Question 1

Mapper:

```
#!/usr/bin/env python
import csv
import io
import sys

headers = sys.stdin.readline().strip().split(",")

startdate_index = headers.index("Start Date")

csv.field_size_limit(sys.maxsize)

for row in csv.reader(sys.stdin.readlines()):
    # Détecter les lignes sans startdate ou avec startdate vide
    if len(row) <= startdate_index or row[startdate_index] == "":
        # clé: ligne du csv en entier
        # valeur: 1
        string = io.StringIO()
        csv_output = csv.writer(string)
        csv_output.writerow(row)
        print(repr(string.getvalue()[:-2]), 1, sep="\t")
```

Reducer:

```
#!/usr/bin/env python
from ast import literal_eval
import sys

for line in sys.stdin:
    line = line.strip()
    row, *_ = line.split('\t', 1)
    row = literal_eval(row)

    print(row)
```

Configuration:

Il faut modifier les paramètres de hadoop pour l'empêcher de découper arbitrairement le fichier CSV

`etc/hadoop/mapred-site.xml`:

```
<configuration>
    <property>
        <name>mapred.min.split.size</name>
        <value>9999999999999999999</value>
    </property>
</configuration>
```

Exécution et mesure:

Le shell que j'utilise, fish, fournit la commande `time`.

```
time ./bin/mapred streaming -input ../tp1/input/clinical_trials/ctg-studies.csv -output ../tp1/input/
clinical_trials_avec_valeurs_mq -mapper ../tp1/programmes/q1/mapper.py -reducer ../tp1/programmes/q1/
reducer.py
```

Mesure du temps:

```
Executed in   10.12 secs    fish           external
   usr time   12.24 secs    0.00 micros   12.24 secs
   sys time    1.61 secs  522.00 micros    1.61 secs
```

Question 2

Mapper

```python
#!/usr/bin/env python
import csv
import sys

headers = sys.stdin.readline().strip().split(",")

phases_index = headers.index("Phases")

csv.field_size_limit(sys.maxsize)

for row in csv.reader(sys.stdin.readlines()):
    if len(row) > phases_index:
        phase = row[phases_index]
        # Vérfier si c'est pas une valeur vide
        if phase != "" and phase != "NA":
            print(phase, 1, sep="\t")
```

Reducer

```python
#!/usr/bin/env python
import json
import sys

res: dict[str, int] = {}

for line in sys.stdin:
    line = line.strip()
    key, value = line.split('\t', 1)

    value = int(value)

    if key in res:
        res[key] += value
    else:
        res[key] = value

print(json.dumps(res))
```

Exécution

```
time ./bin/mapred streaming -input ../tp1/input/clinical_trials/ctg-studies.csv -output ../tp1/input/
clinical_trials_phases -mapper ../tp1/programmes/q2/mapper.py -reducer ../tp1/programmes/q2/reducer.py
```

Mesure du temps:

```
Executed in   10.21 secs    fish           external
   usr time   12.71 secs    0.00 micros   12.71 secs
   sys time    1.64 secs  463.00 micros    1.64 secs
```

Résultat:

{"EARLY_PHASE1": 5345, "PHASE1": 43787, "PHASE1|PHASE2": 15072, "PHASE2": 58897, "PHASE2|PHASE3": 6928, "PHASE3": 38874, "PHASE4": 32880}