SML - PRACTICE EXERCISES

Exercise 1

Explain what is wrong in the following expressions and propose a correction.

```
hd([]);
explode (["toto"]);
implode("a","b");
["t] :: ["o","p"];
6 @ 10;
```

Exercise 2

(3,4) and (3,4,5) have the same type? [3,4] and [3,4,5] have the same type?

Exercise 3

Consider the following definitions:

```
fun fact n = if n=0 then 1
                    else n * fact(n-1);

fun new_if (a,b,c) = if a then b else c;
```

Write a function **new_fact** using **new_if**.
Explain why **new_fact** does not compute the factorial.

Note: What is the evaluation of recursive function in SML?

Exercise 4

Link the variable x to the value 0 when constructing forms to match the following expressions.

For example given the expression **(false,"bonjour",0)** the form (_,_,x) permits us to link x to 0 when we write: val (_,_,x) = (false,"bonjour",0).

```
{a=3,b=0,c=false} - record.
[~2,~1,0,1,2] - ~ unary minus.
[(3,1),(0,9)]
```

Exercise 5

Write a function **power_of_two** that tests if an int is a power of 2.

Write each steps of the evaluation of **(power_of_two 8)**.

Exercise 6

What is the type of the following function , justify your answer.

```
fun f (x,y,z,t) =
    if x=y then z+1
           else if x > y then z else y+t;
```

Exercise 7

Write 2 functions **odd** and **even** that define if an int is even or odd using mutual recursion.

Exercise 8

What are the results of the following declarations and expression. Each one is independent.

```
val x = 2 and y = x+1;

val x = 1; local val x = 2 in val y = x+1 end; val z = x + 1;

let val x = 1 in let val x = 2 and y = x in x + y end end ;
```

Exercise 9

What are the results of the following expressions evaluations.

```
val x = 1 and y = 2 and z = 3;

let val x = x+1 and z=x+4 in x+z end;

let val t = x+1 in let val x = x+1 in x end end;
```

Exercise 10

Write a function **insert** that inserts an int in a (ascending) sorted list.

Exercise 11

Write a function **interclass** that interclasses 2 lists of (ascending) sorted int.

Exercise 12

Write a function **insertion_sort** that implements insertion sorting.

Exercise 13

Bubble sort

1. Define a function **iteration** that repeat the treatment of a data while a condition on this data is not true.
2. Define a function **is_sorted** that returns true if a list is sorted, false otherwise.
3. Write a function **buble** that implements the Bubble sort.

Exercise 14

Write a function that computes the subsets of a set. How to represent a set?

Exercise 15

What is the type of C:

```
fun C f g x = f (g x);
```

Exercise 16

1. Write a function F that takes 2 parameters: a function O and a list l and processes the following way:

   $F(O, l) = O(a_1, O(a_2, O(a_3, ..., O(a_{n-1}, a_n)...)$ où $l = [a_1, ..., a_n]$.

   The list l has 2 or more elements.

2. Write a function G that returns the elements of a list l that satisfy the condition *cond*. What is the type of G? Why?

3. Using F and the function max that returns the maximum of 2 integers (write max) what is the maximum of a list of int, for example $[2, 6, 3, 15, 18, 1, 55, 22]$)?

4. Using F and the function *conc* that returns the concatenation of 2 strings (write conc) what is the concatenation of all the strings of a list, for example $["a", "b", "c", "d"]$?

5. Consider the function $fold$. What is its type?

   ```
   fun fold F nil y = y
   | fold F (x::l) y = F(x,(fold F l y));
   ```

Exercise 17

Consider the function f:

```
fun f (x,nil) = nil
| f (x,a::aa) = if x(a) then a::f(x,aa) else f(x,aa);
```

Let T_e be the type of an expression e. We construct using f the following system of equations.

```
(1) T_f = 'a -> 'b
(2) 'a = T_x * T_{nil}
(3) 'a = T_x * T_{a::aa}
(4) 'b = T_{nil}
(5) 'b = T_{f(x,aa)}
(6) T_{x(a)} = bool
```

1. Justify each line of this system of equations.
2. Compute the type of f.

Exercise 18

Let consider f:

```
fun f (x,nil) = nil
| f (x,a::aa) = if x(a) then x(a)::f(x,aa) else f(x,aa);
```

Let T_e be the type of an expression e. We construct a set of equations from the definition of f.

Write this system and compute the type of f.

Exercise 19

1. Write a *datatype* **COORDS** that defines the coordinates of a point in 3D.
2. Give examples of the use of COORDS.
3. Using **COORDS** write a function *distance* that computes the distance between 2 points.
4. Give an examples of the use of distance.

Exercise 20

Create a datatype PERSON that defines a person defined by its name, fname, age and datebirth.

Exercise 21

1. Create a reference variable i whose value is a reference to 10.
2. Increment the value of i of 1.
3. Decrement the value of i of 1.
4. Change the value of i to 20.

Exercise 22

while $<$ *expression* $>$ do $<$ *expression* $>$ has the following semantics in SML:

a. Evaluate the first expression.

b. If the first expression is false, exit. Else, evaluate the second expression and go to step a.

We want to code the following algorithm in SML:

```
i = 1
while i <= 10
do
afficher i
i= i+1
end
```

1. Why is the use of references needed?
2. Write the SML code.

Exercise 23

Write a function *factorial* that returns:

– 1 for factorial(0)

– generates an exception for a negative parameter and return 0

– $n!$ for a strictly positive parameter.