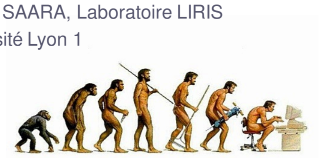
http://licence-info.univ-lyon1.fr/LIFAMI

LIFAMI MATH INFO APP

LIFAMI
APPLICATIONS EN MATH ET INFO

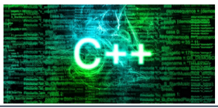
Alexandre Meyer
Equipe SAARA, Laboratoire LIRIS
Université Lyon 1



LIFAP1

Apprendre les bases de la programmation

- Variables, type
- Test : if, ||, &&, ==, !=
- Boucle : for, while
- Fonction / procédure
  - Paramètre Donnée ou Donnée/Résultat
- Chaine de caractères
- Tableau : 1D, 2D
- Structure



Mise en œuvre avec une sous partie de C/C++

Base commune
à tous les langages

LIFAMI

- Algo/Prog
  - Peu de choses nouvelles par rapport à LIFAP1
- Math (expliquées par des informaticiens, aie!)
  - Nombres complexes (Thème 1)
  - Interpolation et monde discret (Thème 2)
  - Calcul vectoriel
  - Dérivée et intégrale
  - Équations différentielles
- AMI = Applications des Math et l'Informatique
  - Physique : système de particules/masse-ressorts (Thème 3)
  - Biologie : simulation d'un éco-système (Thème 4)
  - Economie : simulation économique (Thème 5)

➔ **Tout ceci n'est qu'un prétexte pour coder**

LIFAMI : évaluation

- 1 TP à la maison
  - Code démineur/memory
- 1 interro en TD
- 1 TP noté court
- 1 TP noté long
  - juste avant Noël
- 1 mini-projet à la maison
  - ~finir intégralement un des thèmes 3, 4 ou 5
- 1 CCF
- Rattrapage en juin

LIFAMI
APPLICATIONS EN MATH ET INFO

Nombres complexes

But de LIFAMI
- montrer que les math et l'info peuvent s'associer pour résoudre des problèmes applicatifs
- vous faire coder (on espère avec des applis suscitant de l'intérêt !!!)

➔ Les nombres complexes, déjà vu en math, ont toutes ces bonnes propriétés

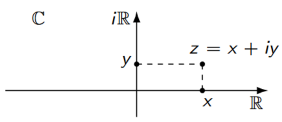
Nombres complexes

Un nombre complexe est un nombre de la forme

$\boxed{z = x + i.y}$

où x et y sont des nombres réels et i est un symbole ayant la propriété de $\boxed{i^2 = -1}$

Le plan complexe :



Nombres complexes

- Pour un nombre complexe

  z = x + i.y

- **Partie réelle**, le nombre réel Re(z) = x
- **Partie imaginaire**, le nombre réel Im(z) = y

- Le conjugué complexe est le nombre complexe

  $\overline{\mathbf{z}}$ **= x - i.y**

Nombres complexes : opérations

- Soit les deux nombres complexes

  z = x + i.y et **w = u + i.v**

- **Egalité** : si et seulement si x=u et y=v

- **Addition** : z+w = (x+u) + i.(y+v)
- **Soustraction :** z-w = (x-u) + i.(y-v)

- **Produit par scalaire :** λ.z = λ.x + i.(λ.y)

- **Multiplication :** z.w = xu + iyu + ixv + i²yv
- = (x.u-y.v) + i.(x.v+y.u)

Nombres complexes : opérations

- Le nombre complexe **z = x + i.y** est associé à son image, le point M(x,y) dans le plan complexe C

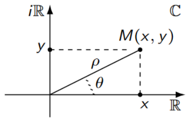
- **Module de z** est la distance |z| = ||OM||

  $$|z| = \sqrt{x^2 + y^2} = \rho$$

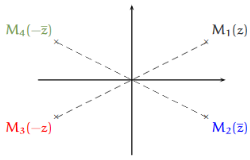
- **Argument de z** est l'angle θ comme sur la figure

  $$\boxed{\mathrm{Re}(z) = x = \rho\cos\theta}$$

  $$\boxed{\mathrm{Im}(z) = y = \rho\sin\theta}$$



Opposé, conjugué d'un nombre complexe



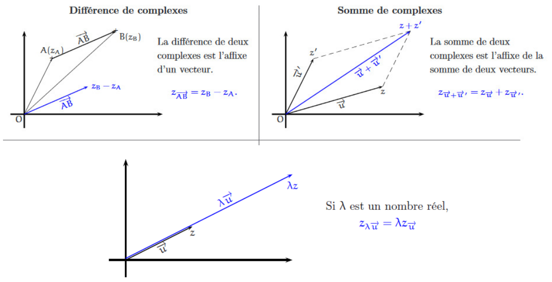
Le point M_2 d'affixe $\bar{z}$ est le symétrique par rapport à (Ox) du point M_1 d'affixe z.

Le point M_3 d'affixe $-z$ est le symétrique par rapport à O du point M_1 d'affixe z.

Le point M_4 d'affixe $-\bar{z}$ est le symétrique par rapport à (Oy) du point M_1 d'affixe z.

Interprétations géométriques

- Lien entre nombre complexe et calcul vectoriel



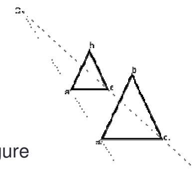
Translation (*Translate* en anglais)

- Soit un triangle ABC représenté par 3 complexes a,b et c
- Addition de ces complexes par un complexe u
  - a_1 = a+u
  - b_1 = b+u
  - c_1 = c+u
- Si
  - u=i ?
  - u=1 ?
  - u=1+i ?

➔ Translation de la figure d'un vecteur u
Le neutre étant u=0+0i = 0

Changement d'échelle (*scale* en anglais)

- Soit un triangle ABC représenté par 3 complexes a, b et c
- Multiplication de ces complexes par un réel
  - $a_1 = \lambda.a$
  - $b_1 = \lambda.b$
  - $c_1 = \lambda.c$
- Si
  - $\lambda=2$?
  - $\lambda=0.5$?
  - $\lambda=0$?



➔ Changement d'échelle de la figure

➔ Homothétie de centre O et de rapport λ

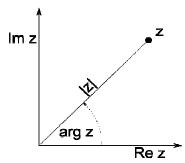
Le neutre étant λ=1

Nombres complexes : forme exponentielle

Tout nombre complexe *z* peut s'écrire sous une forme trigonométrique

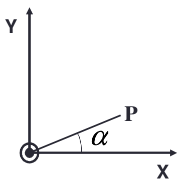
$\boxed{z = r.(\cos(\theta) + i.\sin(\theta))}$ avec r ≥ 0, forme polaire

$\boxed{z = r.e^{i\theta}}$ forme exponentielle



Rotation d'un angle θ autour de l'origine O

- Soit le point P avec pour image le complexe z

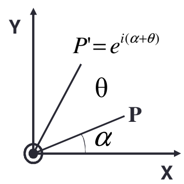


$$P = (x \quad y)$$
$$= (r.\cos(\alpha) \quad r.\sin(\alpha))$$

$$z = r.(\cos(\alpha) + i\sin(\alpha))$$
$$= r.e^{i\alpha}$$

Rotation d'un angle θ autour de l'origine O

- Soit le point P avec pour image le complexe z
- Soit le complexe $R=e^{i\theta}$
- P' le point résultat de la rotation de P autour de l'origine O d'un angle θ

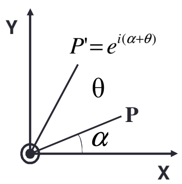


$$P' = r.e^{i.(\alpha+\theta)}$$
$$= r.e^{i\alpha}.e^{i\theta}$$
$$= P.R$$

Interprétations géométriques

Multiplier un complexe z (image du point P) par le complexe $e^{i\theta}$ de module 1 équivaut à faire tourner P(z) d'un angle θ autour de l'origine O

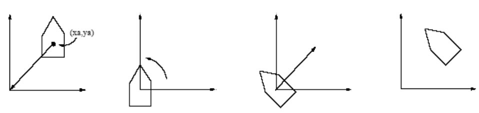
➔ Propriété très sympa !



$$P' = r.e^{i.(\alpha+\theta)}$$
$$= r.e^{i\alpha}.e^{i\theta}$$
$$= P.R$$

Pour tourner une figure sur elle-même

- Il faut faire dans l'ordre
  - Translation de $(-x_a, -y_a)$
  - Rotation d'un angle θ
  - Translation de (x_a, y_a)



Et en C++ ?

```
struct Complex
{
    float x,y;
};

Complex make_complex(float r, float i)
{
    Complex c;
    c.x = r;
    c.y = i;
    return c;
}

int main()
{
        Complex a = make_complex(2,5);
```

Et en C++

```
Complex comAdd(Complex a, Complex b)
{
        Complex r;
        r.x = a.x+b.x;
        r.y = a.y+b.y;
        return r;
}

int main()
{
        Complex c1 = make_complex(2,5);
        Complex c2 = make_complex(1,1);
        Complex c;
        c = comAdd(c1,c2);

        // Il serait vraiment pratique de pouvoir écrire
        // c = c1 + c2;
        …
```

C++ et les operator+, …

```
Complex operator+(Complex a, Complex b)
{
        Complex r;
        r.x = a.x+b.x;
        r.y = a.y+b.y;
        return r;
}

int main()
{
        Complex c1 = make_complex(2,5);
        Complex c2 = make_complex(1,1);
        Complex c;
        c = c1 + c2;  // On peut grâce à la
                      // fonction particulière
                      // opérator+

        …
```

C++ et l'operator-

```
…
Complex operator-(Complex a, Complex b)
{
        Complex r;
        r.x = a.x – b.x;
        r.y = a.y – b.y;
        return r;
}

int main()
{
        Complex c1 = make_complex(2,5);
        Complex c2 = make_complex(1,1);
        Complex c;
        c = c1 + c2 + c1 – c1;    // C'est bien pratique
        …
```

C++ et l'operator*

```
…
Complex operator*(float a, Complex b)
{
        Complex r;
        r.x = a * b.x;
        r.y = a * b.y;
        return r;
}

int main()
{
        Complex c1 = make_complex(2,5);
        Complex c2 = make_complex(1,1);
        Complex c;
        c = c1 – 5.0 * c2;        // C'est bien pratique
        …
```

C++ et l'operator*

```
…
Complex operator*(Complex a, Complex b)
{
        Complex r;
        r.x = a.x*b.x – a.y*b.y;
        r.y = a.x*b.y + a.y*b.x;
        return r;
}

int main()
{
        Complex c1 = make_complex(2,5);
        Complex c2 = make_complex(1,1);
        Complex c;
        c = c1 * c2;       // C'est bien pratique
        …
```

Nombres complexes

Survol des exercices de TD/TP

+

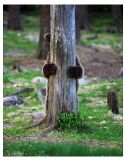
Démo

Nombres complexes : conclusions

- Représentation extrêmement pratique pour les transformations du plan (donc en 2D)
  - Translation
  - Changement d'échelle (homothétie par rapport à un centre/facteur)
  - Rotation par rapport à un centre, d'un angle
- L'autre représentation classique pour ces transformations sont les matrices (Semestre 2).
- Et en 3D ?
  - Quaternion : super nombre complexe avec i,j,k
  - $i^2 = j^2 = k^2 = ijk = -1$
  - Vous verrez ca en Master

LIFAMI
APPLICATIONS EN MATH ET INFO

Interpolation et monde discret



Informatique = un monde discret

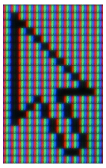
- Un **système discret** est un ensemble qui introduit une relation entre des variables d'entrée et des variables de sortie, dans lequel ces variables ne peuvent avoir qu'un nombre fini de valeurs, par opposition à un système continu, dans lequel entre deux valeurs de variables, quelles qu'elles soient, on peut toujours supposer une valeur intermédiaire.

➔ Ceci est un problème propre à l'informatique. Il y a des théories mathématiques pour aider et comprendre. On va voir des cas simples mais le problème est courant en informatique

Monde discret : par ex. une image

- Rectangle (2D) : tableau de pixels (=picture element)

L'image semble continue car très fine.



Mais elle est composée de pixels

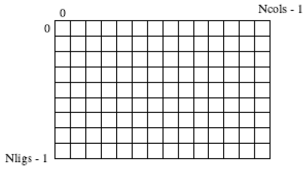
"Pixels"

Monde discret : par ex. une image

Un "Pixel" est un échantillon

Monde discret : par ex. une image

- Image = tableau 2D de pixels
  - nombre de lignes,
  - nombre de colonnes,
  - format des pixels (bit, niveaux de gris, niveaux de couleurs)
  - compression éventuelle.



Monde discret : par ex. une image

- Prendre une image avec la webcam de l'amphi

→Essayer de reconnaitre un visage du fond ou lire un texte en zoomant

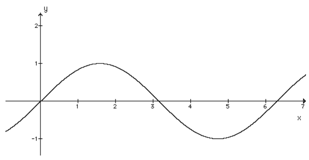
- Le truc des films policiers qui zoome à l'infini dans une image discrète n'est pas possible …

  → il y a un nombre de données (pixels) finis, on ne peut que difficilement réinventer de l'information là où il n'y en a pas

Informatique = un monde discret

Par exemple une fonction *f* analytique en math

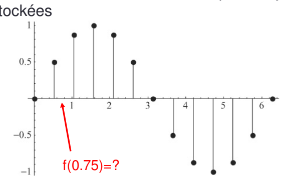
- On peut faire f(x) pour n'importe quel x réel
- Même si x est irrationnel



Informatique = un monde discret

La fonction *f(x)* en informatique `float f(float f)`

- On stocke N valeur de *f* dans un tableau
  - Par exemple ici on stocke pour x=0, 0.5, 1, 1.5, 2, 2.5, 3, …
- Si on veut la valeur de f(0.75) ?
  - 0.75 est au milieu entre 0.5 et 1 donc on peut moyenner les deux valeurs stockées



Monde numérique

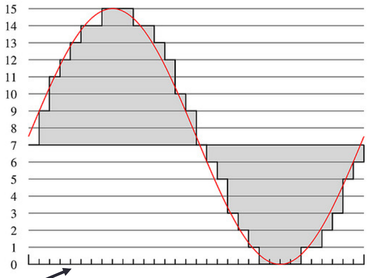


Tableau de valeurs (façon tableau de LIFAP1)

Monde discret : par ex. le son

Fonctionnement d'un microphone

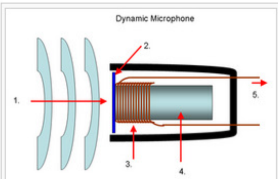
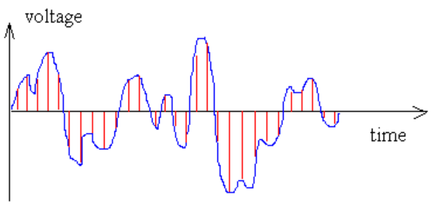


Schéma du microphone dynamique : 1.Onde sonore, 2.Membrane, 3.Bobine mobile, 4.Aimant, 5.Signal électrique.

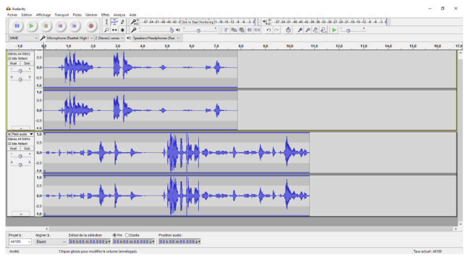
Monde discret : par ex. le son

- A la sortie du microphone → courbe bleu : continue
- En informatique, nous stockons que les valeurs rouges
  - Valeurs discrètes = échantillons
- A partir des valeurs rouge → reconstruction du signal (bleu)



Monde discret : par ex. le son

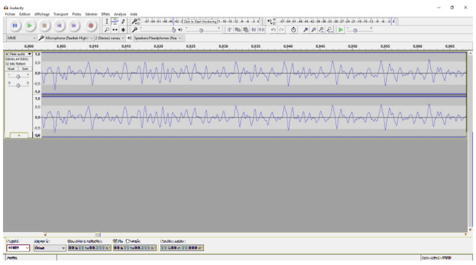
- Une courbe de son dans un logiciel de traitement du son



DEMO de AUDACITY

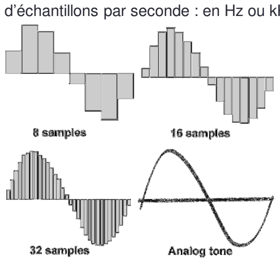
Monde discret : par ex. le son

- Une courbe de son dans un logiciel de traitement du son
  - Zoom sur la courbe



Monde discret : par ex. le son

- Le nombre d'échantillons (*samples*) a de l'importance
- On parle de fréquence d'échantillonnage
  - Nombre d'échantillons par seconde : en Hz ou kHz



Monde discret : par ex. le son

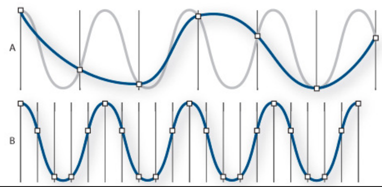
- Le nombre d'échantillons (*samples*) a de l'importance
  - Taille du fichier

Taille d'un fichier d'une minute d'enregistrement

Fréquence	Quantification	en mono	en stéréo
11,025 kHz	8 bits	660 ko	1,3 Mo
	16 bits	1,3 Mo	2,6 Mo
22,05 kHz	8 bits	1,3 Mo	2,6 Mo
	16 bits	2,6 Mo	5,3 Mo
44,1 kHz	8 bits	2,6 Mo	5,3 Mo
	16 bits	5,3 Mo	10,6 Mo
48 kHz	16 bits	5,8 Mo	11,5 Mo
	24 bits	8,6 Mo	17,3 Mo

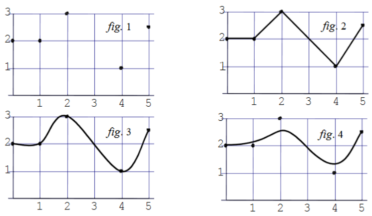
Monde discret : par ex. le son

- Le nombre d'échantillons (*samples*) a de l'importance
  - *Courbe bleu = courbe reconstruite ; Courbe grise = courbe en entrée*
  - *Traits gris verticaux = échantillons stockés dans l'ordinateur*
  - *Cas A → la courbe reconstruite bleu ne correspond pas du tout (**problème d'aliasing, pas assez d'échantillons**)*
  - *Cas B, assez d'échantillons → la courbe reconstruite est proche de celle en entrée*



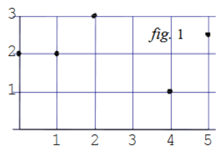
Reconstruction par interpolation

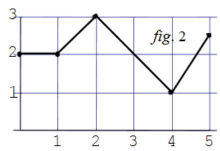
- Reconstruction par interpolation = retrouver la courbe à partir des points



Reconstruction par interpolation

- Interpolation
  - Interpolation linéaire = placer une droite entre les points
  - En LIFAMI, nous ne verrons que cette manière de reconstruire la courbe

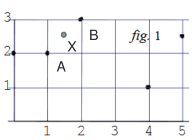


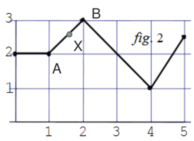


Interpolation linéaire = moyenne pondérée

- Soit A (1, f(1)=2) et B(2, f(2)=3)
- Par interpolation f(1.5) = (2+3)/2 = 2.5

- Comment calculer f(x) pour tout x dans l'interval [1;2] ?
  - Soit X(x, f(x))
  - Pente entre A et B : p_{AB}=(f(2)-f(1)) / (2-1) = 1
  - Pente entre A et X identique : p_{AX}=(f(x)-f(1)) / (x-1) = p_{AB}
    donc f(x) = p_{AB}.(x-1) + f(1) = 1.(x-1) + 2 = x+1

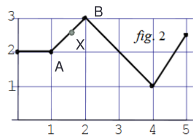




Interpolation linéaire = moyenne pondérée

Cas général

- Soit A (a, f(a)) et B(b, f(b))
- Calculer f(x) avec x dans [a;b] ?



$$p_{AB} = \frac{f(b) - f(a)}{b-a}$$

$$p_{AX} = \frac{f(x) - f(a)}{x-a} = p_{AB}$$

$$donc$$

$$f(x) = \frac{(f(b) - f(a)).(x-a)}{b-a} - f(a)$$

$$= f(a).(1 - \frac{x-a}{b-a}) + f(b).\frac{x-a}{b-a} = f(a).(1-t) + f(b)t \quad \text{avec} \quad t = \frac{x-a}{b-a}$$

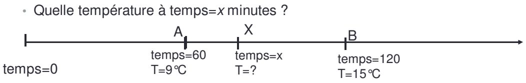
Moyenne pondérée
entre f(a) et f(b)
avec comme poids t et (1-t)

Moyenne pondérée entre f(a) et f(b) avec comme poids t et (1-t)

$$f(x) = f(a).(1-t) + f(b)t \quad \text{avec} \quad t = \frac{x-a}{b-a} \text{ avec } t \in [0;1]$$

Exemple : une station météo fait des mesures de température

- à temps=60min, Température=9°C
- à temps=120min, Température=15°C
- Quelle température à temps=*x* minutes ?

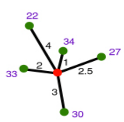


$$t = \frac{x-a}{b-a} = \frac{AX}{AB} = \frac{x-60}{120-60} = \frac{x-60}{60} \quad \text{et} \quad f(x) = 9.(1-t) + 15t$$

Par exemple temps = 70min $t = \frac{70-60}{60} = \frac{1}{6}$ $f(x) = 9.\frac{5}{6} + 15.\frac{1}{6} = 10$

Interpolation par distance inverse

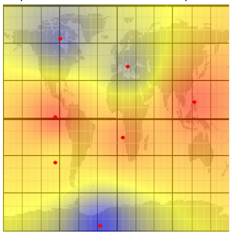
- Le but d'une interpolation est de calculer une moyenne pondérée en
  - donnant un poids fort à une valeur de référence quand on est proche
  - donnant un poids faible à une valeur de référence quand on est loin
- Exemple exercice de TD
  - N stations météo sur une carte 2D
  - Chaque station mesure une température
  - Calculer la température d'un point de la carte quelconque par interpolation
    - Moyenne de toutes les stations avec comme poids 1 / distance_à_la_station



$$Z(x) = \frac{\Sigma w_i z_i}{\Sigma w_i} = \frac{\frac{34}{1^2} + \frac{33}{2^2} + \frac{27}{2.5^2} + \frac{30}{3^2} + \frac{22}{4^2}}{\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{2.5^2} + \frac{1}{3^2} + \frac{1}{4^2}} = \mathbf{32.38}$$

Interpolation par distance inverse

- Exemple exercice de TD/TP
  - N stations météo sur une carte 2D
  - Chaque station mesure une température
  - Calculer la température sur toute la carte par interpolation



Utilisation de l'interpolation

Survol des exercices de TD/TP

\+

Démo

Monde discret : conclusion

- En informatique, stocke que des échantillons de valeurs
  - Nombre d'échantillons peut être grand pour faire croire au continue
  - Doit être grand dans certains cas
- Interpolation linéaire est une solution simple pour reconstruire les valeurs manquantes : « une droite » entre les points
- Problème courant en informatique
  - Ne paniquer pas si il vous échappe un peu … comprenez le sur les exemples simples (droite en couleur, température)
- Nombreuses manières de reconstruire le signal
  - Parfois assez bouffante en se basant sur des bases de données

  ➔ Zoom dans une image peut redonner un visage statistiquement plausible mais pas le visage du tueur

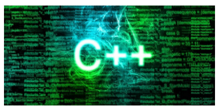
Exemple de reconstruction

- Image en entrée

- Image reconstruite
  - gauche : interpolation linéaire
  - droite : en utilisant une base de données d'images

TD/TP

- Regarder les consignes
- Regarder un code Grapic de base



LIFAMI

MATH INFO APP