

Modules de formation 2022





Bioinformatics platform dedicated to the genetics and genomics of tropical and Mediterranean plants and their pathogens

genome assembly SNP detection
phylogeny structural variation
comparative genomics transcriptome assembly differential expression
GWAS pangenomics
population genetics metagenomics
polyploidy



Rice



Banana



Palm



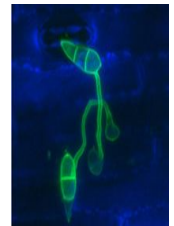
Sorghum



Coffee



Cassava



Magnaporthe



Charte



Larmande Pierre
Orjuela-Bouniol Julie
Sabot François
Tando Ndomassi
**Tranchant-Dubreuil
Christine**



Comte Aurore
Dereeper Alexis
Ravel Sébastien



Bocs Stephanie
Boizet Alice
De Lamotte Frédéric
Droc Gaetan
Dufayard Jean-François
Hamelin Chantal
Martin Guillaume
Pitollat Bertrand
Ruiz Manuel
Sarah Gautier
Summo Marilyne



Rouard Mathieu
Guignon Valentin
Catherine Breton



Sempere Guilhem

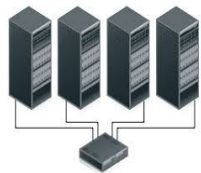
Workflow manager

TOOLBOX
Toolbox for generic NGS analyses

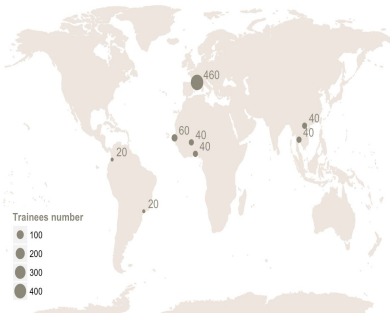
SNAKEMAKE

Galaxy

HPC and trainings....



37 courses organized last 7 years



IRD
Institut de Recherche
pour le Développement

cirad

Genome Hubs & Information System



Gigwa

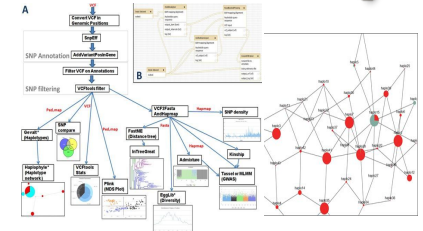
SNPs and Indels

GreenPhyl

Family Id	Family Name	Number of sequences	Status
GP000010	Cytochrome P450 superfamily	6542	●
GP000017	AP2/EREBP transcription factor family: ERF/ERF3 group (partial)	5142	●
GP000020	NAC transcription factor family	4574	●
GP000028	MADS transcription factor family		
GP000018	Hamem peroxidase superfamily		
GP000095	General substrate transporter superfamily		
GP000022	Subtilisin-like Serine Proteases family		
GP000019	NPF, NRT1/PTR FAMILY		

Gene families

SNIPlay



<https://github.com/SouthGreenPlatform>



@green_bioinfo

I-Trop

Plant & Health Bioinformatics Platform



<https://bioinfo.ird.fr/>



AUORE
COMTE

IE bioinfo



ALEXIS
DEREEPER

IE bioinfo



BRUNO
GRANOULLAC

IE systèmes
d'information



JULIE
ORJUELA

IE bioinfo



NDOMASSI
TANDO

IE systèmes



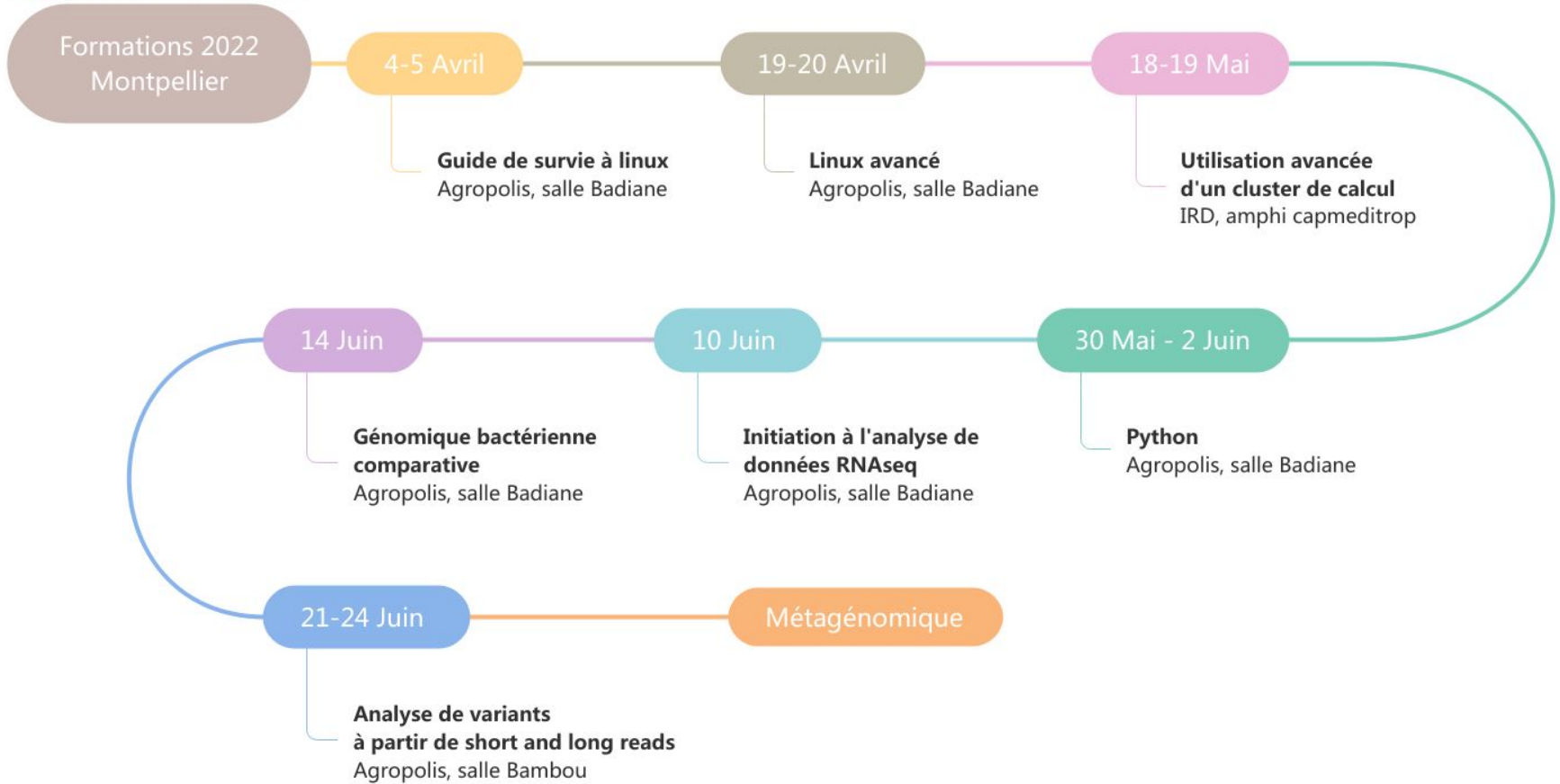
CHRISTINE
TRANCHANT

IR bioinfo

bioinfo@ird.fr



[@ltropBioinfo](https://twitter.com/ltropBioinfo)



Modules de formation 2022

- Toutes nos formations :
<https://southgreenplatform.github.io/trainings/>
- Topo & TP : [Linux For Jedi](#)
- Environnement de travail : [Logiciels à installer](#)

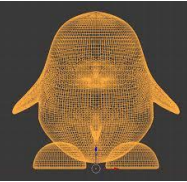




Linux Avancé

www.southgreen.fr

<https://southgreenplatform.github.io/trainings>



The objectif!

Optimiser vos analyses bioinformatiques sur un cluster en utilisant la puissance de Linux



Applications

- Travailler avec de larges volumes de données (eg.: fastq, bam, gff, vcf).
- Filtrer rapidement des fichiers volumineux pour par ex substituer un motif, filtrer sur la taille de séquence, sur un chromosome
- Modifier le contenu d'un fichier avec des outils puissants : *sed*, *awk*
- Réaliser rapidement la même action sur plusieurs fichiers
- Ecrire de simple scripts bash



Rappel Commandes de Base





Previously

- pwd** Affiche le chemin (où je suis)
- ls -alrt** Liste le contenu d'un répertoire
- cd** Change de répertoire



Previously

pwd Affiche le chemin (où je suis)
ls -alrt Liste le contenu d'un répertoire
cd Change de répertoire

mkdir Crée un répertoire
rmdir Supprime un répertoire vide
rm Supprime un fichier
rm -r Supprime répertoire & fichiers
cp source cible Copie/renomme
mv Déplace un fichier/répertoire



Previously

pwd Affiche le chemin (où je suis)
ls -alrt Liste le contenu d'un répertoire
cd Change de répertoire

mkdir Crée un répertoire
rmdir Supprime un répertoire vide
rm Supprime un fichier
rm -r Supprime répertoire & fichiers
cp source cible Copie/renomme
mv Déplace un fichier/répertoire

cat Affiche fichier (court)
less Affiche fichier (long)
head/tail Affiche début/fin fichier
wc -l Compte nombre de lignes



Previously

pwd Affiche le chemin (où je suis)
ls -alrt Liste le contenu d'un répertoire
cd Change de répertoire

mkdir Crée un répertoire
rmdir Supprime un répertoire vide
rm Supprime un fichier
rm -r Supprime répertoire & fichiers
cp source cible Copie/renomme
mv Déplace un fichier/répertoire

(z)cat Affiche fichier (court)
less Affiche fichier (long)
head/tail Affiche début/fin fichier
wc -l Compte nombre de lignes

(z)grep -icv rechercher un motif
cut -d -f Extrait colonnes d'un fichier
sort -t -kgr Trie une colonne d'un fichier
uniq Garder les valeurs uniques



Previously

pwd Affiche le chemin (où je suis)
ls -alrt Liste le contenu d'un répertoire
cd Change de répertoire

mkdir Crée un répertoire
rmdir Supprime un répertoire vide
rm Supprime un fichier
rm -r Supprime répertoire & fichiers
cp source cible Copie/renomme
mv Déplace un fichier/répertoire

chmod Change les droits
chown Change le propriétaire
chgrp Change le groupe

(z)cat Affiche fichier (court)
less Affiche fichier (long)
head/tail Affiche début/fin fichier
wc -l Compte nombre de lignes

(z)grep -icv rechercher un motif
cut -d -f Extrait colonnes d'un fichier
sort -t -kngr Trie une colonne d'un fichier



Previously

pwd Affiche le chemin (où je suis)
ls -alrt Liste le contenu d'un répertoire
cd Change de répertoire

mkdir Crée un répertoire
rmdir Supprime un répertoire vide
rm Supprime un fichier
rm -r Supprime répertoire & fichiers
cp source cible Copie/renomme
mv Déplace un fichier/répertoire

chmod Change les droits
chown Change le propriétaire
chgrp Change le groupe

find rechercher un fichier

(z)cat Affiche fichier (court)
less Affiche fichier (long)
head/tail Affiche début/fin fichier
wc -l Compte nombre de lignes

(z)grep -icv rechercher un motif
cut -d -f Extrait colonnes d'un fichier
sort -t -kng Trie une colonne d'un fichier

history **zcat, zgrep**
tar / gzip Compresser, Décompresser
df -h **du -sh**
wget **ln -s**



Previously

Caractères joker

- * N'importe quel caractère
- [sb] Caractère de l'ensemble



Previously

Caractères joker

- * N'importe quel caractère
- [sb] Caractère de l'ensemble

Redirection Entrées/sorties

- > >> vers un fichier
- | vers une commande



Previously

Caractères joker

* N'importe quel caractère
[sb] Caractère de l'ensemble

Redirection Entrées/sorties

> >> vers un fichier
| vers une commande

Interagir avec les processus

<Ctrl> + C Arrêter le processus en cours sous le terminal



Previously

Caractères joker

* N'importe quel caractère
[sb] Caractère de l'ensemble

Redirection Entrées/sorties

> >> vers un fichier
| vers une commande

Interagir avec les processus

<Ctrl> + C Arrêter le processus en cours sous le terminal

Tab completion

<Tab> Complète automatiquement le nom d'un fichier/ répertoire qui est en cours de saisie (choix unique)

<Tab><Tab> Affiche la liste des différentes possibilités si le choix n'est pas unique



Previously

Interagir avec l'historique de commandes

Flèche bas/haut

Afficher la commande précédente/suivante

Presser plusieurs fois pour naviguer dans l'historique

<Ctrl> + R

Afficher la dernière commande qui contient les caractères saisis.

Presser les touches et commencer à taper la commande recherchée



Previously

Interagir avec l'historique de commandes

Flèche bas/haut

Afficher la commande précédente/suivante

Presser plusieurs fois pour naviguer dans l'historique

<Ctrl> + R

Afficher la dernière commande qui contient les caractères saisis.

Presser les touches et commencer à taper la commande recherchée

Nomenclature fichiers

- Linux = sensible à la casse
- PAS d'espaces, accents et caractères spéciaux `& ~ # " ' { ([| ` \ ^ @)] } $ * % ! / ; , ?`
- Suffixe des noms de fichiers (.txt, .fasta, .fa, .fq etc.) optionnel

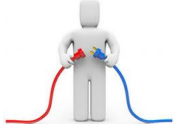


Environnement de travail

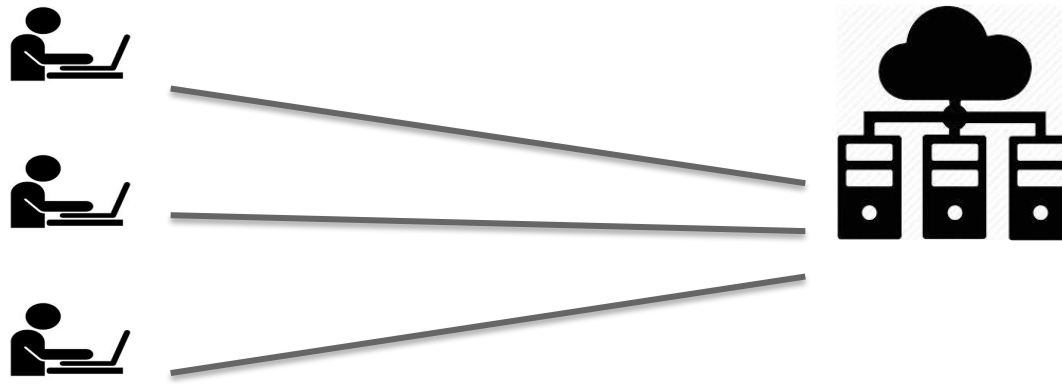
Comment travailler sur le serveur ?



Comment travailler sur le serveur ?



En se connectant sur un serveur linux distant de son ordinateur via le *protocole ssh*



HPC South Green

- itrop (IRD)

bioinfo-inter.ird.fr



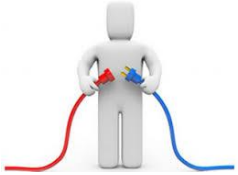
Environnement de travail

Comment transférer un fichier de son PC sur le serveur ?

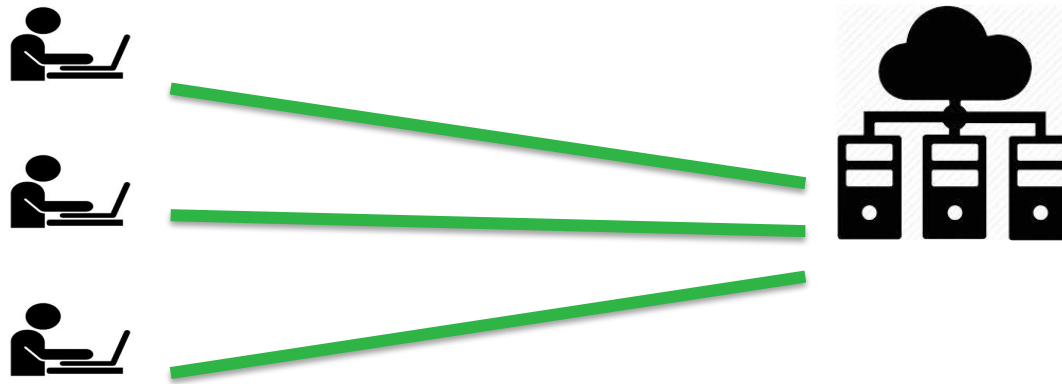
Comment éditer un fichier à distance ?



Copier un fichier de son PC sur le serveur ?



- En se connectant sur un serveur linux distant de son ordinateur via le *protocole sftp*



HPC South Green

- itrop (IRD)

bioinfo-nas.ird.fr



Practice

mobaXterm
terminal, ssh

qrsh, cd, mkdir

1

Go to [Practice 1](#) & [Practice 2](#) on our github



Process monitoring

commande w, ps, kill, top



Comment suivre l'activité sur un serveur ?

w

affiche les utilisateurs et les processus associés

```
[tranchant@master0 ~]$ w
 16:27:57 up 129 days,  5:28, 27 users,  load average: 0,20, 0,25, 0,23
USER      TTY      FROM          LOGIN@      IDLE        JCPU        PCPU        WHAT
klein     pts/5    10.21.129.115  lun.17     1:38m      10.57s     9.93s     qrsh -pe ompi 8
escobar   pts/7    10.23.128.31  14:37      46:05      0.22s     0.09s     ssh node20
daron     pts/8    10.21.141.158  mer.12     1:17m      3:43      10.21s    -bash
tranchan  pts/9    ngo34-1-78-210-1 09:16     31:01      1.69s     1.55s     qrsh -pe ompi 12
```

Nom
utilisateur

Connecté
depuis

Temps cumulé
par tous les ps

Temps cumulé
par le ps actif

ps en cours



Comment suivre l'activité sur un serveur ?

ps

liste les processus en train de tourner

ps -uax

affiche la liste de tous les processus associés à chaque utilisateur

```
[tranchant@node10 ~]$ ps aux | head -4
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
trancha+ 1272  0.0  0.0 116768 3376 pts/2    Ss   09:52   0:00 -bash
trancha+ 3753  0.0  0.0 139512 1680 pts/2    R+   10:34   0:00 ps au
mariac   26118 197  9.1 4598024 4514192 pts/0  RN1+ 07:34 356:07 sniffles ...
```

Etat du processus

R running
S sleeping
T Stopped
Z Zombie



Comment suivre l'activité sur un serveur ?

top

liste les processus en train de tourner

```
top - 16:44:51 up 156 days, 23:10, 1 user, load average: 10,37, 9,80, 9,71
Tasks: 200 total, 3 running, 197 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0,0 us, 0,1 sy, 88,5 ni, 11,5 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
KiB Mem : 65774384 total, 42442784 free, 1907228 used, 21424372 buff/cache
KiB Swap: 8388604 total, 5512296 free, 2876308 used. 62871460 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
18905	daron	30	10	916508	307308	976	R	960,8	0,5	51:38.57	admixture
3446	daron	30	10	1130556	937640	2584	R	100,0	1,4	308:00.92	treemix
19307	trancha+	20	0	146164	2124	1424	R	0,3	0,0	0:00.02	top
22389	root	20	0	0	0	0	S	0,3	0,0	0:00.17	kworker/10:2

c → Afficher la commande complète en exécution

V → Afficher en mode arborescence

M,P → Trier les ps par %mem et %cpu

l → Afficher l'activité CPU (une ligne/CPU)

u → Faire une recherche sur un utilisateur en particulier

i → Ne pas afficher les tâches inactives (idle)

q → **pour quitter**



Comment supprimer un processus ?

kill -9 PID

tuer un processus

```
[tranchant@master0 ~]$ ps aux | grep "tranchant"
tranchant  20999  0.0  0.0 116748  3532 pts/1      Ss+  13:24   0:00 -bash
tranchant  21669  0.0  0.0 176384 22752 pts/1      R    13:33   0:00 perl
toggleGenerator.pl -d /data3/projects/riceAnnot/TOG5681/Illumina/

[tranchant@master0 ~]$kill -9 21669
```




Lancer plusieurs commandes
simultanément

Lancer plusieurs commandes en une ligne

; cmd2 exécutée une fois la cmd1 finie `cmd1 ; cmd2`

&& cmd2 exécutée uniquement si cmd1 correctement finie `cmd1 && cmd2`

```
wget linux.tar.gz && tar -zxvf linux.tar.gz
```



Comment lancer plusieurs ps en même temps ?

Lancer un processus en “arrière plan”

- &** Lancer un processus en arrière plan `cmd1 &`
- jobs** Connaître les processus qui tournent en arrière-plan `jobs`
- fg** Récupérer un processus au premier plan `fg <job_number>`
- bg** Envoyer un processus en arrière plan `bg <job_number>`
- nohup** “Détacher” le processus de la console. Fonctionne même quand la console est fermée, si deconnexion `nohup cmd1 &`
- Ctrl + Z** Stopper un processus



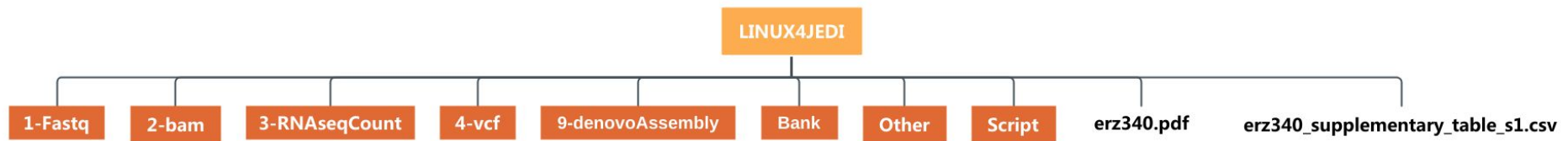
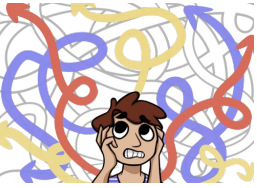
Practice

&&

3

Go to [Practice 3](#) on our github


kezako ces données ?



RESEARCH PAPER



A set of *AP2*-like genes is associated with inflorescence branching and architecture in domesticated rice

Thomas W. R. Harrop¹, Otho Mantegazza², Ai My Luong², Kevin Béthune², Mathias Lorieux³, Stefan Jouannic² and Hélène Adam^{2*}, 

<https://academic.oup.com/jxb/article/70/20/5617/5538968>

Panicle branching diversity and the 2 processes of rice domestication

Thomas Harrop, Otho Mantegazza, Ai My Luong, Mathias Lorieux, Kevin Bethune, Stefan Jouannic, **Hélène Adam**

Wild species

Cultivated species

Asia



O. rufipogon

10 000 ya
→



O. sativa indica and *japonica*

Africa

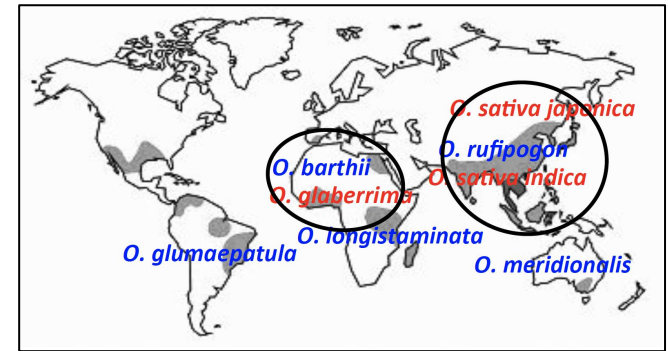


O. barthii

3000 ya
→

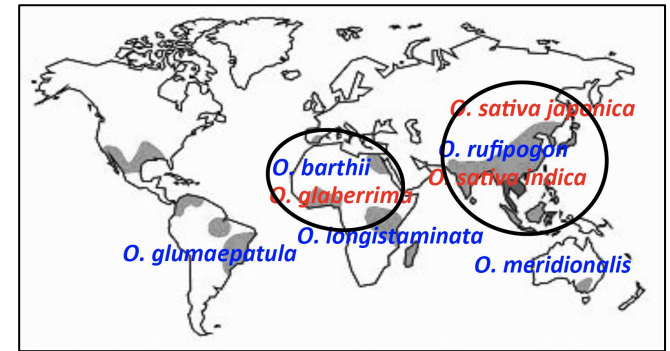
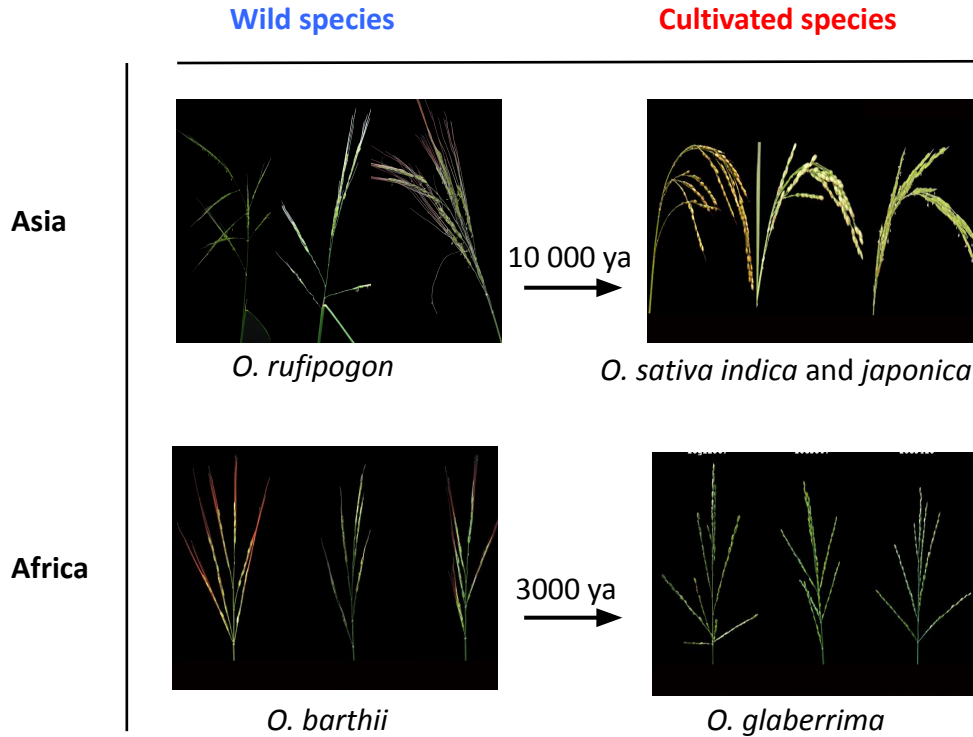


O. glaberrima



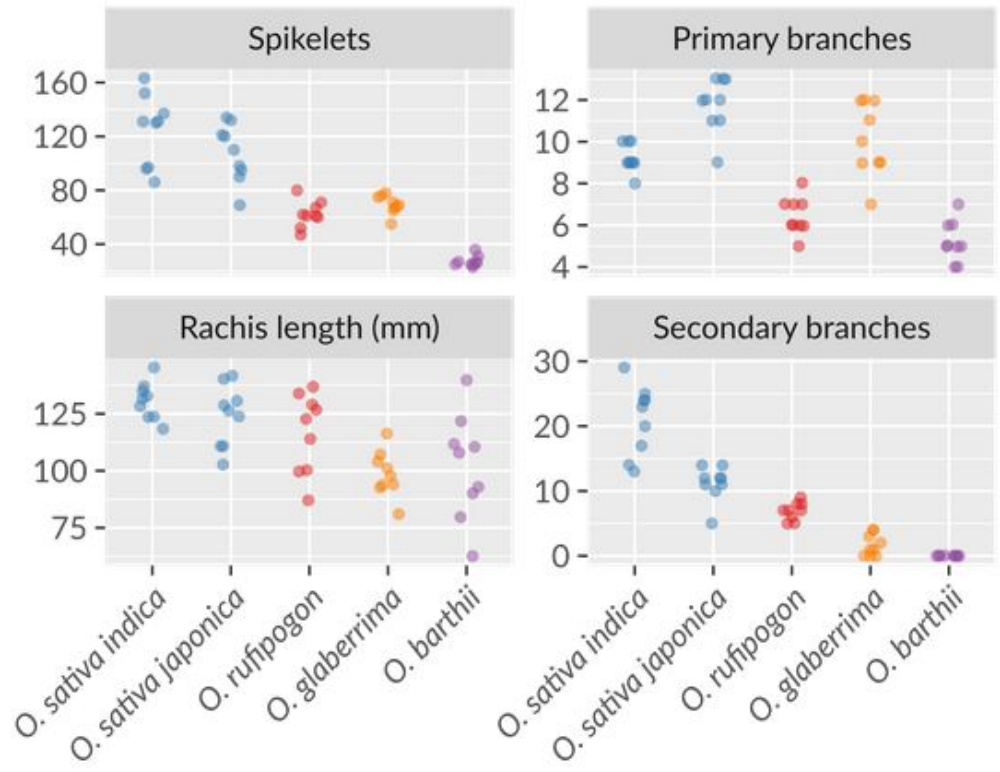
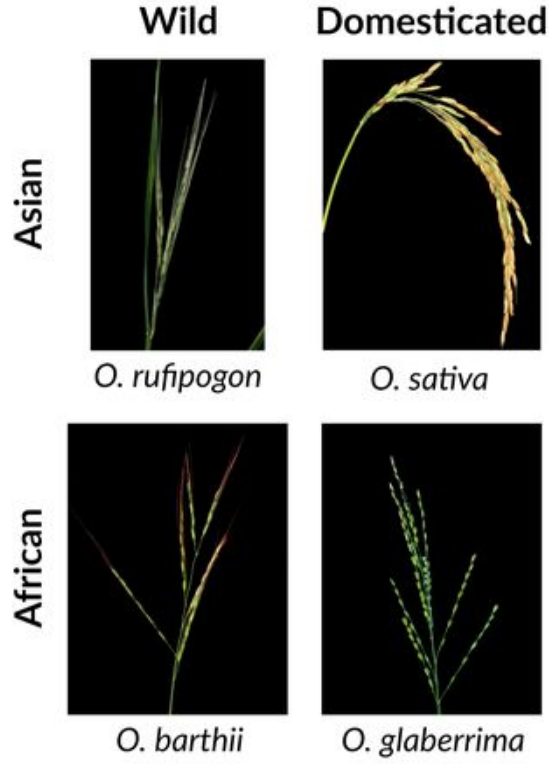
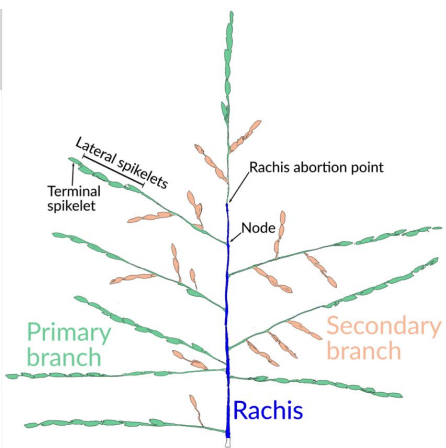
Panicle branching diversity and the 2 processes of rice domestication

Thomas Harrop, Otho Mantegazza, Ai My Luong, Mathias Lorieux, Kevin Bethune, Stefan Jouannic, **Hélène Adam**



- What are the molecular mechanisms related to panicle branching complexity ?
- In which way they explain the diversity of panicle branching observed between these 4 species?

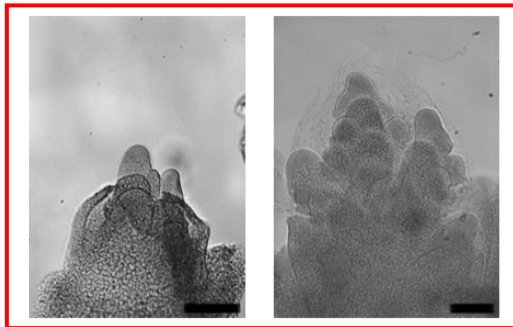
Panicle architecture diversity



Several approaches

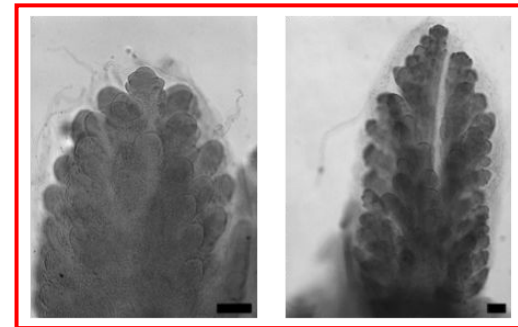
- Panicle morphological traits related to panicle diversity - 90 African and Asian rice accessions
- **Molecular mechanisms related to panicle branching diversity?**

Whole transcriptome RNA sequencing
of **indeterminate** vs **determinate** stages of young inflorescences in the 4 species



primary and higher order branches
initiation and formation (**IM**)

VS



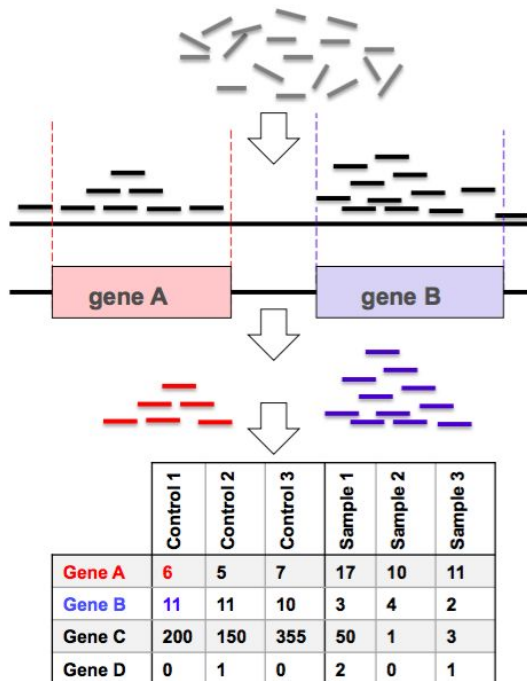
Spikelet/floret
differentiation (**DM**)

Several approaches

- Panicle morphological traits related to panicle diversity - 90 African and Asian rice accessions
- **Molecular mechanisms related to panicle branching diversity?**

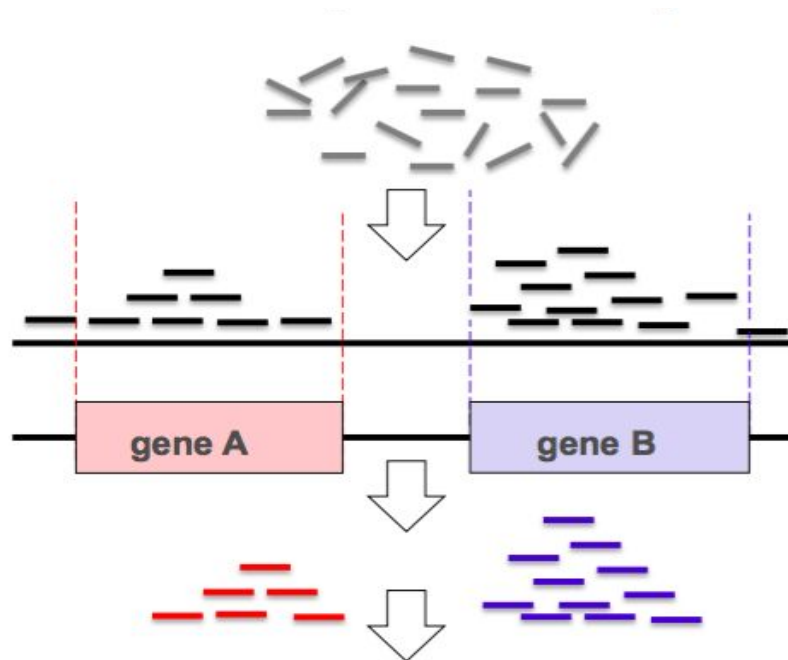
Whole transcriptome RNA sequencing
of **indeterminate** vs **determinate** stages of young inflorescences in the 4 species

RNA-seq data analysis:



- Mapping vs genome reference (*O. sativa japonica*)
- Exploring/ PCA analysis
- DESeq analysis
- Correlation with phenotype

RNA-seq data analysis : typical steps



	Control 1	Control 2	Control 3	Sample 1	Sample 2	Sample 3
Gene A	6	5	7	17	10	11
Gene B	11	11	10	3	4	2
Gene C	200	150	355	50	1	3
Gene D	0	1	0	2	0	1

STEP:	TOOLS:	FILE:
Quality control	FastQC	FASTQ
Pre-processing	Trimmo-matic	FASTQ
Alignment	TopHat	BAM
Quality control	RSeQC	
Quantitation	HTSeq	Read count file (TSV)
Combine count files to table	Define NGS experiment	Read count table (TSV)
Quality control	PCA, clustering	
Differential expression analysis	DESeq2, edgeR	Gene lists (TSV)

csc



Practice

w
ps
kill
top

4

Go to [Practice 4](#) on our github



What is a fastq file ?

1 séquence = 4 lignes

FASTQ file sample:

```
@SRR6407486.1 1 length=100
CCTCGTCTACAGCGACAACGTCCAGACCCGCGAACGGGTGATGCGGGCCCTGGGCAAACGGTTGCACCCGGATCTGCCCGATTGACCTACGTCGAAGTG
+SRR6407486.1 1 length=100
BBBBBFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF<FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7FFFF<FF
```

@SRR6407486.1 1 length=100

CCTCGTCTACAGCGACAAC ... GATTTGACCTACGTCGAAGTG

+SRR6407486.1 1 length=100

BBBBBFFFFFFFFFFFFFFFFFFFFFFFF ... FBFFFFFFFFFFFFFFFF7FFFF<FF

Sequence name

DNA sequence

Quality line break

Quality scores

Base: T
Quality: 7

Quality scores as ASCII characters:

!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJK

Q:	0	5	15	30	40
P _{error} :	1.0	0.32	0.032	0.001	0.0001

$$Q = -10 \log_{10} P_{\text{error}}$$



Illumina paire-end

==> T_1.fq <==

@H4:C399DACXX:7:1101:1551:33084/1

CTATAACTAAGTAAAAACAGCAGAATATGTGCTTAACGACATCTAAGTTAAGATTACATCAAAC
ACAACATGATTATTTGCACAAATGTAATTACCATGAGC

+

@@BFFFFFFGHHFFHEGDHIIIFIJGJGIIJJJJJEHGGGICEGD<DGIHII@FDEGCBHIIJGJJGIE
HEFEEHFDEFFDEEEEFDFEDDDCCAC

@H4:C399DACXX:7:1101:1598:2675/1

AAATATATAAATTTATAGAGTATAGAAATTTGTTGTATGAGACTTTAATATTTATGAATTTGTAA
TGCGAGACTTTTATGGAAATTTCAAGGGCATGGAAG

+

@@CFFFFFFH8FFHIDHIFIC<EHIHH>?FHIECEHBHHAEG?EGGEGHICGHIGIIE:F?DHFEHIIIG
GFECF7@GHGBCH>EHFHB>C@DACECBCCC

@H4:C399DACXX:7:1101:1627:23379/1

AAATTTCTAGCTTTTTCGTTACATCATTTTAAATTTCAACAAAACCTTCAATTTTGACGTGAACTA
AACATTCCAGAATGATCAGCTGCTGGGCAAACCGT

==> T_2.fq <==

@H4:C399DACXX:7:1101:1551:33084/2

GAAAGACATCAACAAAAACATTTTCTTGTCAGTGAGACAGAATTTGATCCCAAGATCGGTGC
TTGATGCCTTGACAGTACAACAATATGCAAATCTTT

+

@CCFFDFDFHHHGIJGJJJJGIGIIGGGGGIHIJJJJGII9FGGIFHGHIIJJIIJGIEIJGDIHGFFHG
DDFFDCCEED?CCDDDD@ACDDDD

@H4:C399DACXX:7:1101:1598:2675/2

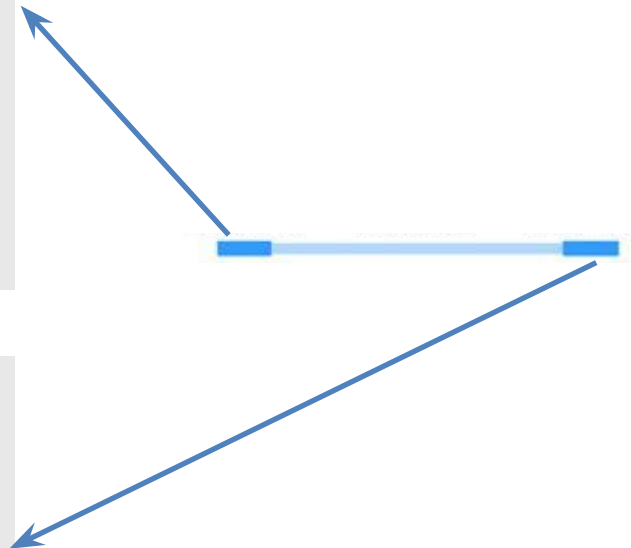
TATATAAAATTAGCATTATATGAAAGTACTTCAAAATTTGAATCTAGTGATATAACATGCATAA
CACTTAGTATAGATATAGTTAGTATGACTATTAGTAA

+

@@@FFDFBF?FDDHGI@B<CIGIJ9EC>EHJHGIGCFHJIGCHIIJFECFHIDDHIIIGGGIIDEJGIHII
JIIGEC@FGGG=CGHJEIC>C>CEEFE@CE

@H4:C399DACXX:7:1101:1627:23379/2

GTTCACTAAACTGATACAGTGCAGTGCAGTTAAATACTACTATATTAATACGACACCACGA
TGATTCCAGCCGACCCGTGAACCAGAAATTCAGAATCG





Practice

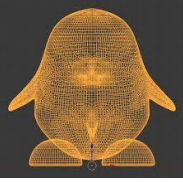
w
ps
kill
top

4

Go to [Practice 4](#) on our github



Expression Régulière (ER)



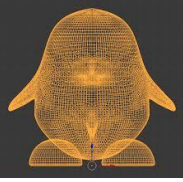
Commande grep

grep

pour rechercher un motif dans un fichier

```
[tranchant@node10 Bank]$ grep "gene" all.gff3 | head -3
Chr1 MSU_osalr7      gene 2903 10817      .      +      .
ID=LOC_Os01g01010;Name=LOC_Os01g01010;Note=TBC%20domain%20containing%20protein%2C%20expressed
Chr1 MSU_osalr7      gene 11218      12435      .      +      .
ID=LOC_Os01g01019;Name=LOC_Os01g01019;Note=expressed%20protein
Chr1 MSU_osalr7      gene 12648      15915      .      +      .
ID=LOC_Os01g01030;Name=LOC_Os01g01030;Note=monocopper%20oxidase%2C%20putative%2C%20expressed

[tranchant@node10 Bank]$ grep "gene" all.gff3 | tail -3
ChrSy      MSU_osalr7      mRNA 589676      589999      .      +      .
ID=ChrSy.fgenes.h.mRNA.89;Parent=ChrSy.fgenes.h.gene.89;Name=ChrSy.fgenes.h.mRNA.89
ChrSy      MSU_osalr7      CDS 589676      589999      11.35      +      0
ID=ChrSy.fgenes.h.CDS.327;Parent=ChrSy.fgenes.h.mRNA.89;score=11.35
ChrSy      MSU_osalr7      exon 589676      589999      11.35      +      .
ID=ChrSy.fgenes.h.exon.327;Parent=ChrSy.fgenes.h.mRNA.89;score=11.35
```



Commande grep

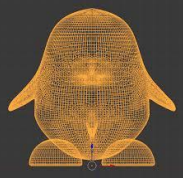
grep

pour rechercher un motif dans un fichier

```
[tranchant@node10 Bank]$ grep "gene" all.gff3 | head -3
Chr1 MSU_osa1r7      gene 2903 10817      .      +      .
ID=LOC_Os01g01010;Name=LOC_Os01g01010;Note=TBC%20domain%20containing%20protein%2C%20expressed
Chr1 MSU_osa1r7      gene 11218      12435      .      +      .
ID=LOC_Os01g01019;Name=LOC_Os01g01019;Note=expressed%20protein
Chr1 MSU_osa1r7      gene 12648      15915      .      +      .
ID=LOC_Os01g01030;Name=LOC_Os01g01030;Note=monocopper%20oxidase%2C%20putative%2C%20expressed

[tranchant@node10 Bank]$ grep "gene" all.gff3 | tail -3
ChrSy      MSU_osa1r7      mRNA 589676      589999      .      +      .
ID=ChrSy.fgenes.h.mRNA.89;Parent=ChrSy.fgenes.h.gene.89;Name=ChrSy.fgenes.h.mRNA.89
ChrSy      MSU_osa1r7      CDS 589676      589999      11.35      +      0
ID=ChrSy.fgenes.h.CDS.327;Parent=ChrSy.fgenes.h.mRNA.89;score=11.35
ChrSy      MSU_osa1r7      exon 589676      589999      11.35      +      .
ID=ChrSy.fgenes.h.exon.327;Parent=ChrSy.fgenes.h.mRNA.89;score=11.35
```

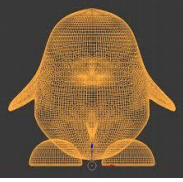
grep -E "gene\s" all.gff3



Expression Régulière

Rechercher un motif (pattern) dans une chaîne de caractère
/MOTIF/

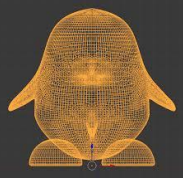
site restriction <i>EcoRI</i>	ATCGCGAATTCAC	/ATCGCGAATTCAC/
site <i>Avall</i>	GGACC ou GGTCC	/GGACC GGTCC/
		/GG[AT]CC/
site restriction <i>BisI</i>	GCNGC	?



Expression Régulière

Rechercher un motif (pattern) dans une chaîne de caractère
/MOTIF/

site restriction <i>EcoRI</i>	ATCGCGAATTCAC	/ATCGCGAATTCAC/
site <i>Avall</i>	GGACC ou GGTCC	/GGACC GGTCC/
		/GG[AT]CC/
site restriction <i>BisI</i>	GCNGC	/GC[ACGT]GC/



Expression Régulière

Rechercher un motif (pattern) dans une chaîne de caractère
/MOTIF/

site restriction <i>EcoRI</i>	ATCGCGAATTCAC	/ATCGCGAATTCAC/
site <i>Avall</i>	GGACC ou GGTCC	/GGACC GGTCC/
		/GG[AT]CC/
site restriction <i>BisI</i>	GCNGC	/GC[ACGT]GC/

^ATG[ATGC]{3,1000}A{5,10}\$



Expression Régulière

Rechercher un motif (pattern) dans une chaîne de caractère
/MOTIF/

Motif avec une base T présente 3 à n fois	GATC GATTC ...	/A{3,}/
Motif avec une base T présente 0 à 7 fois	GAC GATC GATTC ...	/T{,7}/
Motif présent en début de chaîne	AAAGGG	^AAA
Motif présent en fin de chaîne	AAAGGG	GGG\$

Expression Régulière

Rechercher un motif (pattern) dans une chaîne de caractère
/MOTIF/

Motif avec une base T présente 3 à n fois	GATC GATTC ...	<code>/A{3,}/</code>
Motif avec une base T présente 0 à 7 fois	GAC GATC GATTC ...	<code>/T{,7}/</code>
Motif présent en début de chaîne	AAAGGG	<code>^AAA</code>
Motif présent en fin de chaîne	AAAGGG	<code>GGG\$</code>

`^ATG[ATGC]{30,1000}A{5,10}$`



Expression régulière ou rationnelle

Motif qui décrit un ensemble de chaînes de caractères possibles permettant de faire des sélections



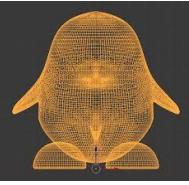
Expression régulière ou rationnelle

Chaîne de caractères qui décrit un ensemble de chaînes de caractères possibles permettant de faire des sélections

Communes aux ERs basiques et étendues

<code>^</code>	début de ligne	<code>^LOC1</code>
<code>\$</code>	fin de ligne	<code>LOC1\$</code>
<code>.</code>	n'importe quel caractère	<code>^L.C1</code>
<code>*</code>	0 à n fois	<code>ATCA*T</code>
<code>[...]</code>	plage de caractères permis	<code>[ATGC]</code>
<code>[^...]</code>	plage de caractères interdits	<code>[^ATGC]</code>

Expression régulière ou rationnelle



- [0-9]** N'importe quel chiffre
- [a-z]** N'importe quelle lettre en minuscule
- [^A-Z]** N'importe quel caractère excepté une lettre en majuscule
- [a-zA-Z]** N'importe quelle lettre en minuscule ou majuscule

- \s** espace
- \t** tabulation



Practice

5

Go to [Practice 5](#) on our github



Practice

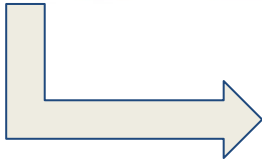
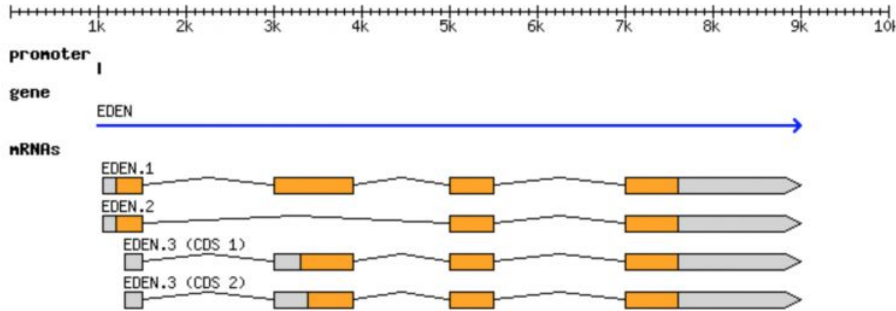
grep -E

6

Go to [Practice 6](#) on our github



What is a gff file ?

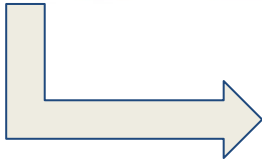
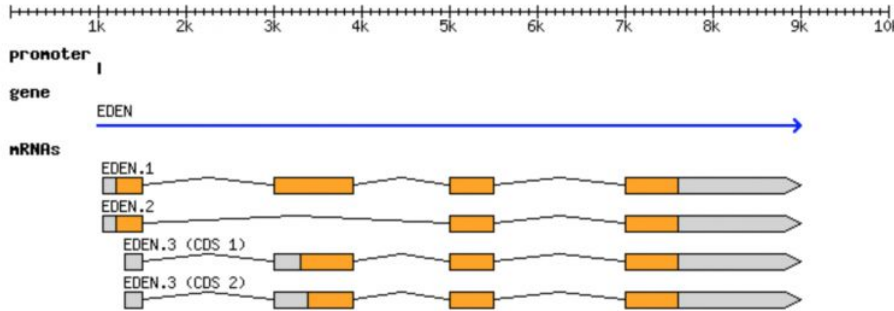


same information can be represented
in GFF3 format:

```
0 ##gff-version 3.2.1
1 ##sequence-region ctg123 1 1497228
2 ctg123 . gene 1000 9000 . + . ID=gene00001;Name=EDEN
3 ctg123 . TF_binding_site 1000 1012 . + . ID=tfbs00001;Parent=gene00001
4 ctg123 . mRNA 1050 9000 . + . ID=mRNA00001;Parent=gene00001;Name=EDEN.1
5 ctg123 . mRNA 1050 9000 . + . ID=mRNA00002;Parent=gene00001;Name=EDEN.2
6 ctg123 . mRNA 1300 9000 . + . ID=mRNA00003;Parent=gene00001;Name=EDEN.3
7 ctg123 . exon 1300 1500 . + . ID=exon00001;Parent=mRNA00003
8 ctg123 . exon 1050 1500 . + . ID=exon00002;Parent=mRNA00001,mRNA00002
9 ctg123 . exon 3000 3902 . + . ID=exon00003;Parent=mRNA00001,mRNA00003
10 ctg123 . exon 5000 5500 . + . ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
11 ctg123 . exon 7000 9000 . + . ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003
12 ctg123 . CDS 1201 1500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
13 ctg123 . CDS 3000 3902 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
14 ctg123 . CDS 5000 5500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
15 ctg123 . CDS 7000 7600 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
16 ctg123 . CDS 1201 1500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
17 ctg123 . CDS 5000 5500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
18 ctg123 . CDS 7000 7600 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
19 ctg123 . CDS 3301 3902 . + 0 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
20 ctg123 . CDS 5000 5500 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
21 ctg123 . CDS 7000 7600 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
22 ctg123 . CDS 3391 3902 . + 0 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
23 ctg123 . CDS 5000 5500 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
24 ctg123 . CDS 7000 7600 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
```



What is a gff file ?



same information can be represented
in GFF3 format:

```
0 ##gff-version 3.2.1
1 ##sequence-region ctg123 1 1497228
2 ctg123 . gene 1000 9000 . + . ID=gene00001;Name=EDEN
3 ctg123 . TF_binding_site 1000 1012 . + . ID=tfbs00001;Parent=gene00001
4 ctg123 . mRNA 1050 9000 . + . ID=mRNA00001;Parent=gene00001;Name=EDEN.1
5 ctg123 . mRNA 1050 9000 . + . ID=mRNA00002;Parent=gene00001;Name=EDEN.2
6 ctg123 . mRNA 1300 9000 . + . ID=mRNA00003;Parent=gene00001;Name=EDEN.3
7 ctg123 . exon 1300 1500 . + . ID=exon00001;Parent=mRNA00003
8 ctg123 . exon 1050 1500 . + . ID=exon00002;Parent=mRNA00001,mRNA00002
9 ctg123 . exon 3000 3902 . + . ID=exon00003;Parent=mRNA00001,mRNA00003
10 ctg123 . exon 5000 5500 . + . ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
11 ctg123 . exon 7000 9000 . + . ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003
12 ctg123 . CDS 1201 1500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
13 ctg123 . CDS 3000 3902 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
14 ctg123 . CDS 5000 5500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
15 ctg123 . CDS 7000 7600 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
16 ctg123 . CDS 1201 1500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
17 ctg123 . CDS 5000 5500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
18 ctg123 . CDS 7000 7600 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
19 ctg123 . CDS 3301 3902 . + 0 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
20 ctg123 . CDS 5000 5500 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
21 ctg123 . CDS 7000 7600 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
22 ctg123 . CDS 3391 3902 . + 0 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
23 ctg123 . CDS 5000 5500 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
24 ctg123 . CDS 7000 7600 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
```



Download it into your directory:

http://rice.uga.edu/pub/data/Eukaryotic_Projects/o_sativa/annotation_dbs/pseudomolecules/version_7.0/all.dir/all.gff3

Take a look at it and see what it looks like!



Practice

grep

6

Go to [Practice 6](#) on our github



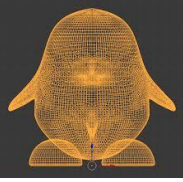
Des commandes pour rechercher et modifier des fichiers

commande sed

Sélection et affichage de lignes dans un fichier

par numero de ligne

```
sed -n 'line p ' inputfile
```



Sélection et affichage de lignes dans un fichier *par numero de ligne*

```
sed -n 'line p' inputfile
```

Affiche la 5ème ligne

```
sed -n '5p' all.gff3
```

```
sed -n '5p' *.fastq
```

```
sed -n -s '5p' *.fastq
```

Affiche la ligne 1 et 8

```
sed -n "1p; 8p" test.txt
```

Affiche la ligne 1 à 8

```
sed -n "1,8 p" test.txt
```

**Affiche à partir de la ligne 1,
toutes les 4 lignes**

```
sed -n '1~4p' ir.fastq
```

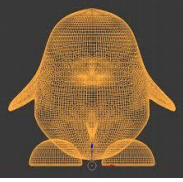


Practice

printing with sed

7

Go to [Practice 7](#) on our github



Suppression de lignes dans un fichier

par numero de ligne

```
sed 'line d' inputfile
```

```
sed "2d; 4d" test.txt           # supprime ligne 2 et 4  
sed "2,4 d" test.txt           # supprime ligne 2 à 4  
sed '2~4d' irigin1_1.fastq
```

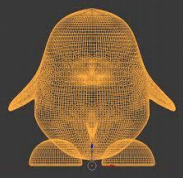


Practice

deleting with sed

8

Go to [Practice 8](#) on our github



Sélection de lignes dans un fichier

par motif

```
sed 'ER' inputfile
```

```
sed '/^#/d' test.sed
```

```
sed -n '/^Bonjour/p; /^Au revoir/p' test.sed
```

```
sed -n '/^Bonjour/,/4.$/p' test.sed
```



Practice

sed using ER

9

Go to [Practice 9](#) on our github



What is a vcf file ?

```
##fileformat=VCFv4.3
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
```

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	NA00001	NA00002	NA00003
20	14370	rs6054257	G	A	29	PASS	NS=3;DP=14;AF=0.5;DB;H2	GT:GQ:DP:HQ	0 0:48:1:51,51	1 0:48:8:51,51	1/1:43:5:...
20	17330	.	T	A	3	q10	NS=3;DP=11;AF=0.017	GT:GQ:DP:HQ	0 0:49:3:58,50	0 1:3:5:65,3	0/0:41:3
20	1110696	rs6040355	A	G,T	67	PASS	NS=2;DP=10;AF=0.333,0.667;AA=T;DB	GT:GQ:DP:HQ	1 2:21:6:23,27	2 1:2:0:18,2	2/2:35:4
20	1230237	.	T	.	47	PASS	NS=3;DP=13;AA=T	GT:GQ:DP:HQ	0 0:54:7:56,60	0 0:48:4:51,51	0/0:61:2
20	1234567	microsat1	GTC	G,GTCT	50	PASS	NS=3;DP=9;AA=G	GT:GQ:DP	0/1:35:4	0/2:17:2	1/1:40:3

Métadonnées

Descripteur des colonnes

Données de l'individu NA0001

Données pour une variation

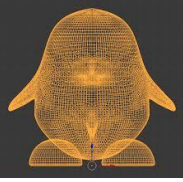


Practice

sed using ER

9

Go to [Practice 9](#) on our github



Substitution/Remplacement dans lignes

Sélection de lignes dans un fichier vérifiant une expression régulière
ET appliquant une modification ou un traitement

```
sed "s/motif recherché/nouveau motif/" file
```

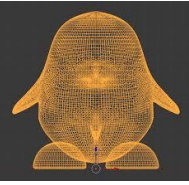
substitution

séparateur

motif recherché

nouveau motif

fichier à parser



Sed : Quelques exemples

Example

```
sed "s/day/night/" file
```

Description

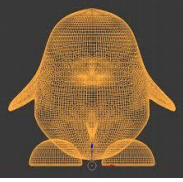
Change la 1ère occurrence de “day”
par “night” **par ligne**

```
sed "s/linux/LINUX/2" file
```

Change la 2ème occurrence de “linux”
par “LINUX” **par ligne**

```
sed "s/[lL]inux/LINUX/g"  
file
```

Change toutes occurrences de “linux”
par “LINUX”



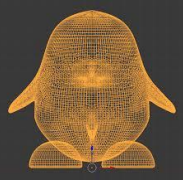
Sélection et Substitution de lignes dans un fichier

par motif

```
sed 's/ / /' inputfile  
sed 'y/éè/ee/' inputfile
```

```
sed -n '2~4s/T/u/p;' irigin1_1.fastq
```

```
sed -n '2~4y/Tt/Uu/p;' irigin1_1.fastq
```



sed : rechercher et modifier une ligne

Selection de lignes dans un fichier vérifiant une expression régulière
ET appliquant une modification ou un traitement

```
sed "s/[0-9][0-9]*/nouveau motif/" file
```

substitution

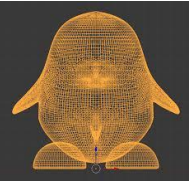
motif recherché

nouveau motif

fichier à parser

Recherche une chaîne de caractères
commençant par un chiffre suivi par 0 ou plusieurs nombres

=> Chaîne de caractère enregistrée dans la variable `\1`



Example

```
sed 's/\([a-z]*\) \t/\1/' abcd.txt
```

abcd

```
sed -E 's/([a-z]*) \t(.*)/\2 \1/' abcd.txt
```

123 abcd



Practice

sed

10

Go to [Practice 10](#) on our github



Des commandes pour rechercher et modifier des fichiers

commande awk



awk

awk: Langage pour manipuler un fichier ligne par ligne

- Nom des auteurs : “Aho, Weinberger, and Kernighan”



awk

awk: Langage pour manipuler un fichier ligne par ligne

- Nom des auteurs : “Aho, Weinberger, and Kernighan”
- Un langage de programmation qui permet facilement de manipuler des fichiers tabulés (blast, sam, vcf) et d’extraire une partie des données
- Un langage utilisé pour rechercher des motifs et pour effectuer des opérations, des actions associées.



awk

awk: Langage pour manipuler un fichier ligne par ligne

Principales caractéristiques d'awk

- Fichier en entrée tabulé
- Comme tout langage de programmation, awk a des variables et peut appliquer des conditions
- awk peut faire des opérations sur les nombres et les chaînes de caractères
- awk peut générer et afficher des données/rapports suite à des manipulations



awk

awk: Langage pour manipuler un fichier ligne par ligne

Syntax : awk [-F] 'program' file

Option	Description
-F	Donne la nature des séparateurs de champs



awk

awk: Langage pour manipuler un fichier ligne par ligne

Syntax : awk [-F] 'program' file

Option	Description
-F	Donne la nature des séparateurs de champs

Variables prédéfinies utilisées par awk

Variable	Description
\$0	ligne entière
NR	Numéro de la ligne lue
NF	Nombre de champs dans la ligne



awk

awk voit le fichier en entrée
comme des enregistrements et des champs

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcaron@videotron.ca

File: contact.txt



awk

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcaron@videotron.ca

File: contact.txt

```
awk '{print $0}' contact.txt
```

```
Helene 56 edu hcyr@sun.com  
jean 32 ri jeanc@inexpress.net  
julie 22 adm juliem@sympatico.ca  
michel 24 inf michel@uqo.ca  
richard 25 inf rcaron@videotron.ca
```

Affiche chaque
ligne



awk

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcaron@videotron.ca

File: contact.txt

```
$awk '{print NR, $1, $2}' contact.txt
```

```
1 Helene 56  
2 jean 32  
3 julie 22  
4 michel 24  
5 richard 25
```

Affiche
le numéro de la ligne lue
Puis le 1^{er} champ
puis le 2^e champ du fichier
tabulé



awk

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcaron@videotron.ca

```
$awk '{print $1,$2};  
END { print NR "lignes lues en tout" }' contact.txt
```

```
Helene 56  
Jean 32  
Julie 22  
Michel 24  
Richard 25  
5 lignes lues en tout
```

Instruction exécutée une fois le fichier lu dans son intégralité



awk

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcaron@videotron.ca

```
$awk '{print $1,$3; somme+= $2}
END { print "Somme des ages égale à ", somme }' contact.txt
```

```
Helene edu
jean ri
julie adm
michel inf
richard inf
Somme des ages égale à 159
```

On ajoute l'âge (\$2) à la variable **somme** à chaque ligne lue

Puis on **affiche la somme** calculée à la fin de la lecture du fichier



awk

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcaron@videotron.ca

```
File: contact.txt  
$awk ' {somme += $2 }  
END { print " Age moyen = ", somme/NR } ' contact.txt
```

Age moyen = 31,8

On ajoute l'âge (\$2) à la variable **somme** à chaque ligne lue

Puis on **affiche la moyenne** une fois le fichier lu



awk

awk: Langage pour manipuler un fichier ligne par ligne

avec une liste d'instructions et **de conditions aussi**

Condition {Instr-1; Instr-2; ...; Instr-n}

```
awk '{if($2 > 24 && $2 < 50) { print "Age de ", $1,  
"compris entre 24 et 50 : egal a ", $2 } }' contact.txt
```

```
Age Helene compris entre 24 et 50 : egal a 56  
Age jean compris entre 24 et 50 : egal a 32  
Age richard compris entre 24 et 50 : egal a 25
```

**Avec 2
conditions**



awk

awk: Langage pour manipuler un fichier ligne par ligne

```
awk '{if($3 == "inf") {print $0}}' contact.txt
```

```
michel 24 inf michel@uqo.ca  
richard 25 inf rcaron@videotron.ca
```

```
$awk '/j/ {print $0}' contact.txt
```

```
jean 32 ri jeanc@inexpress.net  
julie 22 adm juliem@sympatico.ca
```



awk

awk: Langage pour manipuler un fichier ligne par ligne

```
awk '{print $1, $2-10}' contact.txt
```

```
Helene 46  
Jean 12  
Julie 12  
Michel 14  
Richard 15
```

```
awk '{if($2 > 30 && $3 == "ri") {print $0}}' contact.txt
```

```
jean 32 ri jeanc@inexpress.net
```

Ces commandes peuvent être utilisées avec en entrée la sortie standard ou un fichier tabulé (comme .gff, fichier blast m8 , .vcf)



Practice

sed using ER

9

Go to [Practice 9](#) on our github

enleve ligne vide

fastq -> fasta

Manipulating all files with a given extension

```
1. # power chasis
2. for f in directory/*.ext ; do n=`basename $f` fn=${n%.ext}; mycodehere >
   outdir/${fn}.newext ; done
3.
4. # for example quality filter all bam files in a directory
5. for f in bam-uf/*.bam ; do n=`basename $f` fn=${n%.bam}; samtools view -b -q 20 -f 0x002 -F
   0x004 -F 0x008 $f > bam-mq20/${fn}.q20.bam ; done
```

Remove empty lines



Practice

awk

11

Go to [Practice 11](#) on our github



awk - fonctions

Manipulation de chaîne de caractères

<code>length(myText)</code>	longueur de myText
<code>substr(myText, start, length)</code>	Extrait la sous chaîne de la chaîne <i>myText</i> à partir de la position <i>start</i> sur une longueur <i>length</i>
<code>tolower(myText)</code>	Modifie la casse de myText en minuscule
<code>toupper(myText)</code>	Modifie la casse de myText en majuscule
<code>split(myText, array, fieldsep)</code>	decoupe myText <code>split(\$2, monTab, " ") ; print(monTab[2])</code>
<code>gsub(search, replace, var)</code>	<code>gsub(";", "- ", \$3)</code>
<code>sub(ER, replace, var)</code>	



awk - fonctions

Manipulation de nombres

`int(myNb)`

partie entière de myNb

`log(myNb)`

logarithme de myNb

`sqrt(myNb)`

racine carée de myNb



Pour automatiser le lancement de commandes

bash





Boucle for



Exécuter une boucle

for...



```
Instruction1;  
instruction2;  
Instruction3;
```



- To parse a directory
- To run the same instruction on each file of the directory Exécuter les mêmes instructions sur chaque élément de la liste

```
for file in * ;  
do  
    instruction1  
    instruction2  
done
```



Practice

12

Go to [Practice 12](#) on our github



Exécuter un script bash

sh nom_script.sh

```
[tranchant@node10 Bash]$ sh helloWorld.sh
```



Premier script en bash

- Toujours débiter par : `#!/bin/sh`
- Suivis par les instructions, une instruction par ligne
- **Chaque instruction doit se terminer par ;**
- N'hésitez pas à commenter votre script en plaçant un `#` devant votre commentaire



Premier script en bash

- Toujours débiter par : `#!/bin/sh`
- Suivis par les instructions, une instruction par ligne
- **Chaque instruction doit se terminer par ;**
- N'hésitez pas à commenter votre script en plaçant un `#` devant votre commentaire
 - Pour vous et vos collègues pour comprendre le code
 - ignore le texte placé après un `#`
 - Commentaires libres



Premier script en bash

- Pas d'accent
- Premières instructions

`echo 'text';` pour écrire sur la sortie (écran)
`echo -e "text \n";` pour réaliser un saut de ligne



Modifier le script

P1.2

- Sauver le script *helloWorld.sh* sous un nouveau nom (ex : helloWorld-v2.sh)
- Modifier le code de ce nouveau script en affichant d'autres textes avec `\n`
- Exécuter ce nouveau script



Modifier le script

P1.3

- Créer volontairement des erreurs dans votre code en retirant un `;` puis un `#` et un `“`
- Observer les messages d'erreurs



Modifier le script

P1.3

- Créer volontairement des erreurs dans votre code en retirant un `;` puis un `#` et un `“`
- Observer les messages d'erreurs

Une des principales activités du programmeur est de « déboguer »...
Souvent aussi longue qu'écrire le code !
Il faut donc s'entraîner à décoder les messages d'erreurs !





Les variables



Qu'est ce une variable ?

Variable...

```
nom="Hello World";  
echo $nom;
```

« conteneur », « boîte » dans lesquels on peut stocker un objet, une information.

Règles

- Noms de variables uniquement avec des caractères *alpha-numériques* (A-Z, a-z, 0-9) ou *underscore*
- **Sensible à la casse et pas d'espace**



Une variable variable...

Variable...

```
maVar= "Hello World!!!";  
echo $maVar;           # Hello World  
echo ${maVar:6}       # World!!!  
echo ${maVar:0:3};    # Hel  
echo ${maVar:6:3};    # Wor  
echo ${maVar: -2};    # !!
```




Une variable variable...

Variable...

```
file=BCU_AAOSW_3_1_C39R6ACX.bam
echo $file                # BCU_AAOSW_3_1_C39R6ACX.bam
echo ${file:5}           # AOSW_3_1_C39R6ACXX.bam

echo ${file/.bam/.sam}   #BCU_AAOSW_3_1_C39R6ACXX.sam
```



Substitution au sein d'une variable

Variable...

```
maVar= "Hello World!!!";  
echo $maVar;           # Hello World!!!  
echo ${maVar/o/}      # HeLL World!!!  
echo ${maVar//o/};    # HeLL WrLd!!!
```



Practice

13

Go to [Practice 13](#) on our github



Arguments d'un script



Condition avec des nombres

- Transmettre au script des valeurs saisies dans la ligne de commande : arguments, paramètres
- Affectées aux variables réservées 1, 2, ... et appelées \$1, \$2, ...

```
sh testNum.sh 25
```

```
#!/bin/bash
myNum=$1;
if [[ $myNum = 10 ]]
then
    echo "Egal a 10";
elif [[ $myNum -le 10 ]]
then
    echo "Inferieur ou egal a 10";
else
    echo "Superieur a 10";
fi
```



Les conditions



Condition avec une chaîne de caractère

Variable...

```
#!/bin/bash

myText="Hello world ! ";

if [[ $maText = "Hello" ]]; then
    echo "Very Nice";
else
    echo "No nice";
fi
```

```
sh script.sh
```



Condition avec une chaîne de caractère

Variable...

```
#!/bin/bash

myText="Hello world ! ";

if [[ $maText =~ "Hello" ]]; then
    echo "Very Nice";
else
    echo "No nice";
fi
```

```
sh script.sh
```

```
# Very Nice
```




Condition avec des nombres

```
#!/bin/bash

myNum=18;

if [[ $myNum = 10 ]]
then
    echo "Egal a 10";
elif [[ $myNum -le 10 ]]
then
    echo "Inferieur ou egal a 10";
elif [[ $myNum -gt 10 ]]
then
    echo "Superieur a 10";
else echo "C'est quoi ce bins?";
fi
```



Opérateur de comparaison

Nombres

$\$a$ -eq $\$b$

$\$a$ égal à $\$b$

$\$a$ -ne $\$b$

$\$a$ différent de $\$b$

$\$a$ -lt $\$b$

$\$a$ inférieur à $\$b$

$\$a$ -gt $\$b$

$\$a$ supérieur à $\$b$

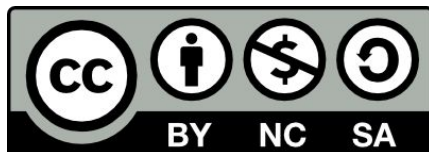
$\$a$ -le $\$b$

$\$a$ inférieur ou égal à $\$b$

$\$a$ -ge $\$b$

$\$a$ supérieur ou égal à $\$b$

- **Christine Tranchant-Dubreuil**
- **Gautier Sarah**
- **Valérie Noël**
- **Ndomassi Tando**
- **Frédéric Mahé**
- **François Sabot**



Si vous utilisez les ressources du plateau i-Trop.

Merci de nous citer avec:

“ The authors acknowledge the ISO 9001 certified IRD i-Trop HPC (South Green Platform) at IRD montpellier for providing HPC resources that have contributed to the research results reported within this paper.

URL: <https://bioinfo.ird.fr/>- <http://www.southgreen.fr>”

- Pensez à inclure un budget ressources de calcul dans vos réponses à projets
- Besoin en disques dur, renouvellement de machines etc...
- Devis disponibles
- Contactez bioinfo@ird.fr : aide, définition de besoins, devis...

En informatique,
la pensée magique ne fonctionne pas !

Il faut pratiquer ... et ... *restez calme !*
à vous de jouer !



Copyright © Randy Glasbergen. www.glasbergen.com



Le matériel pédagogique utilisé pour ces enseignements est mis à disposition selon les termes de la licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage dans les Mêmes Conditions (BY-NC-SA) 4.0 International: <http://creativecommons.org/licenses/by-nc-sa/4.0/>

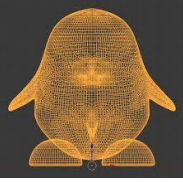
Merci pour votre attention !



Le matériel pédagogique utilisé pour ces enseignements est mis à disposition selon les termes de la licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage dans les Mêmes Conditions (BY-NC-SA) 4.0 International:

<http://creativecommons.org/licenses/by-nc-sa/4.0/>

File format conversion



Commande dos2unix, mac2unix

- Diversité de formats, même entre différents OS

<code>\n</code>	UNIX
<code>\r</code>	Mac
<code>\r\n</code>	Windows

dos2unix, mac2unix

*convertir un fichier texte au format UNIX
pour qu'il soit lu correctement*



Fusionner des fichiers : la commande join



Commande join

join fichier1 fichier2

```
root@bigbluewin:~$ cat fichier1
1 Bash
2 Python
3 Perl
4 Java
5 C++
root@bigbluewin:~$ cat fichier2
1 sympa
2 cool
3 no comment
4 pfff
5 ouille
```

```
root@bigbluewin:~$ join fichier1 fichier2
1 Bash sympa
2 Python cool
3 Perl no comment
4 Java pfff
5 C++ ouille
```



Commande join

Fusionner en précisant les colonnes communes :

```
join -1 2 -2 1 fichier1 fichier3
```

Préciser les colonnes à afficher :

```
join -1 2 -2 1 fichier1 fichier3 -o 2.1,2.2
```

Les fichiers doivent être triés au préalable