

23 NSI 08

```
In [ ]: class Pile:
        """
        Classe definissant une structure de pile.
        """
        def __init__(self):
            self.contenu = []

        def est_vide(self):
            """
            Renvoie le booleen True si la pile est vide, False sinon.
            """
            return self.contenu == []

        def empiler(self, v):
            """
            Place l'element v au sommet de la pile
            """
            self.contenu.append(v)

        def depiler(self):
            """
            Retire et renvoie l'element place au sommet de la pile,
            si la pile n'est pas vide.
            """
            if not self.est_vide():
                return self.contenu.pop()
```

```
In [ ]: def eval_expression(tab):
        p = Pile()
        for ... in tab:
            if element != '+' ... element != '*':
                p.empiler(...)
            else:
                if element == '+':
                    resultat = p.depiler() + ...
                else:
                    resultat = ...
                p.empiler(...)
        return ...
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js