Exercice 1

bug : an error in the logic of a program ;
debugging : finding and fixing such errors.

Le texte qui suit est extrait de l'excellent livre "Introduction to Computation and Programming Using Python", Spring 2013 Edition, par John V. Guttag (MIT Press Cambridge, Massachusetts), basé sur un cours du Massachussets Institute of Technology et dont environ la moitié correspond au programme de CPGE.

Traduire ce texte.

There is a charming urban legend about how the process of fixing flaws in software came to be known as debugging. The photo below is of a September 9, 1947 page in a laboratory book from the group working on the Mark II Aiken Relay Calculator at Harvard University.
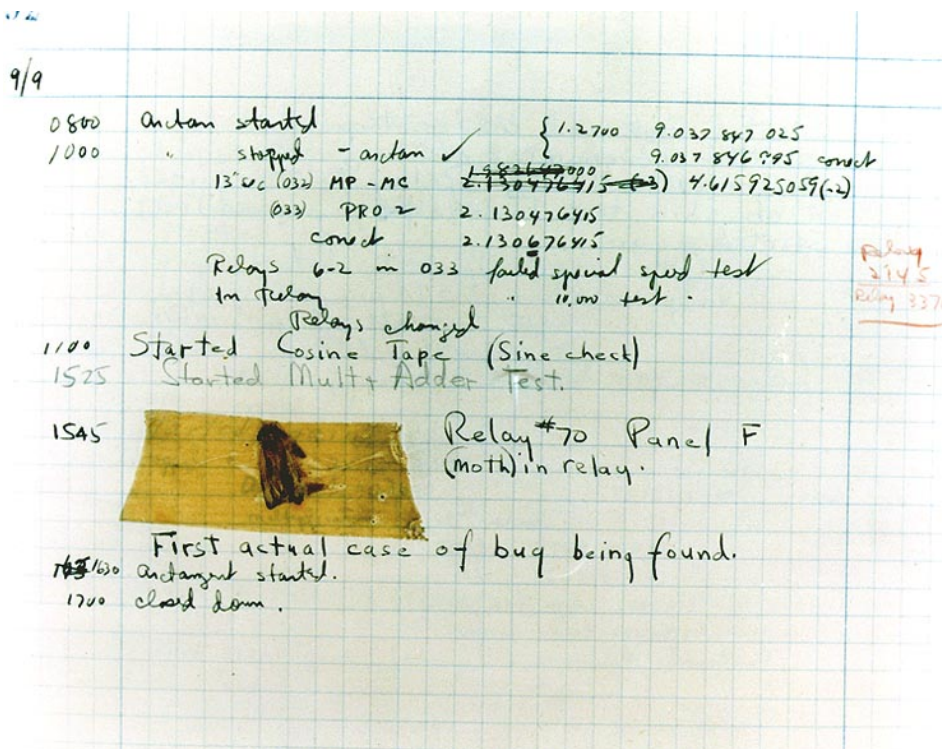

FIGURE 1 – Photo # NH 96566-KN (Color) The First Computer "Bug" (U.S. Navy)

Some have claimed that the discovery of that unfortunate moth trapped in the Mark II led to the use of the phrase debugging. However the wording, "First actual case of bug being found" suggests that a less literal interpretation of the phrase was already common. Grace Murray Hopper, a leader of the Mark II project, made it clear that the term "bug" was already in wide use to describe problems with electronic systems during World War II. And well prior to that, Hawkin's New Catechism of Electricity, an 1896 electrical handbook, included the entry, "The term 'bug' is used to a limited extent to designate any fault or trouble in the connections or working of electric apparatus." In English usage the word "bugbear" means "anything causing seemingly needless or excessive fear or anxiety." (Webster's New World College

Dictionnary). Shakespeare seems to have shortened this to "bug", when he had Hamlet kvetch about "bugs and goblins in my life." (Act 5, scene 2).

The use of the word "bug" sometimes leads people to ignore the fundamental fact that if you wrote a program and it has a "bug", you messed up. Bugs do not crawl unbidden into flawless programs. If your program has a bug, it is because you put it there. Bugs do not breed in programs. If your program has multiple bugs, it is because you made multiple mistakes.

Exercice 2

Cet exercice se traite sans machine, uniquement sur papier. Corriger le programme suivant (environ 6 erreurs) :

```
a=input('Entrer un nombre entier a= ')
b=input('Entrer un nombre entier b= ')
q=a/b
r=a-b*q
if r=0:
    print(a,'est divisible par', b,', le quotient vaut', q)
    else:
    print('Le quotient de', a,'par',b,'est q et le reste est',r)
```

Exercice 3

Commenter les phrases suivantes :

"Cette phrase contient deux erreur."
"Cette phrase contient deux errreur."
"Cette phrase contient une erreur."