

## 作业1

### 1

成功;

成功;

当L的成员大于2时成功;

失败;

L的成员等于2时成功;

### 2

[x, y, z]

没有, 可以拆分为lists of length 2 和 lists of length 3两种方式描述。

(x, y)::L

(x::L1, y::L2)

### 3

第4行的x: 类型为int, 值为2

第5行的m: 类型为real, 值为12.4

第6行的x: 类型为int, 值为9001

第14行的计算结果: 27

### 4

错误以红色标注如下：

```
(* pi: real *)  
val pi : real = 3.14159;
```

```
(* fact: int -> int *)  
fun fact (0 : int) : int = 1  
  | fact n = n * (fact (n - 1));
```

```
(* f : int -> int *)  
fun f (3 : int) : int = 9  
  f _ = 4;
```

```
(* circ : real -> real *)  
fun circ (r : real) : real = 2 * pi * r
```

```
(* semicirc : real -> real *)  
fun semicirc : real = pie * r
```

```
(* area : real -> real *)  
fun area (r : int) : real = pi * r * r
```

## 5

```
3 + 4;  
( val it = 7 )
```

```
3 + 2.0;  
( 错误, real和int类型不匹配, 改成3.0 + 2.0 )
```

```
it + 6;  
( val it = 13 )
```

```
val it = "hello";  
( val it = "hello" )
```

```
it + "world";  
( 错误, 字符串类型没有+操作符, 要用^操作符拼接 )
```

```
it + 5;  
( 错误, string和int类型不匹配 )
```

```
val a = 5;  
( val a = 5 )
```

```
a = 6;  
( val it = false )
```

```
a + 8;  
( val a = 13 )
```

```
val twice = (fn x => 2 * x);  
( val twice = fn : int -> int )
```

```
twice a;  
( val it = 10 )
```

```
let x = 1 in x end;  
( 错误, x=1是bool值, 给x赋值要用val x = 1 )
```

```
foo;  
( 错误, foo未定义 )
```

```
[1,"foo"];  
( 错误, list里面不能包含不同类型的成员 )
```