1.1) Special forms required In programming languages because they have different evaluation rules than regular functions, so they akllow us to create new constructs that cannot be expressed using existing functions and operators. For example the IF statement evaluation depends on the condition result.

1.2) There is no function in L1 that cannot be translated to L0 because we can create functions using lambda , we don't need to define them

1.3) No, even though L2 language has recursive functions and loops it can still be replaces with lambda functions

1.4) PrimOP – is more efficient than Closure
Closure – can be more flexibility

1.5)

- Map – Parallel , the function application to each element is independent on each other.
- Reduce – Sequential – the operation performed by reduce usually requires combining the result of the previous application with the next element in the list.
- Filter – Parallel , the functions that satisfy the pred condition are independents on each other
- All – Parallel , all functions checks if every element satisfy some Boolean condition so each element result is independent on the others
- Compose- Sequential , because the function is composed by applying the give list of procedures in order and thus the order of application is important to ensure the correct result.

1.6) Lexical address refers to the location of the variable or function in the source code.
Its determined by the lexical scope