Collection of mapping reductions and pseudocode that prove them.

1. Show a reduction from $H_{TM} \leq_M B$, where H = $\{< M >|$ M halts on $\epsilon\}$

   - Pseudo code

```
def R(<M>):
    def N(x):
        if x == ''00'':
            M("")
            return Accept
        elif x == ''11'':
            return Reject
        else:
            return Reject
    return(N)
```

   - Commentary:
     - If N is in B then N accepts 00 or 11, but not both. By construction N doesn't accept 11, so it must accept 00. The only way it can accept 00 is when $M(\epsilon)$ either accepts or rejects.
     - If N is not in B then it can't accept on the previous conditions: N must accept either both 00 and 11, or neither. By construction N rejects 11, which means it can't also reject 00. Again by construction N can't ever reject 00: it either accepts 00 or it loops.

2. Show a reduction from $H_{TM} \leq_M B$, where H = $\{< M >|$ M halts on $\epsilon\}$

   - Pseudo code

```
def R(<M>):
    def N(x):
        if x == ''00'':
            M("")
            return Accept
        elif x == ''11'':
            return Reject
        else:
            return Reject
    return(N)
```

   - Commentary:
     - If N is in B then N accepts 00 or 11, but not both. By construction N doesn't accept 11, so it must accept 00. The only way it can accept 00 is when $M(\epsilon)$ either accepts or rejects.

  - If N is not in B then it can't accept on the previous conditions: N must accept either both 00 and 11, or neither. By construction N rejects 11, which means it can't also reject 00. Again by construction N can't ever reject 00: it either accepts 00 or it loops.