3 questions

6 points par question

2 points pour les devoirs

C’est toujours une mauvaise idée de ne pas faire une question !

L’interrogation n’a qu’un impact positif
Note = max(Examen , (2*Examen+Interro)/3)

Le dernier devoir !
So easy !

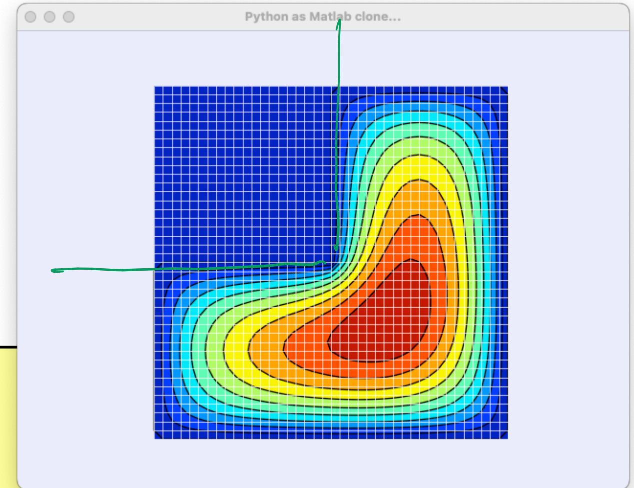
```
def poissonSolve(nCut):
  n = 2*nCut + 1; m = n*n; h = 2/(n-1)

  A = eye(m); B = zeros(m)
  for i in range(1,n-1):
    for j in range(1,n-1):
      index = i + j*n
      A[index,index] = 4
      A[index,index-1] = -1
      A[index,index+1] = -1
      A[index,index+n] = -1
      A[index,index-n] = -1
      B[index] = 1

  return solve(A/(h*h),B).reshape(n,n)
```

Le dernier devoir !

So easy !




```
def poissonSolve(nCut):
  n = 2*nCut + 1; m = n*n; h = 2/(n-1)

  A = eye(m); B = zeros(m)
  for i in range(1,n-1):
    for j in range(1,n-1):
      if (i > nCut or j < nCut):
          index = i + j*n
          A[index,index] = 4
          A[index,index-1] = -1
          A[index,index+1] = -1
          A[index,index+n] = -1
          A[index,index-n] = -1
          B[index] = 1
  return solve(A/(h*h),B).reshape(n,n)
```

Le méchant ?

1 La méthode de Francesco à deux pas !

Pour résoudre un problème de Cauchy pour $u'(x) = f(u,x)$, Francesco écrit la méthode suivante :

$\mathcal{O}(1)$
$\mathcal{O}(h)$
$\mathcal{O}(h^2)$
$\mathcal{O}(h^3)$

$$U_{n+1} = aU_n + bU_{n-1} + cF_n + dF_{n-1}$$

où $U_n \approx u(X_n)$ et $F_n = f(X_n, U_n)$ avec $X_n = X_0 + nh$.
Il y a 4 paramètres magiques a, b, c et d qu'il faut choisir judicieusement en fonction du pas h.

$\boxed{\mathcal{O}(h^4)}$

1. Quel est le meilleur ordre global de précision qu'on pourrait obtenir ?
   Justifier brièvement votre réponse !

2. Calculer les valeurs de a, b, c et d afin que la méthode soit la plus précise possible.

Exactement un simple verso !
Expliquer et justifier son résultat !
Il faut obtenir les valeurs correctes !

La solution...

Le correcteur est en général très tolérant et honteusement généreux !

Une copie propre est toujours une bonne idée : vous aviez largement le temps pour la faire !

La question était courte et ne nécessitait pas beaucoup de calcul !

LEPL1104	Nom :	Numéro magique
Juin 2023	Prénom :	
Méth. Num.	Bloc annuel :	

1 La méthode de Francesco à deux pas !

Pour résoudre un problème de Cauchy pour $u'(x) = f(u,x)$, Francesco écrit la méthode suivante :

$$U_{n+1} = aU_n + bU_{n-1} + cF_n + dF_{n-1}$$

où $U_n \approx u(X_n)$ et $F_n = f(X_n, \hat{U}_n)$ avec $X_n = X_0 + nh$.
Il y a 4 paramètres magiques a, b, c et d qu'il faut choisir judicieusement en fonction du pas h.

1. Quel est le meilleur ordre global de précision qu'on pourrait obtenir ?
   Justifier brièvement votre réponse !
2. Calculer les valeurs de a, b, c et d afin que la méthode soit la plus précise possible.

$\boxed{1}$ 4 PARAMETRES $\rightarrow$ EGALER TERMES $\mathcal{O}(1)\ \mathcal{O}(h)\ \mathcal{O}(h^2)\ \mathcal{O}(h^3)$

ERREUR LOCALE PROBABLE $\mathcal{O}(h^4)$ $\boxed{\text{ORDRE GLOBAL} = 3}$

$\boxed{3}$

$\boxed{2}$ $U_{n+1} = U_n + hU'_n + \frac{h^2}{2}U''_n + \frac{h^3}{6}U'''_n \ldots$

$$\begin{aligned} U_{n+1} = \ & a\,U_n \\ & + b\left[U_n - hU'_n + \frac{h^2}{2}U''_n - \frac{h^3}{6}U'''_n \ldots\right] \\ & + c\,U'_n \\ & + d\left[U'_n - hU''_n + \frac{h^2}{2}U'''_n \ldots\right] \end{aligned}$$

$\boxed{3}$

$$\boxed{\begin{aligned} U_{n+1} = & -4U_n + 5U_{n-1} \\ & + h(4F_n + 2F_{n-1}) \end{aligned}}$$

$$\left\{\begin{aligned} 1 &= a + b \\ h &= -bh + c + d \\ \frac{h^2}{2} &= b\frac{h^2}{2} - dh \\ \frac{h^3}{6} &= -\frac{bh^3}{6} + \frac{dh^2}{2} \end{aligned}\right.$$

$\boxed{a = 1 - b}$

$\boxed{c/h = 1 + b - d/h}$

$$\boxed{\begin{aligned} 1 &= b - 2d/h \\ 1 &= -b + 3d/h \end{aligned}}$$

$\boxed{2 = d/h}$ $\boxed{d = 2h}$

$\boxed{c = 4h}$

$\boxed{a = -4}$

$\boxed{b = 1 + 4 = 5}$

$\boxed{4}$

Vous pouvez rédiger cet examen avec un crayon (mais bien taillé !)

Question 1.1
Easy game !

1. Quel est le meilleur ordre global de précision qu'on pourrait obtenir ?
   Justifier brièvement votre réponse !

   Comme il y a quatre paramètres, on devrait pouvoir éliminer les termes d'erreur en $\mathcal{O}(1)$, $\mathcal{O}(h)$, $\mathcal{O}(h^2)$ et $\mathcal{O}(h^3)$. La méthode n'est pas symétrique et il est peu probable d'avoir des miracles, numériques, on peut donc déduire que l'erreur locale devrait être en $\mathcal{O}(h^4)$.

   On peut donc espérer un ordre global de précision de trois : $\boxed{E^h \quad = \quad \mathcal{O}(h^3)}$

   La question est très simple !
   Mais il faut être bien attentif à fournir l'ordre global de précision et pas l'ordre local !
   Préciser l'ordre local et global est sans doute une manière efficace d'éviter toute ambiguité !
   Eviter aussi de donner une réponse à choix multiples : cela énerve bêtement le correcteur !
   Quelques étudiants perdent ici bêtement un point facile à gagner pourtant !

Il suffit juste de donner l'ordre !
Pas de réponse à choix multiple !

Question 1.2
Taylor, Taylor !

2. Calculer les valeurs de a, b, c et d afin que la méthode soit la plus précise possible.

Il suffit d'écrire des développements en série de Taylor pour les deux côtés de l'égalité de la méthode de Francesco :

$$\begin{aligned}
U_{n+1} &= U_n + hU'_n + \frac{h^2}{2}U''_n + \frac{h^3}{6}U'''_n + \ldots \\
U_{n+1} &= a\,U_n \\
&+ b\left[U_n - hU'_n + \frac{h^2}{2}U''_n - \frac{h^3}{6}U'''_n + \ldots\right] \\
&+ c\,U'_n \\
&+ d\left[U'_n - hU''_n + \frac{h^2}{2}U'''_n + \ldots\right]
\end{aligned}$$

Identifier les termes…

$$
\begin{aligned}
U_{n+1} &= U_n + hU'_n + \frac{h^2}{2}U''_n + \frac{h^3}{6}U'''_n + \ldots \\
U_{n+1} &= a\,U_n \\
&+ b\left[U_n - hU'_n + \frac{h^2}{2}U''_n - \frac{h^3}{6}U'''_n + \ldots\right] \\
&+ c\,U'_n \\
&+ d\left[U'_n - hU''_n + \frac{h^2}{2}U'''_n + \ldots\right]
\end{aligned}
$$

$$
\boxed{
\begin{aligned}
1 &= a + b \\
h &= -bh + c + d \\
\frac{h^2}{2} &= \frac{bh^2}{2} - dh \\
\frac{h^3}{6} &= -\frac{bh^3}{6} + \frac{dh^2}{2}
\end{aligned}
}
$$

Le système n'est pas bien compliqué à résoudre :

$$\left\{\begin{array}{rcl} 1 & = & a+b \\ 1 & = & -b+c/h+d/h \\ 1 & = & b-2d/h \\ 1 & = & -b+3d/h \end{array}\right.$$

En additionnant les deux dernières lignes, on obtient $d/h = 2$.
Il est ensuite aisé d'obtenir $b = 5$, *puis* $c/h = 4$ *et* $a = -4$.

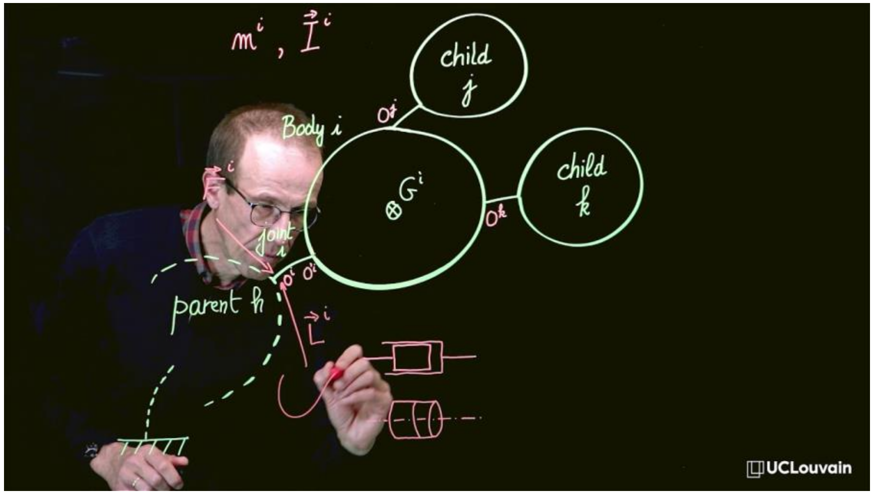
Et on peut donc finalement conclure : $\boxed{U_{n+1} \quad = \quad -4U_n + 5U_{n-1} + h(4F_n + 2F_{n-1})}$

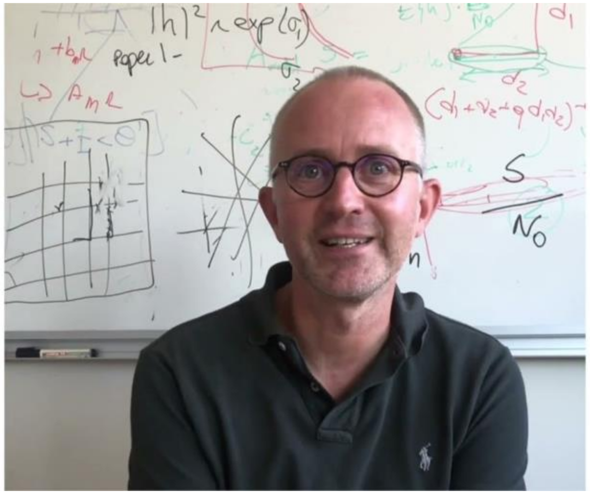
Cette question est une simple application d'une technique classique à appliquer avec rigueur !
Il faut juste être attentif dans les calculs :-)

Il faut vraiment trouver les 4 valeurs !

Physique 2
Intégration composite

$$I \;=\; \int_{-1}^{1} u(x)\; dx \;\approx\; \alpha u(-1) + \beta u(-\zeta) + \beta u(\zeta) + \alpha u(1)$$

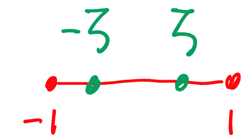




2 Une quadrature composite pour Claude et Paul !

Il s'agit d'intégrer une fonction sur un intervalle $[-1, 1]$. La quadrature de Oestges-Fisette à quatre points consiste à remplacer l'intégration exacte I par une somme pondérée de la fonction en quatre points.

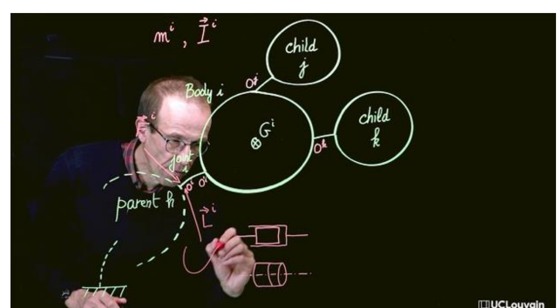
$$I \;=\; \int_{-1}^{1} u(x)\;dx \;\approx\; \alpha u(-1) + \beta u(-\zeta) + \beta u(\zeta) + \alpha u(1)$$

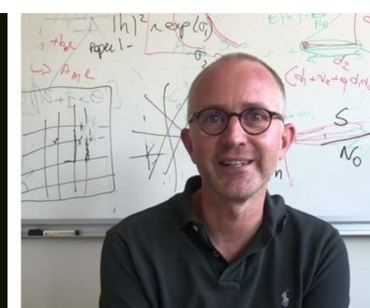


Deux des points sont les extrémités de l'intervalle : c'est ce qui caractérise Oestges-Fisette par rapport à Gauss-Legendre. Les coefficients α, β et ζ sont choisis afin que le degré de précision de la quadrature soit le plus élevé possible. La quadrature composite de Oestges-Fisette consiste ensuite à diviser l'intervalle $[-1, 1]$ en n sous-intervalles de longueur h. Sur chaque sous-intervalle, on utilise simplement la quadrature de Oestges-Fisette en y adaptant judicieusement les poids et points d'intégrations

1. Calculer[1] α, β et ζ afin de permettre l'intégration exacte de tous les polynômes jusqu'au degré 4.
2. Quel est l'ordre de précision de la quadrature composite de Claude et Paul ?
3. On a effectué le calcul de l'intégrale en utilisant $2n$ et n sous-intervalles et on a obtenu respectivement I_h et I_{2h}. Donner la combinaison linéaire de I_h et I_{2h} qui sera théoriquement la meilleure extrapolation de Richardson pour estimer I.

Plus hard !





Exactement un simple verso !
Expliquer et justifier son résultat !
Il faut obtenir les valeurs correctes !

La solution...

LEPL1104	Nom :	Numéro magique
Juin 2023	Prénom :	
Méth. Num.	Bloc annuel :	

2 Une quadrature composite pour Claude et Paul !

Il s'agit d'intégrer une fonction sur un intervalle $[-1,1]$. La quadrature de Oestges-Fisette à quatre points consiste à remplacer l'intégration exacte I par une somme pondérée de la fonction en quatre points.

$$I = \int_{-1}^{1} u(x)\,dx \;\approx\; \alpha u(-1) + \beta u(-\zeta) + \beta u(\zeta) + \alpha u(1)$$

Deux des points sont les extrémités de l'intervalle : c'est ce qui caractérise Oestges-Fisette par rapport à Gauss-Legendre. Les coefficients α, β et ζ sont choisis afin que le degré de précision de la quadrature soit le plus élevé possible. La quadrature composite de Oestges-Fisette consiste ensuite à diviser l'intervalle $[-1,1]$ en n sous-intervalles de longueur h. Sur chaque sous-intervalle, on utilise simplement la quadrature de Oestges-Fisette en y adaptant judicieusement les poids et points d'intégrations

1. Calculer[1] α, β et ζ afin de permettre l'intégration exacte de tous les polynômes jusqu'au degré 4.
2. Quel est l'ordre de précision de la quadrature composite de Claude et Paul ?
3. On a effectué le calcul de l'intégrale en utilisant $2n$ et n sous-intervalles et on a obtenu respectivement I_h et I_{2h}. Donner la combinaison linéaire de I_h et I_{2h} qui sera théoriquement la meilleure extrapolation de Richardson pour estimer I.

$\boxed{1}$

$u(x) = 1 \longrightarrow 2 = 2(\alpha + \beta)$

$u(x) = x \longrightarrow 0 = 0$:-)

$u(x) = x^2 \longrightarrow \frac{2}{3} = 2\alpha + 2\beta\zeta^2$

$u(x) = x^3$:-)

$u(x) = x^4 \qquad \frac{2}{5} = 2\alpha + 2\beta\zeta^4$

$$\boxed{\begin{aligned} 1 &= \alpha + \beta \\ 1 &= 3\alpha + 3\beta\zeta^2 \\ 1 &= 5\alpha + 5\beta\zeta^4 \end{aligned}} \qquad \boxed{3}$$

$$\begin{cases} 1 = 3 - 3\beta + 3\beta\zeta^2 \\ 1 = 5 - 5\beta + 5\beta\zeta^4 \end{cases} \qquad \boxed{2}$$

$$\begin{cases} 2 = 3\beta(1-\zeta^2) \\ 4 = \underbrace{5\beta(1-\zeta^2)}_{2/3}(1+\zeta^2) \end{cases} \qquad \boxed{2}$$

$6 = 5(1+\zeta^2)$

$6/5 - 1 = \zeta^2 \qquad \boxed{\zeta = \frac{1}{\sqrt{5}}} \qquad \boxed{3}$

$2 = 3\beta\left(1 - \frac{1}{5}\right)$

$2 = \frac{12}{5}\beta \longrightarrow \boxed{\beta = \frac{5}{6}}$

$\boxed{\alpha = \frac{1}{6}}$

$\boxed{2}$ $\boxed{m = 6}$ $\qquad d = 5$:-)

$\boxed{3}$ $\boxed{\dfrac{2^6 I_h - I_{2h}}{2^6 - 1} = \dfrac{64 I_h - I_{2h}}{63}}$

$2^2 = 4 \qquad 2^5 = 32$

$2^3 = 8 \qquad 2^6 = 64$:-)

$2^4 = 16$

[1] Obtenir α, β et ζ peut paraître un peu calculatoire, si on omet d'observer que $(1-\zeta^4) = (1-\zeta^2)(1+\zeta^2)$.

Vous pouvez rédiger cet examen avec un crayon (mais bien taillé !)

Question 2.1
Juste un peu d'algèbre

1. Calculer[1] α, β et ζ afin de permettre l'intégration exacte de tous les polynômes jusqu'au degré 4.

   Il faut que l'intégration exacte d'un polynôme quelconque $a + bx + cx^2 + dx^3 + ex^4$ ou l'application de la quadrature de Oestges-Fisette soient équivalents.

$$
\begin{aligned}
\int_{-1}^{1} a + bx + cx^2 + dx^3 + ex^4 \; dx \quad &= \quad \alpha\Big(a - b + c - d + e\Big) + \beta\Big(a - b\zeta + c\zeta^2 - d\zeta^3 + e\zeta^4\Big) \\
&\quad + \beta\Big(a + b\zeta + c\zeta^2 + d\zeta^3 + e\zeta^4\Big) + \alpha\Big(a + b + c + d + e\Big) \; ,
\end{aligned}
$$

$$
2a + \frac{2c}{3} + \frac{2e}{5} \quad = \quad 2a\Big(\alpha + \beta\Big) + 2c\Big(\alpha + \beta\zeta^2\Big) + 2e\Big(\alpha + \beta\zeta^4\Big)
$$

En exigeant que cela soit vrai pour toutes valeurs de a, c et e,

$$
\begin{aligned}
1 &= \alpha + \beta \\
1 &= 3\alpha + 3\beta\zeta^2 \\
1 &= 5\alpha + 5\beta\zeta^4
\end{aligned}
$$

Oui : il faut trouver les valeurs !

En incluant $\alpha + \beta = 1$ dans les deux autres équations, on déduit que :

$$\begin{aligned} 2 &= 3\beta(1-\zeta^2) \\ 4 &= 5\beta(1-\zeta^4) \end{aligned}$$

En observant que $(1-\zeta^4) = (1-\zeta^2)(1+\zeta^2)$,

$$\begin{aligned} 2 &= 3\beta(1-\zeta^2) \\ 4 &= 5\beta(1-\zeta^2)(1+\zeta^2) \end{aligned}$$

En substituant ensuite la première équation dans la seconde :-),

$$6 = 5(1+\zeta^2)$$

On conclut finalement que : $\boxed{\zeta = \frac{1}{\sqrt{5}} \quad \alpha = \frac{1}{6} \quad \beta = \frac{5}{6}}$

Question 2.2
Facile mais pas stupide !

Ici, il n'est pas inutile d'observer que Gauss-Lobatto intégre aussi parfaitement n'importe quel polynôme de degré cinq, puisque tous les termes impairs sont toujours parfaitement intégrés comme on vient de l'observer pour x et x^3 dans le calcul effectué ! Le degré de précision est donc cinq (et non quatre. Eh oui, la question suivante est facile, mais pas totalement stupide !).

2. Quel est l'ordre de précision de la quadrature composite de Claude et Paul ?

   Une quadrature numérique dont le terme d'erreur $I - I_h$ est $\mathcal{O}(h^m)$ est dite d'ordre m.

Pour la quadrature composite de Claude et Paul : $\boxed{m \quad = \quad 6}$

On doit intégrer sur chaque intervalle, une erreur $u - u^h$ qui est un polynôme de degré six: cela génère une erreur locale sur chaque intervalle en $\mathcal{O}(h^7)$ et une erreur globale $\mathcal{O}(h^6)$ en additionnant les $n = 2/h$ erreurs locales ! Globalement, on a toujours $m = d + 1$.

Question 2.3
Richarson !

Cela devait être une question vraiment facile et pourtant les résultats sont assez mitigés !

3. On a effectué le calcul de l'intégrale en utilisant $2n$ et n sous-intervalles et on a obtenu respectivement I_h et I_{2h}. Donner la combinaison linéaire de I_h et I_{2h} qui sera théoriquement la meilleure extrapolation de Richardson pour estimer I.

Comme il s'agit d'éliminer un terme d'ordre 6 : $\boxed{\dfrac{\left(64 I_h - I_{2h}\right)}{63}}$

L'erreur de cette extrapolation sera en $\mathcal{O}(h^8)$, puisqu'il n'y a pas de termes d'erreur de degré impair.

Nombre d'or ?

3 Une convergence un peu exceptionnelle !

Considérons la méthode de la sécante pour la recherche d'une racine x simple : nous avons donc $f'(x) \neq 0$.

$$\boxed{x_{i+1} \quad = \quad x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}}$$

L'erreur commise à chaque itération i est définie par $e_i = x_i - x$.

1. Ecrire une fonction **python** : `x = secante(x0,x1,tol,nmax,f)`
   qui applique la méthode de la sécante pour trouver une racine d'une fonction `f`.
   On exige une précision inférieure à une tolérance `tol` et un nombre maximal d'itérations `nmax`.
   Deux premières valeurs sont données dans `x0` et `x1` pour démarrer les itérations.

2. Démontrer qu'il existe une constante D en termes de $f'(x)$ et $f''(x)$ telle que

   $$e_{i+1} = D\ e_i e_{i-1}$$

   lorsque la méthode de la sécante converge et que $i \to \infty$.

3. Finalement, en déduire[2] le taux de convergence α de la méthode de la sécante.

Exactement un simple verso !
Expliquer et justifier son résultat !
Il faut obtenir les valeurs correctes !

La solution...

Ne pas écrire le programme
est impardonnable !

Une seule évaluation de f par itération
est le point clé !

LEPL1104	Nom :	Numéro magique
Juin 2023	Prénom :	
Méth. Num.	Bloc annuel :	

3 Une convergence un peu exceptionnelle !

Considérons la méthode de la sécante pour la recherche d'une racine x simple : nous avons donc $f'(x) \neq 0$.

$$\boxed{x_{i+1} \;=\; x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}}$$

L'erreur commise à chaque itération i est définie par $e_i = x_i - x$.

1. Ecrire une fonction python : `x = secante(x0,x1,tol,nmax,f)`
qui applique la méthode de la sécante pour trouver une racine d'une fonction `f`.
On exige une précision inférieure à une tolérance `tol` et un nombre maximal d'itérations `nmax`.
Deux premières valeurs sont données dans `x0` et `x1` pour démarrer les itérations.
2. Démontrer qu'il existe une constante D en termes de $f'(x)$ et $f''(x)$ telle que
$$e_{i+1} = D\, e_i e_{i-1}$$
lorsque la méthode de la sécante converge et que $i \to \infty$.
3. Finalement, en déduire[2] le taux de convergence α de la méthode de la sécante.

$\boxed{1}$

```
def secante(x0, x1 tol nmax, f):
  n = 0 ,  delta = float('inf')
  f0 = f(x0)
  while abs(delta)>0 and n < nmax
    n = n+1
    f1 = f(x1)
    delta = -f1 * (x1-x0)/(f1-f0)
    x0 = x1 ; f0 = f1
    x1 = x1 + delta
  return x1
```

$\boxed{3}$

$\boxed{2}$ $e_{i+1} = e_i - \dfrac{f(x+e_i)(e_i - e_{i-1})}{f(x+e_i) - f(x+e_{i-1})}$

$= \dfrac{e_i \cancel{f(x+e_i)} - e_i f(x+e_{i-1}) - f(\cancel{x+e_i})e_i + f(x+e_i)e_{i-1}}{\cdots}$

$= \dfrac{e_{i-1} f(x+e_i) - e_i f(x-e_{i-1})}{f(x+e_i) - f(x+e_{i-1})}$

$= \dfrac{e_{i-1}\left(e_i f' + \frac{e_i^2}{2} f''\right) - e_i\left(e_{i-1} f' + \frac{e_{i-1}^2}{2} f''\right)}{(e_i - e_{i-1}) f'}$

$= e_{i-1} e_i \dfrac{(e_i - \cancel{e_{i-1}})}{(e_i - \cancel{e_{i-1}})} \boxed{\dfrac{f''}{2f'}} = D$

$\boxed{4}$

$\boxed{3}$ $e_{i+1} = D e_i e_{i-1}$

$\boxed{e_i = C e_{i-1}^{\alpha}}$

$C^{1+\alpha} (e_{i-1})^{\alpha\alpha} = D C e_{i-1}^{\alpha} e_{i-1}$

$\alpha^2 = \alpha + 1$

$\boxed{\alpha = \dfrac{1+\sqrt{5}}{2} = 1{,}618}$ $\boxed{3}$

[2] Attention, il faut justifier votre réponse : juste recopier la valeur donnée dans les notes de cours ne rapporte rien :-(

Vous pouvez rédiger cet examen
avec un crayon (mais bien taillé !)

Question 3.1 Python !

```
def secante(x0,x1,tol,nmax,f):
  n = 0; delta = float("inf")
  f0 = f(x0)
  while abs(delta) > tol and n < nmax :
    n = n + 1
    f1 = f(x1)
    delta = -f1*(x1-x0)/(f1-f0)
    x0 = x1; f0 = f1
    x1 = x1 + delta
  return x1
```

Court mais correct !
Pas de commentaires !
Une seule évaluation de f(x1) !

Ce fût un désastre !

Une implémentation efficace ne nécessite qu'un unique calcul de la fonction f par itération. C'est cela qui définit le vrai coût de la méthode par itération. En conséquence, le correcteur a retiré un point par évaluation inutile de cette fonction. Et donc (sans doute à l'étonnement de beaucoup d'étudiants), cette question qui paraissait très simple a été très souvent désastreuse pour tous ceux qui se sont bêtement contentés de recopier la formule de manière servile.

En d'autres mots, il ne sert à rien de venir voir sa copie pour constater le désastre, si vous avez obtenu un seul point malgré l'écriture d'un très long code agrémenté de multiples commentaires totalement inutiles !

2. Démontrer qu'il existe une constante D en termes de $f'(x)$ et $f''(x)$ telle que

$$e_{i+1} = D\ e_i e_{i-1}$$!!

lorsque la méthode de la sécante converge et que $i \to \infty$.

Si la méthode converge, la fonction f tend progressivement vers zéro, on peut donc écrire :

$$x_{i+1} = x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}$$

$$e_{i+1} = e_i - \frac{f(x+e_i)(e_i - e_{i-1})}{f(x+e_i) - f(x+e_{i-1})}$$

$$e_{i+1} = \frac{e_i f(x+e_i) - e_i f(x+e_{i-1}) - e_i f(x+e_i) + e_{i-1} f(x+e_i)}{f(x+e_i) - f(x+e_{i-1})}$$

$$e_{i+1} = \frac{e_{i-1} f(x+e_i) - e_i f(x+e_{i-1})}{f(x+e_i) - f(x+e_{i-1})}$$

En effectuant les développements en série de Taylor :

$$e_{i+1} = \frac{e_{i-1}\left(e_i f'(x) + e_i^2 \frac{f''(x)}{2}\right) - e_i\left(e_{i-1} f'(x) + e_{i-1}^2 \frac{f''(x)}{2}\right))}{\left(e_i - e_{i-1}\right) f'(x)}$$

$$e_{i+1} = \left(e_{i-1} e_i\right) \frac{\left(e_i - e_{i-1}\right) f''(x)}{\left(e_i - e_{i-1}\right) 2 f'(x)}$$

On obtient alors finalement : $\boxed{D = \frac{f''(x)}{2f'(x)}}$

Oui, c'est dur !
Difficile, mais fait en séance !
Si, si !

Question 3.2

3. Finalement, en déduire[2] le taux de convergence α de la méthode de la sécante.

Lorsque $i \to \infty$, on peut écrire que :

$$e_{i+1} = D\ e_i e_{i-1}$$

Asymptotiquement, on peut écrire $e_{i+1} = C\,(e_i)^{\alpha}$

$$\begin{aligned} CC^{\alpha}\,(e_{i-1})^{\alpha\alpha} &= DC\,(e_{i-1})^{\alpha}\,e_{i-1} \\ CC^{\alpha}\,(e_{i-1})^{\alpha^2} &= DC\,(e_{i-1})^{\alpha+1} \end{aligned}$$

Cette égalité ne peut être satisfaite que si $\alpha^2 = \alpha + 1$.
L'unique racine positive de cette équation du second degré est le nombre d'or.

Nous avons donc bien une convergence en or : $\boxed{\alpha = \dfrac{1+\sqrt{5}}{2} = 1.618}$

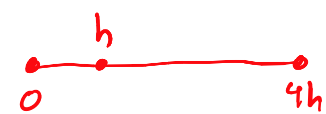
Question 3.3

Les engagés !

1 La dérivée des engagés !

Blondinet souhaite obtenir une estimation de la dérivée à l'origine d'une fonction $u(x)$ à partir de trois valeurs U_0, U_h et U_{4h} pour les abscisses 0, h et $4h$.
On va établir une formule de dérivation numérique en écrivant :

$$u'(0) \approx \underbrace{(u^h)'(0)}_{U'_0}$$

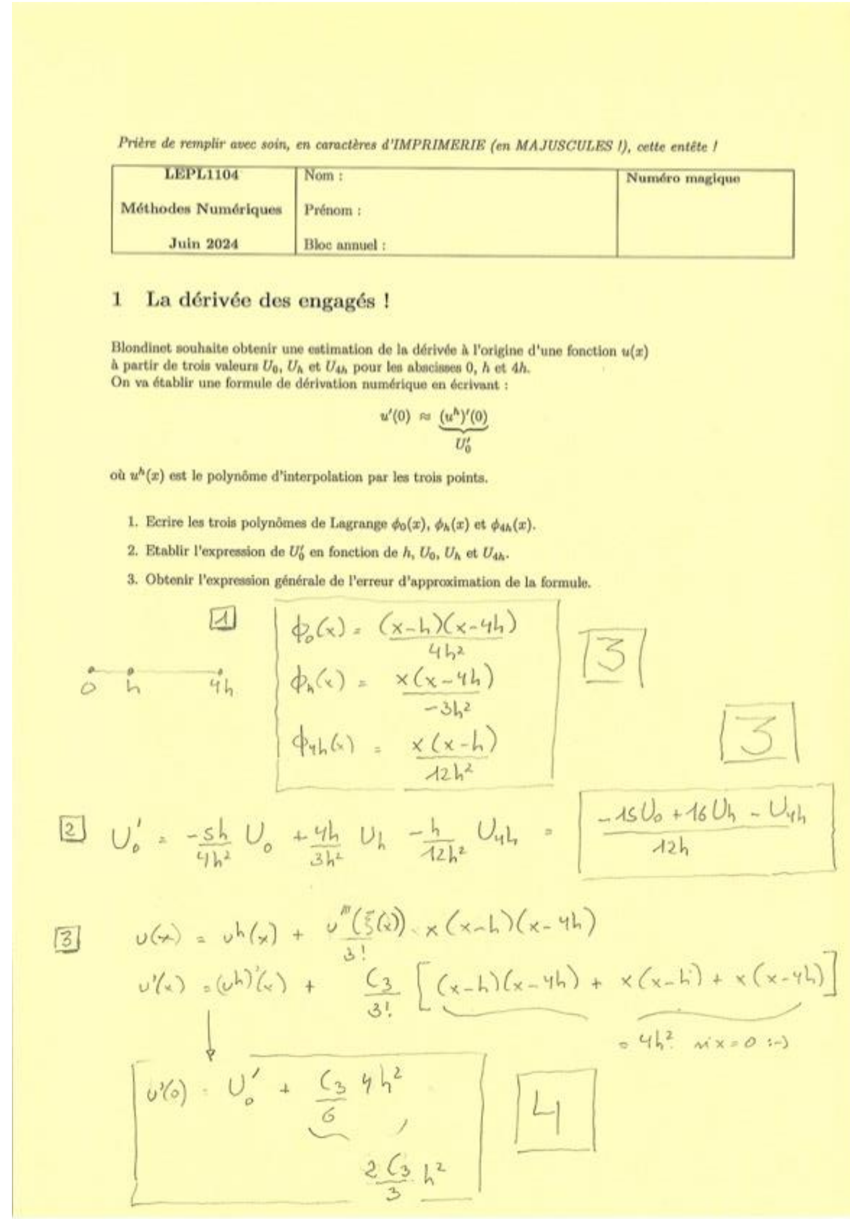


où $u^h(x)$ est le polynôme d'interpolation par les trois points.

1. Ecrire les trois polynômes de Lagrange $\phi_0(x)$, $\phi_h(x)$ et $\phi_{4h}(x)$.
2. Etablir l'expression de U'_0 en fonction de h, U_0, U_h et U_{4h}.
3. Obtenir l'expression générale de l'erreur d'approximation de la formule.

Exactement un simple verso !
Expliquer et justifier son résultat !
Il faut obtenir les valeurs correctes !

La solution...


Prière de remplir avec soin, en caractères d'IMPRIMERIE (en MAJUSCULES !), cette entête !

LEPL1104 Méthodes Numériques Juin 2024	Nom : Prénom : Bloc annuel :	Numéro magique

1 La dérivée des engagés !

Blondinet souhaite obtenir une estimation de la dérivée à l'origine d'une fonction $u(x)$ à partir de trois valeurs U_0, U_h et U_{4h} pour les abscisses 0, h et $4h$.
On va établir une formule de dérivation numérique en écrivant :

$$u'(0) \approx \underbrace{(u^h)'(0)}_{U_0'}$$

où $u^h(x)$ est le polynôme d'interpolation par les trois points.

1. Ecrire les trois polynômes de Lagrange $\phi_0(x)$, $\phi_h(x)$ et $\phi_{4h}(x)$.
2. Etablir l'expression de U_0' en fonction de h, U_0, U_h et U_{4h}.
3. Obtenir l'expression générale de l'erreur d'approximation de la formule.

$\boxed{1}$

0 h 4h

$$\boxed{\begin{aligned} \phi_0(x) &= \frac{(x-h)(x-4h)}{4h^2} \\ \phi_h(x) &= \frac{x(x-4h)}{-3h^2} \\ \phi_{4h}(x) &= \frac{x(x-h)}{12h^2} \end{aligned}}$$

$\boxed{3}$ $\boxed{3}$

$\boxed{2}$ $$U_0' = -\frac{5h}{4h^2}U_0 + \frac{4h}{3h^2}U_h - \frac{h}{12h^2}U_{4h} = \boxed{\frac{-15U_0 + 16U_h - U_{4h}}{12h}}$$

$\boxed{3}$ $$u(x) = u^h(x) + \frac{u'''(\xi(x))}{3!}\, x(x-h)(x-4h)$$

$$u'(x) = (u^h)'(x) + \frac{C_3}{3!}\Big[\underbrace{(x-h)(x-4h) + x(x-h) + x(x-4h)}_{= 4h^2 \text{ si } x=0 \text{ :-)}}\Big]$$

$$\boxed{u'(0) = U_0' + \underbrace{\frac{C_3}{6}\,4h^2}_{\frac{2C_3}{3}h^2}}$$

$\boxed{4}$

Vous pouvez rédiger cet examen avec un crayon (mais bien taillé !)

Le correcteur est en général très tolérant et honteusement généreux !

Une copie propre est toujours une bonne idée : vous aviez largement le temps pour la faire !

La question était courte et ne nécessitait pas beaucoup de calcul !

Question 1.1
So easy !

1. Ecrire les trois polynômes de Lagrange $\phi_0(x)$, $\phi_h(x)$ et $\phi_{4h}(x)$.

   Il faut juste écrire le 3 polynômes...

Il faut l'expression correcte !
C'est facile et donc toute erreur est impardonnable !
Cool, cool, cool !

$$
\begin{aligned}
\phi_0(x) &= \frac{(x-h)(x-4h)}{4h^2} \\
\phi_h(x) &= \frac{x(x-4h)}{-3h^2} \\
\phi_{4h}(x) &= \frac{x(x-h)}{12h^2}
\end{aligned}
$$

Cette question est vraiment facile et peu d'erreurs sont donc admises.
Ecrire la formule générale de polynôme de Lagrange n'est pas accepté comme réponse !

Question 1.2

Oui : c'était aussi simple que cela !
Même le Président du Sénat y arriverait !

2. Etablir l'expression de U'_0 en fonction de h, U_0, U_h et U_{4h}.

On écrit simplement :

$$U'_0 = \phi'_0(0)\ U_0 + \phi'_h(0)\ U_h + \phi'_{4h}(0)\ U_{4h}$$

En calcultant les dérivées de $\phi'_i(0)$

$$= -\frac{5h}{4h^2}\ U_0 + \frac{4h}{3h^2}\ U_h - \frac{h}{12h^2}\ U_{4h}$$

Et on peut donc finalement conclure : $$\boxed{U'_0 = \frac{-15U_0 + 16U_h - U_{4h}}{12}}$$

Cette question est une simple application d'une technique simple.
Vérifier que la somme des trois coefficients doit valoir zéro est bon check !
Ici, il faut juste être attentif dans les calculs :-)

Question 1.3
Taylor

3. Obtenir l'expression générale de l'erreur d'approximation de la formule.

Il suffit d'écrire des développements en série de Taylor pour U_h et U_{4h} :

$$
\begin{aligned}
U_0' &= \frac{1}{h}\left(-\frac{15}{12}U_0 + \frac{16}{12}U_h - \frac{1}{12}U_{4h}\right)\\
&= \frac{1}{h}\left(-\frac{15}{12}U_0 + \frac{16}{12}\left[U_0 + hU_0' + \frac{h^2}{2}U_0'' + \frac{h^3}{6}U_0''' + \ldots\right]\right.\\
&\qquad\left. - \frac{1}{12}\left[U_0 + 4hU_0' + \frac{16h^2}{2}U_0'' + \frac{64h^3}{6}U_0''' + \ldots\right]\right)
\end{aligned}
$$

En regroupant les termes

$$
U_0' = \underbrace{\left[-\frac{15}{12} + \frac{16}{12} - \frac{1}{12}\right]}_{=0} U_0 + \underbrace{\left[\frac{16}{12} - \frac{4}{12}\right]}_{=1} U_0' + \frac{h}{2}\underbrace{\left[\frac{16}{12} - \frac{16}{12}\right]}_{=0} U_0'' + \frac{h^2}{6}\underbrace{\left[\frac{16}{12} - \frac{64}{12}\right]}_{=-4} U_0''' + \ldots
$$

Et on peut donc finalement conclure :

$$
\boxed{u'(0) = \frac{-15U_0 + 16U_h - U_{4h}}{12} + \frac{2C_3}{3}h^2}
$$

Quelques étudiants ajoutent aussi les erreurs d'arrondi du calcul en virgule flottante...
Evidemment, c'est bien, mais ce n'était pas demandé :-)

Réformons !

$x^4 - x^3 - 3x^2 + 5x - 2 = 0$

$4x^3 - 3x^2 - 6x + 5$

$12x^2 - 6x - 6$ $\quad$ $24x - 6$

2 Réformons la méthode de Newton !

Georges-Louis utilise la méthode de Newton-Raphson pour calculer la racine $x = 1$ du polynôme

$$x^4 - x^3 - 3x^2 + 5x - 2 = 0$$

et il n'observe pas une convergence quadratique même avec $x_0 = 0.999$.

$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$

1. Décrire le calcul des itérations successives avec la méthode de Newton-Raphson.
2. Ecrire une fonction **python** : `x = iterateNewton(x0,tol,nmax)`
   qui applique la méthode de Nexton-Raphson pour ce problème : `x0` est le candidat initial, `tol` est la précision requise et `nmax` est le nombre maximal d'itérations. La fonction retourne `x`.
3. Pourquoi est-ce que Georges-Louis n'observe pas une convergence quadratique ?
4. Prédire le taux de convergence qu'observe Georges-Louis pour ce cas précis !
5. Proposer une méthode de Newton-Raphson réformée avec une convergence quadratique pour ce cas.

Exactement un simple verso !
Expliquer et justifier son résultat !
Il faut obtenir les valeurs correctes !

La solution...

Y a évidemment une astuce !
Les notes de bas de page : faut les lire !

Prière de remplir avec soin, en caractères d'IMPRIMERIE (en MAJUSCULES !), cette entête !

LEPL1104	Nom :	Numéro magique
Méthodes Numériques	Prénom :	
Juin 2024	Bloc annuel :	

2 Réformons la méthode de Newton !

Georges-Louis utilise la méthode de Newton-Raphson pour calculer la racine $x = 1$ du polynôme

$$x^4 - x^3 - 3x^2 + 5x - 2 = 0$$

et il n'observe pas une convergence quadratique même avec $x_0 = 0.999$.

1. Décrire le calcul des itérations successives avec la méthode de Newton-Raphson.
2. Ecrire une fonction python : `x = iterateNewton(x0,tol,nmax)` qui applique la méthode de Nexton-Raphson pour ce problème : `x0` est le candidat initial, `tol` est la précision requise et `nmax` est le nombre maximal d'itérations. La fonction retourne `x`.
3. Pourquoi est-ce que Georges-Louis n'observe pas une convergence quadratique ?
4. Prédire le taux de convergence qu'observe Georges-Louis pour ce cas précis !
5. Proposer une méthode de Newton-Raphson réformée avec une convergence quadratique pour ce cas.

$\boxed{1}$

$$x_{n+1} = x_n - \underbrace{\frac{x_n^4 - x_n^3 - 3x_n^2 + 5x_n - 2}{4x_n^3 - 3x_n^2 - 6x_n + 5}}_{\frac{(x_n-1)^3(x_n+2)}{(x_n-1)^2(4x_n+5)}}$$

$\boxed{4}$

$\boxed{2}$

```
def iterateNewton( ):
   n = 0 ; dx = tol+1 ; x = x0
   while (abs(dx)>tol and n<nmax)
      dx = (x-1)(x-2)/(4x+5)
      x += dx
   return(x)
```

$\boxed{3}$ $x = 1$ RACINE TRIPLE

$\boxed{4}$

CONVERGENCE LINEAIRE

$$e_{i+1} = e_i - \underbrace{\frac{f(x+e_i)}{f'(x+e_i)}}_{\frac{\cancel{f}+e_i\cancel{f'}+e_i^2\cancel{f''}/2+e_i^3f'''/6 \ldots}{\cancel{f'}+e_i\cancel{f''}+e_i^2f'''/2 \ldots}} = e_i - \frac{e_i}{3} = \frac{2}{3}e_i$$

$\boxed{3}$

$$f(x) = (g(x))^3$$

NR POUR g $\quad \dfrac{g}{g'}$

NR POUR f $\quad \dfrac{g^3}{3g^2g'} = \dfrac{3g}{g'} = \dfrac{f}{f'}$

$\boxed{5}$

$$x_{n+1} = x_n - \frac{3f_n}{f'_n}$$

$\boxed{3}$

OU

$$x_{n+1} = x_n - \frac{f''_n}{f'''_n}$$

OU

$$x_{n+1} = x_n - \frac{x^2+x-2}{2x+1}$$

MAIS PAS FACTORISER f ET f' PAR $(x-1)^2$!

Vous pouvez rédiger cet examen
avec un crayon (mais bien taillé !)

Question 2.1 : l'itération

1. Décrire le calcul des itérations successives avec la méthode de Newton-Raphson.

   Il suffit juste d'écrire la relation de récurence :

$$\boxed{x_{i+1} \quad = \quad x_I - \frac{f(x_i)}{f'(x_i)} \quad = \quad x_i - \frac{x_i^4 - x_i^3 - 3x_i^2 + 5x_i - 2}{4x_i^3 - 3x_i^2 - 6x_i + 5}}$$

Oui : il faut l'itération pour le polynôme de la question et donc calculer la dérivée de celui !
Ou il faut que l'implémentation qui suit la contienne au moins :-)

Il faut l'expression pour le problème posé !

2. Ecrire une fonction `python : x = iterateNewton(x0,tol,nmax)` qui applique la méthode de Nexton-Raphson pour ce problème : `x0` est le candidat initial, `tol` est la précision requise et `nmax` est le nombre maximal d'itérations. La fonction retourne `x`.

   Une implémentation possible est :

```
def iterateNewton(x0,tol,nmax):
  n = 0; delta = float("inf")
  x = x0
  while abs(delta) > tol and n < nmax :
    n = n + 1
    delta = (x-1)*(x-2)/(4x+5)
    x -= delta
  return x
```

Question 2.2 Python !

   Dans cette implémentation, on a tiré profit que

$$
\begin{array}{rclcl}
f(x) & = & x^4 - x^3 - 3x^2 + 5x - 2 & = & (x+2)(x-1)^3 \\
f'(x) & = & 4x^3 - 3x^2 - 6x + 5 & = & (4x+5)(x-1)^2
\end{array}
$$

   pour écrire l'incrément à chaque itération :-)
   Mais, cela n'est pas indispensable pour valider le code qui était d'une facilité déconcertante à écrire. C'était vraiment une sous-question qui était une partie vraiment très facile de cette question !

   Il est d'autant plus incompréhensible qu'un nombre constant d'étudiants font de manière systématique une impasse sur les codes informatiques et gaspillent bêtement des points si faciles à obtenir !

3. Pourquoi est-ce que Georges-Louis n'observe pas une convergence quadratique ?

Il suffit juste de noter que $f(1) = f'(1) = f''(1) = 0$ pour le polynôme considéré !
En d'autres mots, $x = 1$ est une racine triple de notre polynôme.
Newton-Raphson ne converge quadratiquement qu'au voisinage d'un racine simple !
Il fallait donc juste écrire :

$$\boxed{\textit{La méthode ne converge pas quadratiquement car } x = 1 \textit{ est une racine triple !}}$$

Mentionner un racine double au lieu d'une racine triple était pardonné par le correcteur !

Question 2.3
Georges- Louis n'est pas bourré !

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

$$e_{i+1} = e_i - \frac{f(x+e_i)}{f'(x+e_i)}$$

$$e_{i+1} = e_i - \frac{f(x) + f'(x)e_i + f''(x)e_i^2/2 + f'''(x)e_i^3/6}{f'(x) + f''(x)e_i + f'''(x)e_i^2/2}$$

$$\Big\downarrow \quad \textit{Comme } f(x) = f'(x) = f''(x) = 0 \textit{ pour } x = 1$$

$$e_{i+1} = e_i - \frac{6\ f'''(1)}{2\ f'''(1)}\ \frac{e_i^3}{e_i^2}$$

$$e_{i+1} = \tfrac{2}{3}\ e_i$$

$$\boxed{e_{i+1} = \frac{2}{3}\ e_i}$$

Question 2.4
Taux de convergence

La solution se trouvait dans une note en bas de la page 150 des notes de cours !

Il suffit de considérer que si x est une racine simple de $f(x)$, alors cette même valeur sera une racine double pour $g(x) = (f(x))^2$. L'application de la méthode de Newton-Raphson à g produit des incréments donnés par la relation :

$$\Delta x = \frac{-g(x)}{g'(x)} = \frac{-(f(x))^2}{2f(x)f'(x)} = \frac{-f(x)}{2f'(x)}$$

On observe que les incréments sont deux fois plus petits que pour la méthode de Newton-Raphson appliquée à f et on peut en déduire que :

$$\begin{aligned}
e_{i+1} &= x - x_{i+1} \\
&= x - \left(x_i - \frac{f(x_i)}{2f'(x_i)} \right) \\
&= \frac{2e_i f'(x_i) + f(x_i)}{2f'(x_i)} \\
&= \frac{2e_i f'(x_i) - e_i f'(x_i) - e_i^2 f''(\zeta)/2}{2f'(x_i)} \\
&= \frac{e_i}{2} + Ce_i^2
\end{aligned}$$

La note de bas de page !

$$x_{i+1} = x_I - \frac{3f(x_i)}{f'(x_i)}$$

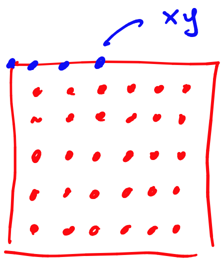
$$x_{i+1} = x_I - \frac{f''(x_i)}{f'''(x_i)}$$

$$x_{i+1} = x_I - \frac{x^2 + x - 2}{2x + 1}$$

Question 2.5
Les méthodes réformées !

Et zou Raoul !

3 Une équation aux dérivées partielles pour Raoul



Sur un domaine carré $(x, y) \in [0, 1] \times [0, 1]$, on va résoudre numériquement :

$$\frac{\partial u}{\partial t}(x, y, t) = \alpha \frac{\partial^2 u}{\partial x \partial y}(x, y, t)$$

A l'instant $t = 0$, nous imposons $u(x, y, 0) = xy$.
Aux instants $t > 0$, on maintient cette valeur uniquement sur les côtés du domaine carré.
Les $m \times m$ valeurs nodales au temps $n + 1$ sont obtenues à partir de celles au temps n avec la relation :

$$U^{n+1}_{j,k} = U^n_{j,k} + \underbrace{\frac{\alpha \Delta t}{4(\Delta x)^2}}_{\beta} \left(U^n_{j+1,k+1} + U^n_{j-1,k-1} - U^n_{j+1,k-1} - U^n_{j-1,k+1} \right)$$

où $U^n_{j,k}$ est l'approximation de $u(X_j, Y_k, T_n)$ avec $X_j = j\Delta x$, $Y_k = k\Delta x$ et $T_n = n\Delta t$.

1. Donner les unités du paramètre α.

2. Ecrire une fonction **python** : `U = raoulEuler(m,n,beta)`
   qui effectue n itérations temporelles et retourne un tableau **numpy** avec $m \times m$ valeurs nodales.

3. Pour quel pas de temps, la méthode est-elle stable ?
   Il faut donc estimer[1] le facteur d'amplification d'une pertubation $U^n_{j,k} = U^n e^{ik_x X_j} e^{ik_y Y_k}$

La solution...

Prière de remplir avec soin, en caractères d'IMPRIMERIE (en MAJUSCULES !), cette entête !

LEPL1104 Méthodes Numériques Juin 2024	Nom : Prénom : Bloc annuel :	Numéro magique

3 Une équation aux dérivées partielles pour Raoul

Sur un domaine carré $(x,y) \in [0,1] \times [0,1]$, on va résoudre numériquement :

$$\frac{\partial u}{\partial t}(x,y,t) = \alpha \frac{\partial^2 u}{\partial x \partial y}(x,y,t)$$

A l'instant $t = 0$, nous imposons $u(x,y,0) = xy$.
Aux instants $t > 0$, on maintient cette valeur uniquement sur les côtés du domaine carré.
Les $m \times m$ valeurs nodales au temps $n+1$ sont obtenues à partir de celles au temps n avec la relation :

$$U^{n+1}_{j,k} \;=\; U^n_{j,k} + \underbrace{\frac{\alpha \Delta t}{4(\Delta x)^2}}_{\beta} \left(U^n_{j+1,k+1} + U^n_{j-1,k-1} - U^n_{j+1,k-1} - U^n_{j-1,k+1} \right)$$

où $U^n_{j,k}$ est l'approximation de $u(X_j, Y_k, T_n)$ avec $X_j = j\Delta x$, $Y_k = k\Delta x$ et $T_n = n\Delta t$.

1. Donner les unités du paramètre α.
2. Ecrire une fonction python : `U = raoulEuler(m,n,alpha)`
   qui effectue n itérations temporelles et retourne un tableau numpy avec $m \times m$ valeurs nodales.
3. Pour quel pas de temps, la méthode est-elle stable ?
   Il faut donc estimer[1] le facteur d'amplification d'une pertubation $U^n_{j,k} = U^n e^{ik_x X_j} e^{ik_y Y_k}$

$\boxed{1}$ $\alpha \; \left[\frac{m^2}{s}\right]$ $\boxed{2}$

$\boxed{2}$ $\boxed{4}$

```
def raoulEuler(m,n,beta):
    alpha = linspace(0,1,m)
    X,Y = meshgrid(alpha,alpha)
    U = X*Y
    for t in range(n):
        U[1:-1,1:-1] += beta*[U[2:,2:] + U[:-2,:-2]
                              - U[2:,:-2] - U[:-2,2:])
    return(U)
```

$\boxed{3}$

$$U^{n+1}_{jk} = U^n_{jk}\left(1 + \beta\left[e^{ik_x\Delta x}e^{ik_y\Delta x} + e^- e^- - e^- e^+ - e^+ e^-\right]\right)$$

$$\left(1 + \beta\left[\left(e^{ik_x\Delta x} - e^{-ik_x\Delta x}\right)\left(e^{ik_y\Delta x} - e^{-ik_y\Delta x}\right)\right]\right)$$

$\boxed{2}$ $\left(1 - 4\beta \sin(k_x\Delta x)\sin(k_y\Delta x)\right)$

par exemple $=1$, $=-1$

IL FAUT TJRS $\left|1 - 4\beta \sin(\,)\sin(\,)\right| < 1$

MAIS $|1 + 4\beta| > 1$ EST POSSIBLE

$\boxed{2}$ INCONDITIONNELLEMENT INSTABLE

[1] La relation $2i\sin(\theta) = e^{i\theta} - e^{-i\theta}$ pourrait être utile où i est le nombre imaginaire.

Vous pouvez rédiger cet examen
avec un crayon (mais bien taillé !)

Question 3.1
So easy !

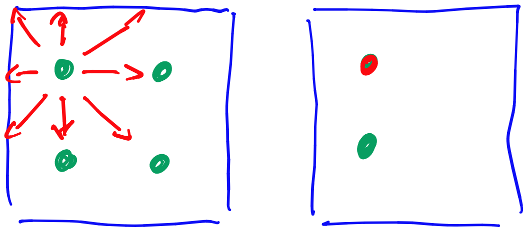
$$\underbrace{\frac{\partial u}{\partial t}(x, y, t)}_{s} = \underbrace{\alpha}_{\left[\frac{m^2}{s}\right]} \underbrace{\frac{\partial^2 u}{\partial x \partial y}(x, y, t)}_{m^2}$$

$$\boxed{\alpha\left[\frac{m^2}{s}\right]}$$

Question 3.2 : python !
Oui : c'est simple !

```
def raoulEuler(m,n,beta) :
  alpha = linspace(0,1,m)
  X,Y = meshgrid(alpha,alpha)
  U = X*Y
  for t in range(n):
    U[1:-1,1:-1] += beta*(U[2:,2:] + U[:-2,:-2]
                                  - U[2:,:-2] - U[:-2,2:] )
  return U
```

Le code !
Trop compliqué !



```
def raoulBasicEuler(m,n,beta) :
  alpha = linspace(0,1,m)
  X,Y = meshgrid(alpha,alpha)
  U = X*Y
  Unew = U.copy()
  for t in range(n):
    for i in range(1,m-1):
      for j in range(1,m-1):
        Unew[i,j] = U[i,j] + beta*(U[i+1,j+1] + U[i-1,j-1]
                                  - U[i+1,j-1] - U[i-1,j+1] )
    U = Unew.copy()
  return U
```

Question 3.3 : un peu complexe !

La pertubation évoluera comme suit lors d'un pas de temps :

$U_j^n = U^n \exp^{ik_x j\Delta x} \exp^{ik_y j \Delta x}$

NBR COMPLEXE

$$
\begin{aligned}
U_{j,k}^{n+1} &= U_{j,k}^n + +\beta\Big(U_{j+1,k+1}^n + U_{j-1,k-1}^n - U_{j+1,k-1}^n - U_{j-1,k+1}^n\Big) \\
&= U_{j,k}^n\Big(1+\beta\Big(e^{ik_x\Delta x}e^{ik_y\Delta x} + e^{-ik_x\Delta x}e^{-ik_y\Delta x} - e^{ik_x\Delta x}e^{-ik_y\Delta x} - e^{-ik_x\Delta x}e^{ik_y\Delta x}\Big)\Big) \\
&= U_{j,k}^n\Big(1+\beta\Big(\Big(e^{ik_x\Delta x} - e^{-ik_x\Delta x}\Big)\Big(e^{ik_y\Delta x} - e^{-ik_y\Delta x}\Big)\Big)\Big) \\
&= U_{j,k}^n\Big(1-4\beta\Big(\sin(k_x\Delta x)\sin(k_y\Delta x)\Big)\Big)
\end{aligned}
$$

OK SI CFL

$-1 < 1 - 4\beta < 1$

OUI ICI CELA POURRAIT MARCHER

Pour que la pertubation ne s'accroisse pas, il faut toujours que :

$$\Big|1-4\beta\Big(\sin(k_x\Delta x)\sin(k_y\Delta x)\Big)\Big| \leq 1$$

LES 3 CAS !

$|1-4\beta| \leqslant 1$

$[\,|1+4\beta| \leqslant 1$ KO

$|1| \leqslant 1$ OK

CAS DEFAVORABLE

POUR $\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$

$|1 - \beta(2\cos(\dots) - 2|$ $\rightarrow$

Mais, en prenant $k_x\Delta x = \pi/2$ et $k_y\Delta x = -\pi/2$,

LES 3 CAS !

$|1-4\beta|$ OK SI CFL

$|1-2\beta|$

$|1|$ OK

$$1 \leq |1+4\beta|$$