```
#!/usr/bin/env python
# author Marilen Corciovei len@len.ro, this code is offered AS IS, use at your
own risk

import re, sys, email, getopt, marshal

msg_start = 'From'
cleaned = None
mids = {}

def parse_mbox(file_name):
    file = open(file_name, 'r')
    msg = ''
    lastLine = ''
    while 1:
        line = file.readline()
        if not line: break
        if line.startswith(msg_start) and lastLine == '':
            if len(msg) > 0:
                parse_msg(msg)
            msg = ''
        msg = msg + line #+ '\n'
        lastLine = line.strip()

def parse_msg(smsg):
    m = email.message_from_string(smsg)
    if 'message-id' in m:
        mid = m['message-id']
        if mid in mids:
            print 'Duplicate Message-ID:', mid
        else:
            print 'New Message-ID:', mid
            mids[mid]=mid
            cleaned.write(smsg)

if __name__=='__main__':
    in_file = ''
    out_file = ''
    hash_file = ''
    try:
        opts, args = getopt.getopt(sys.argv[1:], "i:o:h:")
    except getopt.GetoptError:
        print 'Usage', sys.argv[0], '-i input -o output [-h hash file]'
        sys.exit(2)
    for o, a in opts:
        if o == "-i":
            in_file = a
        if o == "-o":
            out_file = a
        if o == "-h":
            hash_file = a

    if in_file == '' or out_file == '':
        print 'Usage', sys.argv[0], '-i input -o output [-h hash file]'
        sys.exit(2)

    #global cleaned
    cleaned = open(out_file, 'w')
    if hash_file != '':
        try:
            mids = marshal.load(open(hash_file,'r'))
        except:
            pass
```

```
parse_mbox(in_file)
if hash_file != '':
    marshal.dump(mids, open(hash_file,'w'))
```