

Tarea 2

Grupo 11:

Roberto Omaña 06-39990

Leopoldo Pimentel 06-40095

1.

Para ambos casos, escogemos el `foldr`, ya que en la presencia de evaluación perezosa, `foldr` devolverá inmediatamente la aplicación de la función a la cabeza de la lista y al caso recursivo de plegado sobre el resto de la lista. Por ello, si `f` es capaz de producir parte del resultado sin referencia al caso recursivo de su “derecha” de su *segundo* argumento, y el resto del resultado nunca es solicitado entonces la recursión se detiene. Esto le permite a `foldr` operar sobre listas infinitas y en estos casos particulares, realizar menos cantidad de evaluaciones en comparación con `foldl`.

3.

a)

La función `foldr` recibe una función por cada constructor de un tipo en cuestión. En el caso de `ArbolRosa`, recibe sólo una función que corresponde al único constructor que posee este tipo.

4.

a)

```
unfoldrLista :: (b -> Maybe (a,b)) -> b -> [a]
```

```
unfoldrLista f b = case f b of
```

```
  Nothing    -> []
```

```
  Just (a,bb) -> a:unfoldrLista f bb
```

La función toma el elemento y retorna 'Nothing' si ya terminó de producir la lista o retorna 'Just' (a,bb), en cuyo caso, 'a' se antepone en la lista y 'bb' es usado como el siguiente elemento en una llamada recursiva.