Quiz 03 - 03/28/24

Thank you. Your instructor has reviewed your responses. Here are your results.

Score Summary			
(Click on question number to jump to question.)		**points earned**	**points possible**
Question 1	correct	1	1
Question 2	correct	1	1
Question 3	correct	1	1
Question 4	correct	1	1
Question 5	correct	1	1
Question 6	correct	1	1
Question 7	correct	1	1
Question 8	correct	1	1
Question 9	correct	1	1
Question 10	correct	1	1
Question 11	correct	1	1
Question 12	correct	1	1
Question 13	correct	1	1
Question 14	correct	1	1
Question 15	correct	1	1
Question 16	correct	1	1
Question 17	incorrect	0	1
Question 18	correct	1	1
Question 19	correct	1	1
Question 20	correct	1	1
	Score: (95%)	**19**	**20**

1. Given this function definition:

```
fun rev xs =
   let fun aux(xs, acc) =
            case xs of
                 [ ] => acc
               | x :: xs' => aux (xs', x::acc)
   in
      aux(xs, [ ])
   end
```

During the call:

```
val res = rev [1, 2, 3, 4];
```

the aux helper is called 5 times before binding res to [4, 3, 2, 1]. On the third call to aux, what is the value of xs?

- xs = [3, 4] (correct answer, your response)

- xs = [ ]
- xs = [1]
- xs = [2, 3, 4]

Feedback: Correct

Points earned: 1 out of 1

2. `fold` is **tail recursive**

```
fun fold f acc lst =
     case lst of
       [ ] => acc
     | hd::tl => fold f (f(acc,hd)) tl
```

- True (correct answer, your response)
- False

Feedback: Correct

Points earned: 1 out of 1

3. Given this function definition:

```
fun rev xs =
   let fun aux(xs, acc) =
             case xs of
                  [ ] => acc
                | x :: xs' => aux (xs', x::acc)
   in
      aux(xs, [ ])
   end
```

During the call:

```
val res = rev [1, 2, 3, 4];
```

the aux helper is called 5 times before binding res to [4, 3, 2, 1]. On the fourth call to aux, what is the value of xs?

- xs = [4] (correct answer, your response)
- xs = [3, 4]
- xs = [1]
- xs = [2, 3, 4]

Feedback: Correct

Points earned: 1 out of 1

4. What is the type of the following function?

```
fun mystery (xs) =
    if null xs
```

```
    then 1
    else hd(xs) * mystery(tl xs)
```

The following answers are acceptable:

- int list -> int
- int list->int

Your response:

int list -> int

Feedback: Correct.

Points earned: 1 out of 1

5. Given the following ML statement, what is `f` bound to?

```
val (e,(f,g)) = (1,(2,3));
```

The following answer is acceptable:

2

Your response:

2

Points earned: 1 out of 1

6. Given the following ML statement, what is `p` bound to?

```
val (p, _ ) = (12, 10);
```

The following answer is acceptable:

12

Your response:

12

Points earned: 1 out of 1

7. What is the type of the following function?

```
fun mystery x =
    case x of
          (1, _, _) => true
        | (a, _, c) => a > c
```

The following answers are acceptable:

- int * 'a * int -> bool
- int*'a*int->bool

Your response:

int * 'a * int -> bool

Points earned: 1 out of 1

8. What is the type of the following function?

```
fun mystery x =
    case x of
          (1, b, c) => b < c
        | (a, b, c) => a + b > c
```

- int * int * int -> bool (correct answer, your response)
- This function will not typecheck
- int -> int -> int -> bool
- int * 'a * int -> bool

Feedback: Correct.

Points earned: 1 out of 1

9. What is **ans** bound to after the following ML code is evaluated?

```
datatype my_string_list = Nothing | Something of string * my_string_list

exception Foo

fun f (lst,n) =
    if n<=0
    then Nothing
    else case lst of
           Nothing => raise Foo
      | Something(s, lst) => Something(s, f(lst, n-1))

val ans = f(Something("x",Something("y",Nothing)), 2)
```

Your response:

Something("x", Something ("y",Nothing))

Sample answer:

Something ("x", Something ("y", Nothing))

Answers may vary.

Feedback for correct answers: Correct

Feedback for incorrect answers: Given a `my_string_list lst` and a postive number `n`, `f` returns a `my_string_list` that contains the first `n` elements of `lst`. For example, `f(Something("x",Something("y",Nothing)), n)` evaluates to `Something("x",Nothing)` if `n` is 1 and `Nothing` if `n` is 0. If `n` is greater than the length of `lst` (i.e., the number of `Something`

constructors in the value bound to `lst`), then `f` raises the exception `Foo`. If `f` is passed a non-positive number, it returns `Nothing`.

Points earned: 1 out of 1
Instructor's comments:

10. What is the type of the following function?

```
fun mystery (x, y) =
    if y=0
    then true
    else x andalso mystery(x,y-1)
```

The following answers are acceptable:

- bool * int -> bool
- bool*int->bool
- bool*int -> bool
- (bool * int) -> bool
- (bool*int)->bool
- (bool*int) -> bool

Your response:
bool * int -> bool

Points earned: 1 out of 1

11. Given this definition for factorial:

```
fun factorial n =
   let fun aux(n, acc) =
      if n = 0
      then acc
      else aux(n-1, acc*n)
   in
      aux(n, 1)
   end
```

Which of the following calls to aux is equivalent to factorial 5? In other words, which of the following calls results in the same as factorial 5? Hint: fact 5 = 120

- aux(3, 20) (correct answer, your response)
- aux(5, 0)
- aux(1, 24)
- aux(2, 120)

Feedback: Correct

Points earned: 1 out of 1

12. Given the following ML statement, what is q bound to?

```
val (q, 10 ) = (9, 10);
```

The following answer is acceptable:

9

Your response:

9

Points earned: 1 out of 1

13. True or False: Function rev is tail recursive.

```
fun rev lst =
   case lst of
       [ ] => [ ]
     | x::xs => (rev xs) @ [x]
```

- True
- False (correct answer, your response)

Feedback: Correct

Points earned: 1 out of 1

14. What is **ans** bound to after the following ML code is evaluated?

```
fun m2 lst =
      let fun loop (lst1,lst2) =
         case lst1 of
           [ ] => lst2
         | x::[ ] => x::lst2
         | x::(_::z) => loop(z,x::lst2)
       in
         loop(lst,[ ])
       end

val ans = m2 [1, 2, 3, 4]
```

The following answers are acceptable:

- [3,1]
- [3, 1]

Your response:

[3,1]

Feedback: Correct.

Points earned: 1 out of 1

15. Given the following ML statement, what is a bound to?

```
val (a,b) = (5,6);
```

The following answer is acceptable:

5

Your response:

5

Points earned: 1 out of 1

16. What is x bound to after running the following code?

```
exception MyException of int

fun f n =
    if n = 0
    then raise List.Empty
    else if n = 1
          then raise (MyException 4)
          else n * n

val x = (f 1 handle List.Empty => 42) handle MyException n => f n
```

- 16 (correct answer, your response)
- 2
- 4
- 1

Feedback: Correct

Points earned: 1 out of 1

17. What is the type of **majority**?

```
fun majority f lst =
    let fun vote lst =
        case lst of
        [ ] => 0
      | hd::tl => (if f hd then 1 else ~1) + (vote tl)
    in
        (vote lst) > 0
    end
```

Your response:

'a list -> bool

Sample answer:

('a -> bool) -> 'a list -> bool

Answers may vary.

Feedback for correct answers: Correct

Points earned: 0 out of 1

Instructor's comments:

18. What is **ans** bound to after the following ML code is evaluated?

```
exception E
val x = 1
fun f x = if x=2 then raise E else 14
val x = 2
val ans = ((f x) + 4) handle E => 9
```

The following answer is acceptable:

9

Your response:

9

Feedback: Correct

Points earned: 1 out of 1

19. Given the following ML statement, what is `d` bound to?

```
val (c,d) = (2,("xx","yy"));
```

The following answers are acceptable:

- ("xx", "yy")
- ("xx","yy")

Your response:

("xx", "yy")

Points earned: 1 out of 1

20. What is res bound to after running the following code?

```
val x = List.Empty
val res = (hd [ ], 0) handle List.Empty => raise x;
```

- Nothing, a run-time exception halts the program when binding res (line 2) (correct answer, your response)
- Nothing, a run-time exception halts the program when binding x (line 1)
- [ ]

- 0

Feedback: Correct.

Points earned: 1 out of 1