```
"""
Carbon and kWh calculator. Searches folder for slurm job output files.
Calculates the total core hours used.
You can specify a start date for the seach.
Uses simple model (http://www.archer.ac.uk/about-archer/hardware/,
carbon trust) to convert this to kWh and kg of carbon.
Requires Python 3.5. For start date functionality requires Mac os.
"""
import datetime
import time
import os
import glob
import argparse
import sys

def carbon_calculator(folder, startdate):
    TotalNodeHours = 0
    nodeHours = 0
    for filename in glob.iglob(folder+"/**/*.o[!ut]*",recursive=True):
        if startdate:
            stat = os.stat(filename)
            creationDate = stat.st_birthtime
            if creationDate > startdate:
                nodeHours =  read_file(folder,filename)
        else:
            nodeHours = read_file(folder,filename)
        if nodeHours is not None:
            TotalNodeHours += nodeHours

    if TotalNodeHours==0:
        print ("zero node hours found....aborting...")
        return

    kWh = TotalNodeHours * (1200/4920)
    kgCo2 = kWh*0.5246
    text = """Total kWh for this folder: {0} \n
           Total kg of CO2 for this folder: {1}""".format(kWh, kgCo2)
    print (text)

def read_file(folder,filename):
    for line in open(filename):
        if "Resources allocated:" in line:
            ncpus = line.split("ncpus=")[1].split(",vmem")[0]
            walltime = line.split("walltime=")[1]
            with open(folder+"/FilesFound.txt","a") as textfile:
```

```
                textfile.write(filename+'\n')

            nodes = int(ncpus) / 24
            hours = int(walltime[:2])+int(walltime[3:5])/
                    60+int(walltime[6:8])/3600
            return nodes*hours

carbon_calculator(sys.argv[1], sys.argv[2])
```