MAIL SORTING | SELECTION | SWITCHFULLY – Léonie Bouchat

```
// function to sort and send emails with four parameters: The first one is the list of emails from which the emails are deleted, the second one is used to select the email to send ; the third one is used to create the address of destination, the fourth one is used to count the number of emails sent to the selected address of destination.
FUNCTION sendEmail(listOfEmails, email, destination, count)
BEGIN
   send(email, "{destination}@parkshark.com")   // call the functionality to send the email
   delete(email, listOfEmails)    // functionality to delete one email from the list of emails.
   RETURN count + 1    // add one to the number of emails sent to the selected address
END

// subprocedure to print the number of emails sent at the end of the process or at the end of the day
SUBPROCEDURE printTotalOfEmailsSent(total, recruitement, spam, sales, reception)
BEGIN
   WRITE(« We have sorted {total} emails :
   {recruitement} to recruitment, {spam} to spam, {sales} to sales, {reception} to reception »)
END


ALGORITHM sortingEmails

// Declaration of all variables :
// - listOfEmail is an array containing all emails at the beginning of the process
// - recruitement, spam, sales and reception are used to count the number of emails sent during the process
// - totalRecruitement, totalSpam, totalSales, totalReception are used to count the number of emails sent during the day
// totalOfList is the addition of all emails sent during the process
// totalOfDay is the addition of all emails sent during the day
// ended allows to stop the program at the end of the day
VARIABLES
   listOfEmails : array[] of strings
   recruitement, spam, sales, reception, totalofList, totalRecruitement, totalSpam, totalSales, totalReception, totalOfDay : integer
   ended: boolean

BEGIN
   // Attribution of values
   totalRecruitement <- 0
   totalSpam <- 0
   totalSales <- 0
   totalReception <- 0
   totalOfDay ← totalRecruitement + totalSpam + totalSales + totalReception
   ended <- false

   WHILE day is not ended     // Loop for the day
      listOfEmails <- input()    // Getting emails from outside
      recruitement <- 0
```

```
        spam <- 0
        sales <- 0
        reception <- 0
        totalofList <- recruitment + spam + sales + reception
        FOR index FROM 0 TO (length of listOfEmails)-1    // Loop for the process
           IF listOfEmails[index] contains the word "CV" THEN
              recruitement <- sendEmail(listOfEmails, listOfEmails[index], "recruitment", recruitment)
           ELSEIF listOfEmails[index] contains the word "promo" OR email contains the word
          "advertising" THEN
              spam <- sendEmail(listOfEmails, listOfEmails[index], "spam", spam)
           ELSEIF listOfEmails[index] contains the word "proposal" THEN
              sales <- sendEmail(listOfEmails, listOfEmails[index], "sales", sales)
           ELSE
              reception <- sendEmail(listOfEmails listOfEmails[index], "reception", reception)
           ENDIF
        ENDFOR

        // Write message
        PrintTotalOfEmailsSent(totalOfList, recruitement, spam, sales, reception)

        // addition total emails of the day
        totalRecruitement = totalRecruitement + recruitement
        totalSpam = totalSpam + spam
        totalSales = totalSales + sales
        totalReception = totalReception + reception

        ended = input(« Is the end of the day ? »)
     ENDWHILE

// Write message
PrintTotalOfEmailsSent(totalOfDay, totalRecruitement, totalSpam, totalSales, totalReception)

END
```