TD

Sources :

Savoirs et compétences :

- *AN.C1 : Justifier qu'une itération (ou boucle) produit l'effet attendu au moyen d'un invariant*
- *AN.C2 : Démontrer qu'une boucle se termine effectivement*
- *AN.S1 : Recherche dans une liste, recherche du maximum dans une liste de nombres, calcul de la moyenne et de la variance.*
- *AN.S2 : Recherche par dichotomie.*
- *AN.S4 : Recherche d'un mot dans une chaîne de caractères.*

Proposition de corrigé

0.1 Test de primalité

Q 1 : Proposer un invariant de boucle pour démontrer cet algorithme.

```
def est_premier(n):
    """ Renvoie True si n est premier, False sinon
        Préconditon : n est un entier."""
    for d in range(2,n):
        # n n'est pas divisible par 2, 3, ..., d-1.
        if n % d == 0:
            return False
    return True
```

0.2 Fonction mystère

Q 1 : À chaque tour de boucle, k est incrémenté de 1 et p est multiplié par b.

Au début du premier tour de boucle, on a $k=0$ et $p=1$. Au tour suivant, $k=1$ et $p=b$. Au second tour, $k=2$ et $p=b^2$. Le tableau se dresse aisément.

Q 2 : Montrons que « $b=p^k$ » est un invariant pour la boucle `while`. En entrée de boucle, on a $k=0$ et $p=1=b^0$, donc l'invariant est bien initialisé. Supposons qu'au début d'un tour de boucle on ait $b=p^k$. À la fin de la ligne 5, comme k est incrémenté de 1, $p=b^{k-1}$. À la fin de la ligne 6, comme p est multiplié par b, on a $p=b^k$. L'invariant est donc vrai au début du tour de boucle suivant.

Ainsi, « $b=p^k$ » est un invariant pour la boucle `while`.

Q 3 : Montrons que « $a-p$ » est un variant pour la boucle `while`.

- C'est bien un entier car par construction p est toujours un entier et a est entier par définition.
- C'est un entier positif : d'après la condition de la boucle `while`, p divise a. Comme a est strictement positif et comme p est positif, $p \leq a$, donc $a-p \geq 0$.
- C'est un entier positif qui décroît strictement à chaque tour de boucle : à chaque tour de boucle, p est multiplié par b et $b>1$.

Q 4 : On a écrit un variant pour la boucle `while`, donc la fonction renvoie bien un résultat. Par l'invariant, en sortie de boucle, on a $p=b^k$. En sortie de boucle, k est le plus petit entier pour lequel b^k ne divise pas a. On renvoie en sortie $k-1$. Ainsi, la fonction renvoie le plus grand entier k tel que b^k divise a.