Module 2
CS 4450
Mason Fernandez

Do the following 12 problems from Chapter 5 in the book: 3–11, 13–15

You should only need **hd**, **tl**, **::**, **[ ]**, and **explode**, as far as built-in functions go.

Do not use any function call preceded by "List.".

(see m2.sml)

```
PS C:\Users\Mason\OneDrive\Desktop\UVU\03-Spring-2025\4450-Languages> sml m2.sml
Standard ML of New Jersey [Version 110.99.6.1; 32-bit; October 25, 2024]
[opening m2.sml]
m2.sml:5.5-5.42 Warning: match nonexhaustive
          a :: b :: c :: d :: rest => ...

val fourth = fn : 'a list -> 'a
val it = 4 : int
val min3 = fn : int * int * int -> int
val it = 2 : int
val red3 = fn : 'a * 'b * 'c -> 'a * 'c
val it = (1,3) : int * int
val thirds = fn : string -> char
val it = #"c" : char
val cycle1 = fn : 'a list -> 'a list
val testCycle = [1,2,3,4,5] : int list
val it = [2,3,4,5,1] : int list
val sort3 = fn : real * real * real -> real list
val testSort = (3,1,2) : real * real * real
val it = [1,2,3] : real list
m2.sml:47.5-47.48 Warning: match nonexhaustive
          x :: y :: z :: rest => ...

val del3 = fn : 'a list -> 'a list
val testDel3 = [1,2,3,4,5,6] : int list
val it = [1,2,4,5,6] : int list
val sqsum = fn : int -> int
val it = 55 : int
val cycle = fn : 'a list * int -> 'a list
val it = [3,4,5,1,2] : int list
val maxhelper = fn : int list * int -> int
val max = fn : int list -> int
val maxTest = [1,2,3,4,90,3,89,0,0,1,1] : int list
val it = 90 : int
val isPrimeHelper = fn : int * int -> bool
val isPrime = fn : int -> bool
val it = true : bool
m2.sml:92.13 Warning: calling polyEqual
m2.sml:91.5-97.21 Warning: match nonexhaustive
          (x :: rest,f) => ...

val select = fn : ''a list * (''a -> bool) -> ''a list
val it = [2,3,5,7] : int list
```

Do the following problems from Chapter 6 in the book, plus the additional problem below: 1, 2, 4 (part g should read **X := X + Y;**)

1. Let X and Y be any two sets, and let |X| and |Y| be their sizes. What is the size of each of the following sets? If you cannot determine the answer exactly, try to give upper and lower bounds.

a. X union Y = X+Y-(X intersection Y)

b. X x Y = X*Y

c. X^n = X^n

d. X* = 0 for an empty set, 1 for a set of one, and infinite for a set >1

2. Give the ML type corresponding to each of the following sets:

a. {true,false} bool

b. {true,false} -> {true,false} fn: bool -> bool

c. {(true , true) , (true , false) , (false , true) , (false , false)} bool * bool

Note for #2, you are given a set in each part of the problem. Write the ML type constructor that corresponds to that set. The curly braces are not ML records–they're just sets.

4. Suppose there are three variables X, Y, and Z with these types: X: integer that is divisible by 3 Y: integer that is divisible by 12 Z: integer. For each of the following assignments, knowing nothing about the values of the variables except their types, answer whether a language system can tell before running the program whether the assignment is safe? Why or why not?

Z least strict, then Y/12, then X/3

a. X := Y, safe because anything divisible by three is also divisible by 12

b. X := X safe because its the same type

c. Y := Y + 1 not safe because adding one may make it no longer divisible by three

d. Z := X safe because x is a subset of z

e. X := Z not safe because not all integers are divisible by 3

f. X := X + 3 safe because three is divisible by three

g. X := X + Y safe because y is a subset of x

Additional problem: Write a short C/C++ program that reads a 64-bit machine instruction as a hex integer and extracts the values for its components from certain bits, specified left-to-right, as follows:

(see m2.cpp)

For all programming problems, be sure to submit your source code and a copy/screen shot of the execution output. No executables or project files, please.

```
PS C:\Users\Mason>  & 'c:\Users\Mason\.vscode\extensions\ms-vscode.cpptools-1.23.5-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEng
In-xvg5zvyx.5tx' '--stdout=Microsoft-MIEngine-Out-xix105k1.eei' '--stderr=Microsoft-MIEngine-Error-uuhxdp4b.gbx' '--pid=Microsoft-MIEngine-Pid-b5zhhdfj.h2u' '--e
e=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
08800080000004D2
Instruction: 8800080000004d2

rreg = 0
lreg = 2
si = 0
radrm = 0
lardm = 1
code = 1
li = 1234


08800080000004D2
Instruction: 8800080000004d2

rreg = 0
lreg = 2
si = 0
radrm = 0
lardm = 1
code = 1
li = 1234


E05000000000008E
Instruction: e05000000000008e

rreg = 0
lreg = 0
si = 0
radrm = 10
lardm = 0
code = 28
li = 142
```

```
1140408300000000
Instruction: 1140408300000000

rreg = 3
lreg = 2
si = 4
radrm = 8
lardm = 2
code = 2
li = 0


5008000300000000
Instruction: 5008000300000000

rreg = 3
lreg = 0
si = 0
radrm = 1
lardm = 0
code = 10
li = 0


E800000000000020
Instruction: e800000000000020

rreg = 0
lreg = 0
si = 0
radrm = 0
lardm = 0
code = 29
li = 32
```