```python
import os, sys, platform
import psutil, time, smtplib, signal
from email.mime.text import MIMEText



"""
Ejercicio 3. Desarrollar un script que nos muestre por consola el consumo de CPU
y calcule internamente si dicho consumo supera el 70% 4 veces consecutivas,
en un periodo de tiempo preestablecido por el administrador.
En caso de producirse dicho evento, el script deberá enviar un mail al correo
del administrador
"""

"""
Usage: Makes the login and returns the serverSMTP
Name of method: create_session
Date of creation: 25/02/2021
Members: Roberto Jiménez y Alberto Pérez
Last modification: 25/02/2021
Parameters:
    Entry:
        - emisor: Address of the account
        - passwd: The password of the mail
    Out:
        - serverSMTP, for sending the mail
"""

def create_session(emisor, passwd):
    #Connection with SMTP server
    serverSMTP = smtplib.SMTP('smtp.gmail.com',587)
    serverSMTP.starttls()

    #Log-in
    serverSMTP.login(emisor, passwd)
    return serverSMTP


"""
Generation of the mail that will be sended
Name of method: generateMail
Date of creation: 25/02/2021
Members: Roberto Jiménez y Alberto Pérez
Last modification: 25/02/2021
Parameters:
    Entry:
        - mailFrom: Address of the account that the mail will be sended from
        - to: Address of the administrator mail
        - subject: Subject of the mail
        - content: Contet of the mail
    Out:
        - message, mail generated propertly
"""
def generateMail(mailFrom,to,subject, content):
    #Generation of the mail with MIMEText
    message = MIMEText(content, "plain")
    message["From"] = mailFrom
    message["To"] = to
    message["Subject"] = subject
    return message
```

```python


"""
Definition of the main method of the program
Name of method: main
Date of creation: 25/02/2021
Members: Roberto Jiménez y Alberto Pérez
Last modification: 25/02/2021
Parameters:  None, parameters are not needed
"""
def main():
    if  platform.system() != 'Linux':
        #Check if is a UNIX machine
        print("Error, the OS is not a UNIX machine. Getting out...")
        exit(1)
    if len(sys.argv) == 1:

        #Declaration of the correct parameters for the mail
        mailFrom = "roberatecaads2@gmail.com"
        passwd='administracion2'
        mailTo = "rooobertrl@gmail.com"

        content = "Last 60 seconds, CPU has reached 4 times more than 70%"
        subject = "High CPU!"

        #Start an alarm every 60 seconds
        signal.alarm(60)
        reachedTimes = 0

        #Checking when the cpu raises the 70 percent 4 times in a row
        while(True):
            CPUpercent = psutil.cpu_percent(interval=3)
            if CPUpercent >= 70:
                reachedTimes += 1
            else:
                reachedTimes = 0
            if reachedTimes == 4:
                try:
                    smtp = create_session(mailFrom,passwd)
                    email = generateMail(mailFrom, mailTo, subject,content )
                    smtp.sendmail(mailFrom, mailTo, email.as_string())
                    smtp.quit()
                except Exception as e:
                    print("Excepcion: ", e)
                break

    else:
        #Case when parameters are passed to the script
        print("Error, the parameters are not necessary")


if __name__ == "__main__":
    """
    In that case, we check that the program starts in the right function, in our
    case, main()
    """
    main()

```