```
import autograd as ag
import autograd.numpy as np
```

1 Analyse des points stationnaires

L'objectif suivant admet cinq points stationnaires.

$$f(a,b) \quad = \quad a^3 + 2\,a^2 - 2\,a\,b + b^2 + a\,b^3 - 2\,b + 5\,.$$

Question 1. Estimer graphiquement les points stationnaires en adaptant le fichier `graphe.py`.

Question 2. Écrire une fonction `nabla_f` et une fonction `H_f` paramétrées par a et b et qui retournent le gradient et la matrice hessienne de l'objectif.

Question 3. Utiliser les réponses aux deux questions précédentes pour calculer précisément les les coordonnées des points stationnaires de f.

Question 4. En analysant les propriétés de la hessienne, déterminer quels points stationnaires sont des minima locaux de f.

Question 5. Reprendre les questions précédentes en utilisant les fonctions `grad` et `hessian` du paquetage `autograd`. Note : pour utiliser `autograd`, il faut transformer f en une fonction `fbis` paramétrée par un tableau `numpy` (la version `autograd` de `numpy`) contenant a et b :

```
def fbis(u) :
    a = u[0]
    b = u[1]
```

Question 6. Reprendre les questions précédentes en écrivant une fonction `nabla_fter` qui évalue le gradient de f par une méthode de dérivation automatique appliquée au code suivant, qui évalue f (voir la section suivante pour la méthode).

```
def nabla_fter(u) :
    a = u[0]
    b = u[1]
    t1 = b ** 2
    t2 = a ** 2
    t3 = a * t1 * b - 2 * a * b + t2 * a - 2 * b + t1 + 2 * t2 + 5
```

2 Introduction à la dérivation automatique (*backward mode*)

On cherche à calculer le gradient ∇f de la fonction suivante par une méthode algorithmique :

$$f(a,b) \quad = \quad (e^a)^2 + a\,b\;e^a + \sin(e^a)\,.$$

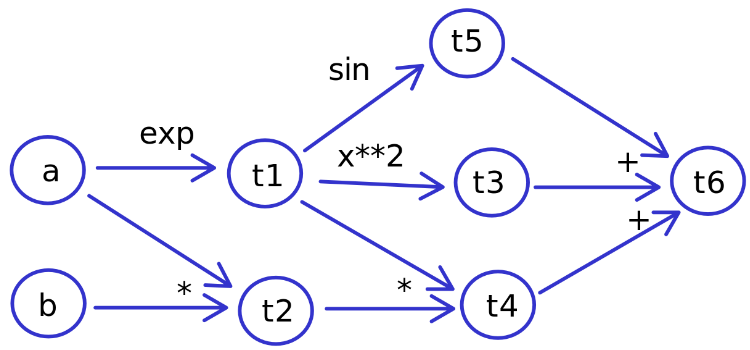
Le résultat attendu est

$$\nabla f \quad = \quad \begin{pmatrix} \dfrac{\partial f}{\partial a} \\ \dfrac{\partial f}{\partial b} \end{pmatrix} \quad = \quad \begin{pmatrix} 2\,(e^a)^2 + (a\,b + \cos\,(e^a) + b)\;e^a \\ a\,e^a \end{pmatrix}.$$

Le code suivant évalue f :

```
t1 = exp(a)
t2 = a*b
t3 = t1**2
t4 = t1*t2
t5 = sin(t1)
t6 = t3 + t4 + t5
```

On peut lui associer le graphe suivant :



Considérons un nœud (disons) t_1. Prenons tous les nœuds qui l'utilisent explicitement (les successeurs de t_1 dans le graphe) : t_3, t_4 et t_5. À partir des instructions, il est facile de calculer $\frac{\partial t_k}{\partial t_1}$ pour $k = 3, 4, 5$:

$$
\begin{aligned}
t_3 &= t_1^2 & \frac{\partial t_3}{\partial t_1} &= 2\,t_1\,, \\
t_4 &= t_1\,t_2 & \frac{\partial t_4}{\partial t_1} &= t_2\,, \\
t_5 &= \sin(t_1) & \frac{\partial t_5}{\partial t_1} &= \cos(t_1)\,.
\end{aligned}
$$

On peut donc évaluer partiellement la formule suivante (pour une justification, voir la section 2.1) :

$$
\begin{aligned}
\frac{\partial f}{\partial t_1} &= \frac{\partial f}{\partial t_3}\frac{\partial t_3}{\partial t_1} + \frac{\partial f}{\partial t_4}\frac{\partial t_4}{\partial t_1} + \frac{\partial f}{\partial t_5}\frac{\partial t_5}{\partial t_1}\,, && (1)\\
&= 2\,t_1\,\frac{\partial f}{\partial t_3} + t_2\,\frac{\partial f}{\partial t_4} + \cos(t_1)\,\frac{\partial f}{\partial t_5}. && (2)
\end{aligned}
$$

Et on voit que, si les valeurs des $\frac{\partial f}{\partial t_k}$ sont stockées dans des variables `df_dtk` pour $k = 3, 4, 5$ et t_2 est évaluée alors il est facile d'obtenir $\frac{\partial f}{\partial t_1}$. La formule (1) peut se formuler de façon générale ainsi :

$$
\frac{\partial f}{\partial t} = \sum_{s \text{ successeur de } t} \frac{\partial f}{\partial s}\frac{\partial s}{\partial t}. \qquad (3)
$$

Le principe de la dérivation automatique (*backward mode*) consiste à évaluer la fonction f complètement (on est alors sûr que toutes les variables t_i sont évaluées) puis calculer les valeurs des $\frac{\partial f}{\partial t_k}$ en partant de la fin du graphe (pour $k = 6, 5, \ldots, 1$, donc) et en finissant par $\frac{\partial f}{\partial a}$ et $\frac{\partial f}{\partial b}$ (les valeurs recherchées). Plus généralement, il faut énumérer les nœuds du graphe suivant un ordre topologique. Sur l'exemple, on s'assure ainsi que les valeurs des $\frac{\partial f}{\partial t_k}$ pour $k = 3, 4, 5$ sont évaluées avant d'évaluer $\frac{\partial f}{\partial t_1}$. On obtient :

```
t1 = exp(a)
t2 = a*b
```

```
t3 = t1**2
t4 = t1*t2
t5 = sin(t1)
t6 = t3 + t4 + t5
df_dt6 = 1
df_dt5 = df_dt6
df_dt4 = df_dt6
df_dt3 = df_dt6
df_dt2 = t1*df_dt4
df_dt1 = 2*t1*df_dt3 + t2*df_dt4 + cos(t1)*df_dt5
df_da = exp(a)*df_dt1 + b*df_dt2
df_db = a*df_dt2
```

Remarque. Les deux modes de dérivation automatique (*forward* et *backward*) calculent les mêmes objets mais pas au même coût. Sur cet exemple, le mode *backward* utilise huit variables. Sur le même exemple, le mode *forward* utiliserait 2 variables fois 6 nœuds, c'est-à-dire douze variables. Quand on doit calculer le gradient ou la jacobienne d'une fonction $\mathbb{R}^m \to \mathbb{R}^p$, le mode *backward* est plus efficace si $m > p$ (c'est le cas ici) alors que le mode *forward* est plus efficace si $m < p$.

Remarque. On a présenté les deux modes de dérivation automatique en insérant du code dans un code existant. Les fonctions du paquetage `autograd` ne fonctionnent pas ainsi. Elles redéfinissent le type des éléments des tableaux `numpy` et surchargent les opérateurs et les fonctions arithmétiques. Une grande partie du travail est donc effectuée dans une structure de données spécialisée. Toutefois, les enchaînements de calculs sont essentiellement les mêmes que ceux que nous avons présentés.

2.1 Justification de la formule (1)

Il suffit de savoir que si f est une fonction de trois variables a, b et c alors la *différentielle* de f s'écrit :

$$\mathrm{d}f = \frac{\partial f}{\partial a}\,\mathrm{d}a + \frac{\partial f}{\partial b}\,\mathrm{d}b + \frac{\partial f}{\partial c}\,\mathrm{d}c\,.$$

La formule se généralise à n variables, bien sûr.

Revenons à la fonction f de l'exemple. On peut la voir comme une fonction $f(t_3, t_4, t_5)$. On a donc

$$\mathrm{d}f = \frac{\partial f}{\partial t_3}\,\mathrm{d}t_3 + \frac{\partial f}{\partial t_4}\,\mathrm{d}t_4 + \frac{\partial f}{\partial t_5}\,\mathrm{d}t_5\,. \tag{4}$$

Les variables t_3 et t_5 sont des fonctions de t_1. La variable t_4 est une fonction de t_1 et de t_2. On a donc

$$\begin{aligned}
\mathrm{d}t_3 &= \frac{\partial t_3}{\partial t_1}\,\mathrm{d}t_1\,,\\
\mathrm{d}t_4 &= \frac{\partial t_4}{\partial t_1}\,\mathrm{d}t_1 + \frac{\partial t_4}{\partial t_2}\,\mathrm{d}t_2\,,\\
\mathrm{d}t_5 &= \frac{\partial t_5}{\partial t_1}\,\mathrm{d}t_1\,,
\end{aligned}$$

En remplaçant les variables $\mathrm{d}t_k$ par leur valeur dans (4) (pour $k = 1, 2, 3$) on obtient

$$\begin{aligned}
\mathrm{d}f &= \frac{\partial f}{\partial t_3}\frac{\partial t_3}{\partial t_1}\,\mathrm{d}t_1 + \frac{\partial f}{\partial t_4}\left(\frac{\partial t_4}{\partial t_1}\,\mathrm{d}t_1 + \frac{\partial t_4}{\partial t_2}\,\mathrm{d}t_2\right) + \frac{\partial f}{\partial t_5}\frac{\partial t_5}{\partial t_1}\,\mathrm{d}t_1\,, &(5)\\
&= \left(\frac{\partial f}{\partial t_3}\frac{\partial t_3}{\partial t_1} + \frac{\partial f}{\partial t_4}\frac{\partial t_4}{\partial t_1} + \frac{\partial f}{\partial t_5}\frac{\partial t_5}{\partial t_1}\right)\mathrm{d}t_1 + \frac{\partial f}{\partial t_4}\frac{\partial t_4}{\partial t_2}\,\mathrm{d}t_2 &(6)
\end{aligned}$$

Mais la fonction f est aussi une fonction $f(t_1, t_2)$ et donc

$$\mathrm{d}f = \frac{\partial f}{\partial t_1}\,\mathrm{d}t_1 + \frac{\partial f}{\partial t_2}\,\mathrm{d}t_2\,. \tag{7}$$

En identifiant les coefficients de $\mathrm{d}t_1$ et $\mathrm{d}t_2$ dans (6) et (7) on trouve (la première ligne correspond à la formule (1)) :

$$\begin{aligned}
\frac{\partial f}{\partial t_1} &= \frac{\partial f}{\partial t_3}\frac{\partial t_3}{\partial t_1} + \frac{\partial f}{\partial t_4}\frac{\partial t_4}{\partial t_1} + \frac{\partial f}{\partial t_5}\frac{\partial t_5}{\partial t_1}\,, \\
\frac{\partial f}{\partial t_2} &= \frac{\partial f}{\partial t_4}\frac{\partial t_4}{\partial t_2}
\end{aligned}$$