion use definition to choose Conversion, but overloading at least two definitions advanced. and use types to choose the definition.

Yes. they are kinds of polymorphism.

2. (10 points) For each of the following programs, give the value that **ans** is bound to after evaluation.

1)
```
val x = 1;
val y = x + 1;   2
val x = y + 1;   3
val ans = x + y;
```

2)
```
val x = 1;
fun f y = x;
val x = (f 3)1 + (f 2);1   2
val ans = f x;   f2
```

1) ans = 5

2) ans = 2

3. (15 points) Read the following ML function definition:

12 3

```
- fun final1 x =
=   if null x then 0
=   else hd x + final1(tl x);
```

1+2+3

1) What is the purpose of the function **final1**?
2) What does the "**null x**" do? Can "**null x**" be replaced by "**x = []**"? Why or why not?
3) What is the function type of **final1**?
4) What is the return value after calling the function shown below?

   final1 [5,2,0,4,5];

1) Add up each of these elements of x

2) null x is a predefind function, it is used to judge whether it is empty.

No, null x can't be replaced by "x=[]". because. it avoids unnecessary type restrictions, reals can't be test to equality.

(3) val final1 = fn: 'a list -> int

(4) 16

4. (15 points)

1) Write a function `final2` of type `int list -> int list` that takes a list of integers and returns the list of all the odd elements from the original list (in the original order). For example, if you evaluate `final2 [1,2,3,4]` you should get `[1,3]`.
2) Write a function `final3` of type `int list -> char list` that takes a list of integers and returns the list of charaters. For example, if you evaluate `final3 [65,66,67,68]` you should get `[#”A”,#”B”, #”C”, #”D”]`.
3) Write a function `final4` of type `bool list -> bool` that takes a list of boolean values and returns the logical `AND` of all of them. If the list is empty, your function should return `true`.

```
1) —funal2 x= If null x then []
   else
        let val a= hd x
        in
          If a mod 2 =1 then [ a ] @ final2 tl(x)
          else find2 tl(x)
        end;

(2) —funal3 x = map.chr x ;

(3) — fun funal4 x =
        foldl (fn(a.b)=> a andalso b) true x;
```

5. (15 points) The function `final5` is defined as follows:

```
    fun final5 nil = (nil, nil)
    |   final5 [a] = (nil,[a])
    |   final5 (a::b::cs) =
          let
            val (x, y) = final5 cs
          in
            (a::x, b::y)
          end;
  val final5 = fn : 'a list -> 'a list * 'a list
```

Please write the returned value for each function call indicated below.

1) `final5 [4];`

2) `final5 [14,24];`

3) `final5 [14,15,24,25,34,35];`

1) (nil , [4])

2) ([14] , [24])

3) ([14, 24, 34] , [15, 25, 35])

6. (10 points) Give the ML type corresponding to each of the following sets:

   1) {true, false}
   2) {true,false} -> {true,false}
   3) {(true,false), (false,true), (true,false), (false,true)}

i) bool

2) bool -> bool

3). bool * bool

7. (10 points) Consider an unknown language with a left-associative + operator that is overloaded to have the following types: `int * real -> real`, `int * int -> int`, `real * int -> real`, and `real * real -> real`. Suppose the variable `i` has type `int` and the variable `r` has type `real`. For each + operator in each of the following expressions, say which type of + is used:

1) `i+r`
2) `r+i+r`
3) `i+(r+i)`
4) `i+i+r+(r+i)`

1) int * real -> real

2) real * int -> real , real * real -> real

3) real * int -> real , int * real -> real

4) real * int -> real , int * int -> int
int * real -> real , real * real -> real

8. (15 points) What is the order of functions with each of the following ML types?

```
1) int * int list -> bool
2) int list * (int * int -> bool) -> int list
3) int -> int -> int-> int -> int-> bool list
4) (int -> int) * (int -> int) * (int -> int) -> int
5) int -> string
```

1) 1

2) 2

3) 5

4) 2

5) 1