

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

INF4500

---

## Examen intra

---

*Par :*

Guillaume Lahaie  
LAHG04077707

*Remis à :*

Abdoulaye Baniré Diallo

*Date de remise :*

Le 9 décembre 2013



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Produisez une analyse sommaire de ces contigs en présentant la distribution des tailles et taux de GC</b>	<b>4</b>
<b>3</b>	<b>Identifiez les annotations Genbank de ces contigs et présentez les dans une table contenant les colonnes : contigs, numéros Accession, description, uniref id</b>	<b>7</b>
a	Numéros d'accessions et description . . . . .	7
b	Uniref . . . . .	7
c	Construction du tableau . . . . .	8
<b>4</b>	<b>Identifiez les contigs qui coderaient pour des protéines et donnez une table de ceux-ci, contenant contig, numéros d'Accession, Uniref, séquences protéiques</b>	<b>16</b>
<b>5</b>	<b>Question 4</b>	<b>18</b>
a	Alignement multiple . . . . .	18
b	Facilité et rapidité des programmes . . . . .	18
c	Qualité des alignements . . . . .	18
d	Meilleur alignement . . . . .	18
<b>6</b>	<b>Annexes</b>	<b>19</b>
1	Tableau de la taille et du taux de GC des contigs . . . . .	19
2	Script biopython de calcul des fréquences nucléotidiques . . . . .	22
3	Script biopython Pour choisir 5 contigs au hasard à partir du résultat de CAP3, et d'effectuer un blast sur ces contigs . . . . .	22
4	Script Biopython pour obtenir les fichiers d'accession des 10 premiers résultats du blast pour un contig donné. . . . .	23
5	Log de l'exécution de clustalw2. . . . .	24
6	Log du travail de RepeatMasker sur le fichier foxp4_ortho.fa. . . . .	25

7	Log de l'exécution de Mavid sur foxp4_ortho.fa. . . . .	25
8	Script biopython pour identifier la composition du vecteur pANNE . . . . .	27
9	Temps d'exécution des alignements multiples . . . . .	28
10	Script Biopython pour obtenir les fichiers d'accension des 10 premiers résultats du blast pour un contig donné. . . . .	28
11	Fichiers genbank utilisés pour ce rapport . . . . .	30
12	Fichiers de gène de NCBI utilisé pour ce rapport . . . . .	32

# 1 Introduction

Le but de ce travail est d'annoter des contigs du génome du blé. Nous n'avons pas d'information concernant la provenance de ces contigs, ou même l'espèce exacte de provenance. Afin de pouvoir fournir une information pertinente, j'ai tout d'abord recherché ce qui est connu concernant le génome du blé.

J'ai tout d'abord cherché à connaître l'état d'avancement des travaux de séquençage du blé. Pour ce faire, j'ai consulté la base de données des génomes de NCBI [1]. On y apprend des informations de base sur le génome du blé. On y apprend que le génome du blé a une taille de 16000 Mb distribué en 21 chromosomes. De plus, les chromosomes ont une forme allohexaploid composée de trois sous-génomes. La nature hexaploid de son génome a ralenti les efforts de séquençage.

Une première référence de génome du blé a été créée avec l'espèce *Triticum urartu* [2]. Ce génome est toutefois celui d'un progéniteur du *Triticum aestivum*, il peut être utile pour aider à améliorer le génome du blé.

On peut obtenir une information plus complète concernant l'avancement du séquençage du *Triticum aestivum* sur le site du International Wheat Genome Sequencing Consortium. On y retrouve deux projets parallèles : en premier lieu, un projet de survey sequencing, afin de produire un contenu de gène potentiel et un ordre de gène virtuel [3]. Un autre projet en cours est de produire une séquence de référence pour le génome du *Triticum aestivum* [4]. Ce projet semble être à ses débuts, car il semble être en cours d'obtention de financement.

D'autres bases de données offrent de l'information à propos du génome du blé, par exemple CerealsDB [5], ayant un génome de travail du blé. Il y a aussi beaucoup d'autres projets, considérant la place importante occupée par le blé dans l'agriculture moderne.

Basé sur ces informations, j'ai décidé de concentrer mes recherches pour l'annotation des contigs fournis sur les données déjà connues du génome du blé. Je vais donc seulement garder les résultats de Blast provenant du *Triticum aestivum*. Bien sûr, il s'agit ici d'une première étape de recherche, il serait ensuite possible d'élargir la recherche pour identifier des zones fonctionnelles possibles des contigs, ce qui ne sera pas fait dans ce travail.

## 2 Produisez une analyse sommaire de ces contigs en présentant la distribution des tailles et taux de GC

Afin de compiler et de représenter la taille et le taux de GC des contigs produits par CAP3, j'ai écrit un script python (question1.py) permettant d'extraire les informations du fichier seq.data.cap.contigs. Le fichier contient 346 contigs.

Le script produit deux types de graphiques, à l'aide de gnuplot. Le premier type est un histogramme, un pour la taille des contigs, et un pour le taux de GC des contigs. On peut alors remarquer la distribution de ces valeurs. Voici les deux histogrammes :

J'ai ensuite produit deux graphiques permettant de visualiser différemment ces résultats. On peut y retrouver la moyenne de taille, la moyenne de taux de GC, ainsi que les contigs se situant en haut ou en bas de cette moyenne. On peut aussi voir les valeurs exactes dans le tableau en annexe 1

La taille moyenne des 346 contigs est de 109 nucléotides, avec un taux de GC moyen de 42,96%. Ce taux semble indiquer une prépondérance de région non-codante dans les contigs, car généralement les séquences codantes ont un taux de GC supérieur aux séquences non-codantes [6].

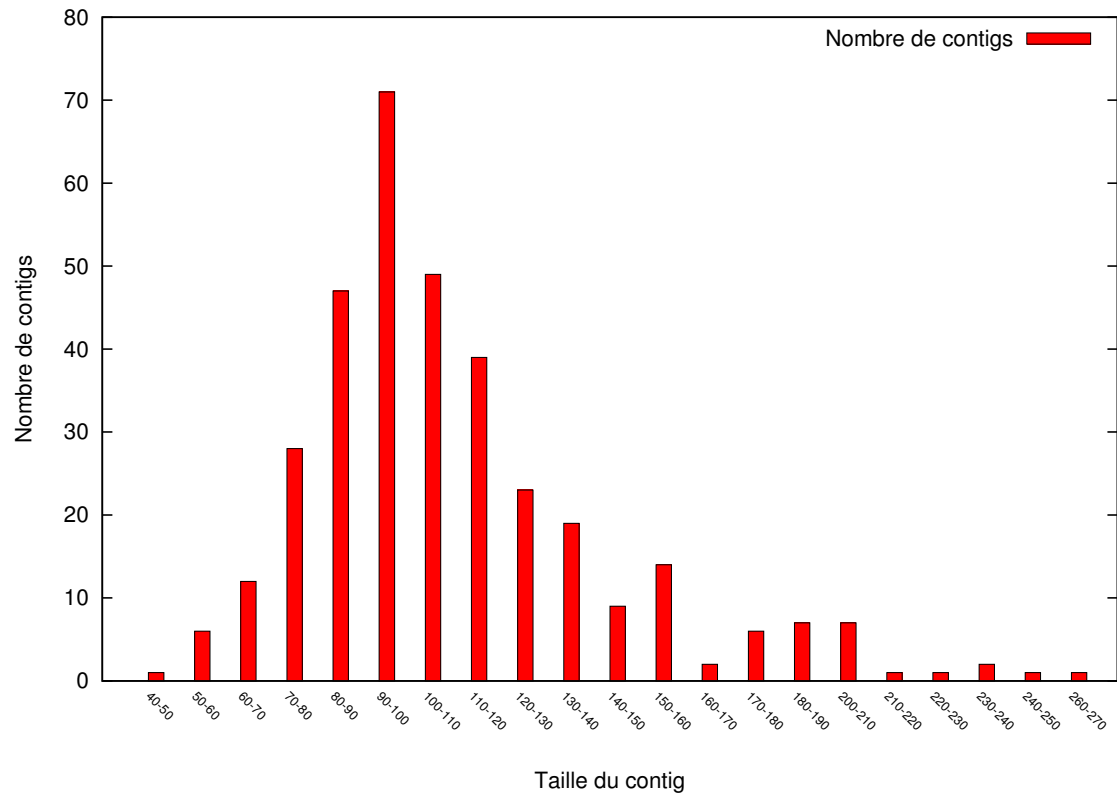


FIGURE 1 – Histogramme de la taille des contigs

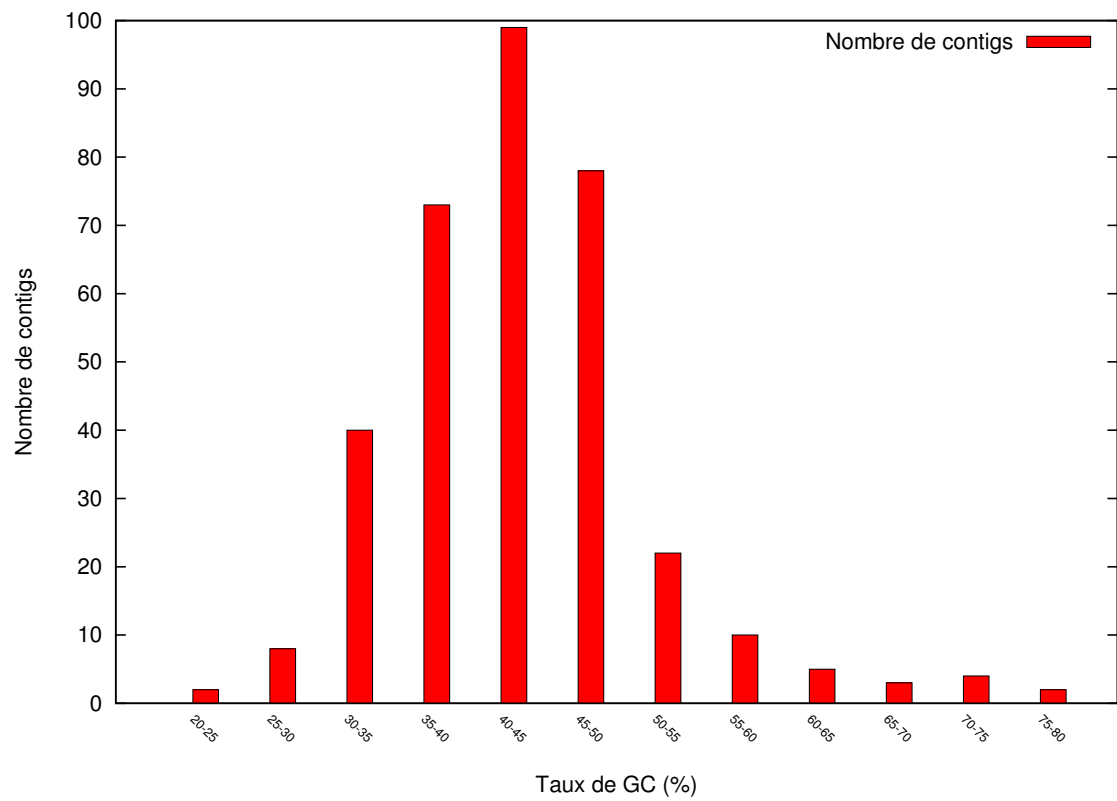


FIGURE 2 – Histogramme du taux de GC des contigs

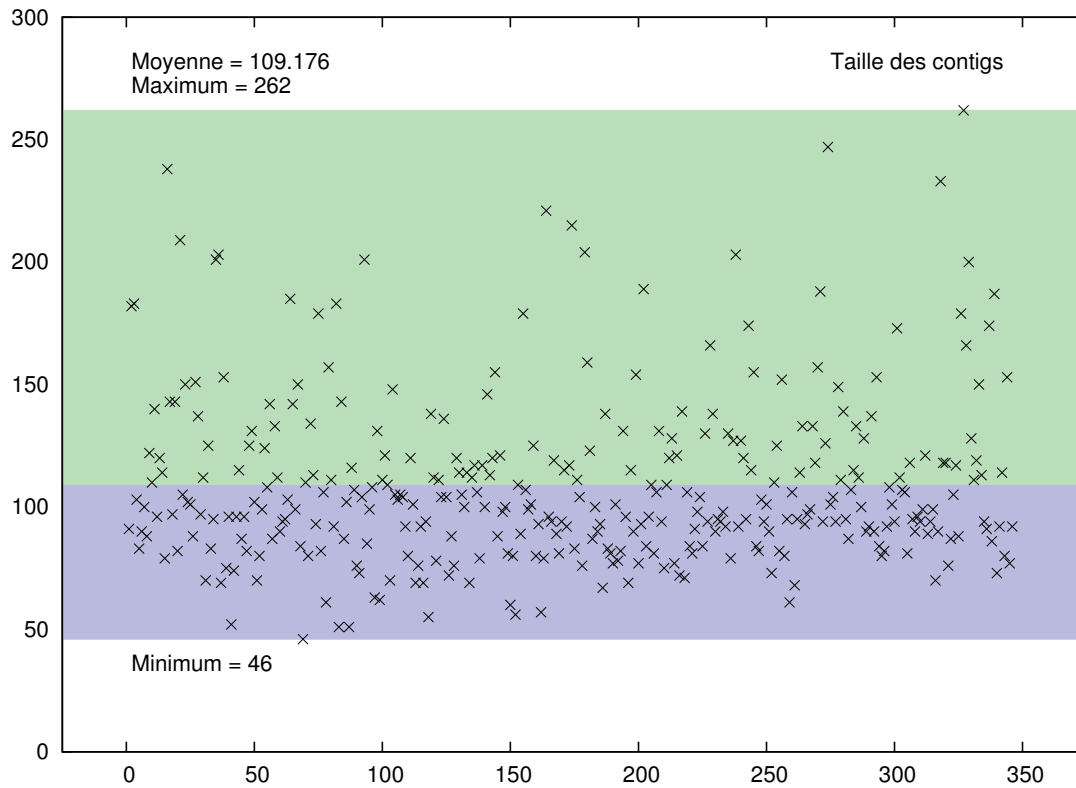


FIGURE 3 – Nuage de points de la taille des contigs

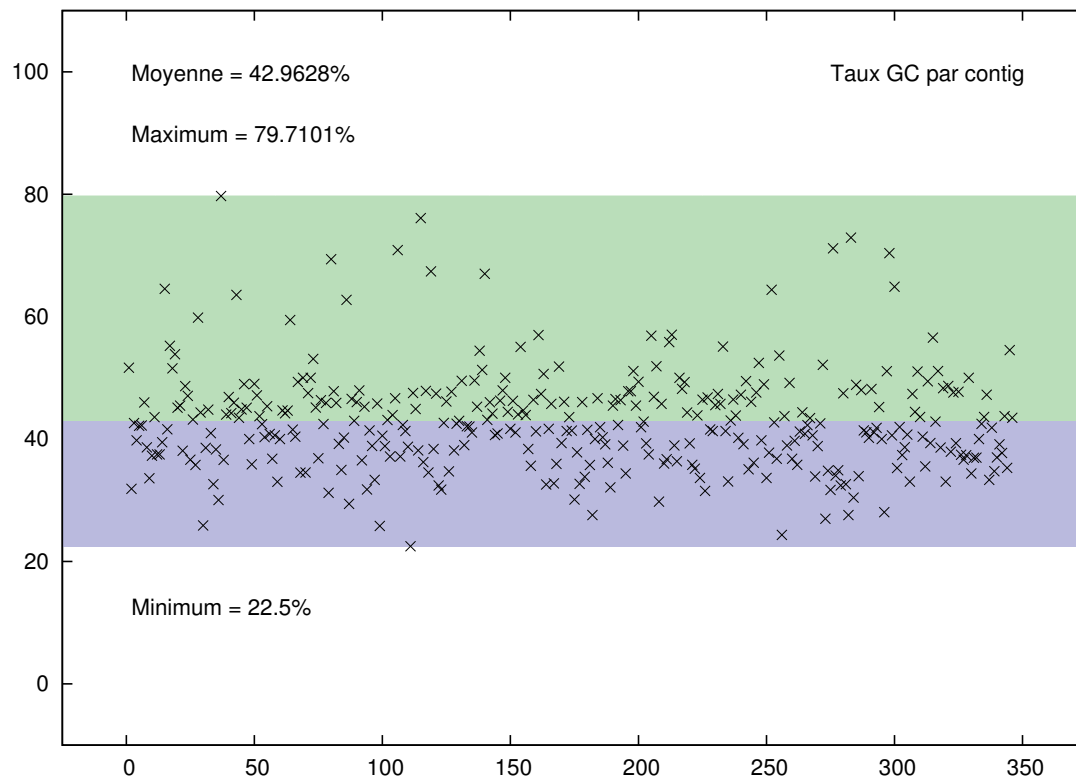


FIGURE 4 – Nuage de points du taux de GC des contigs



### **3 Identifiez les annotations Genbank de ces contigs et présentez les dans une table contenant les colonnes : contigs, numéros Accession, description, uniref id**

#### **a Numéros d'accessions et description**

Pour trouver les annotations Genbank des contigs, j'ai tout d'abord effectué un blast de chaque contig sur la base de données nr/nt de NCBI [8]. J'ai utilisé le script biopython question2.py pour effectuer tous les blasts, et enregistrer les résultats.

En examinant les résultats de façon sommaire, on remarque une très grande différence entre la qualité des résultats. Certains ont des E-value très haute, alors que certains ont des valeurs indiquant un résultat de haute qualité. On peut s'attendre à cela, considérant la grande variabilité des contigs.

Pour traiter les contigs selon leur taille, je calcule la valeur médiane des E-value pour les contigs plus petits que la taille moyenne. Je fais le même exercice pour les contigs plus grands que la moyenne. Pour le moment, je m'intéresse au meilleur résultat obtenu seulement pour la médiane.

Comme mentionné en introduction, comme cette analyse s'intéresse seulement au contigs ayant des résultats pour le Triticum, je ne considère pas dans mes résultats les valeurs de blast pour des espèces différentes du blé. Je prends donc, dans les résultats de blast, le premier correspondant à un match avec le blé.

J'ai enregistré les résultats dans les fichiers evaluate\_lower.txt et evaluate\_higher.txt, à l'aide du script q2\_meanEvaluate.py. On peut remarquer que la grande majorité des résultats obtenus ont des E-values de bonne qualité, avec un ordre de grandeur permettant d'avoir une grande confiance dans le hit. Basé sur ces données, je garderai donc tous les résultats, peu importe la taille du contig, ayant une E-value plus petite que 0.01.

Afin d'obtenir les données de numéro d'accession, j'ai modifié le script précédent pour créer un fichier associant le numéro du contig avec le hit gardé (pour le moment, je garde seulement le premier hit de blé du résultat), avec le numéro d'accession et la description du hit. Ces données sont gardées seulement si le hit correspond aux exigences de E-value et de description de hit.

#### **b Uniref**

Pour obtenir un Uniref [9] pour les contigs retenus, j'ai ensuite utilisé le module bioservices de python permettant de se connecter au service idmapping de uniprot, pour trouver les identifiants uniref des contigs conservés.

Des 223 contigs restant, 113 ont obtenu des résultats de mapping. Avant de sortir les résultats, j'ai vérifié le format des données obtenues par ce mapping. Pour certains contigs, un seul résultat est obtenu, alors que pour certains, on obtient plusieurs mappings différents. Les fichiers XML ne comprennent aucune information concernant le meilleur résultat, toutefois le service REST utilisé pour le mapping demande de trier les résultats selon le meilleur score.

Afin de vérifier le résultat, j'ai tenté de blaster un des contigs directement sur la base de données Uniref100, sur le site <http://www.uniprot.org>. Le résultat a été surprenant. J'ai utilisé le contig 2 comme essai, et le blastx sur Uniref100 n'a retourné aucun hit. Afin de confirmer ce résultat, j'ai effectué le même blastx en utilisant le service d'EBI et en blastant sur toutes les bases de données de protéines de uniprot. J'ai obtenu le même résultat.

Je crois que ce résultat est dû au mécanisme de mapping. Comme nous avons pu le constater à la questions 1, la plupart des contigs donnés ont une longueur moyenne de 109 nucléotides. Toutefois, le numéro d'accension donnée pour effectuer le id mapping peut correspondre à une très longue séquence. C'est le cas du numéro d'accension pour le contig 2, il s'agit en fait d'un chromosome complet du blé, ce qui explique les nombreux résultats du mapping.

J'ai donc décidé de procéder différemment pour obtenir les identifiants uniref correspondant spécifiquement au contig. J'ai effectué un blastx directement sur la base de données uniref100 pour chaque contig. Pour ce faire, j'ai utilisé le script `q2_ebi.py`. Encore une fois, j'ai utilisé le module bioservices de python pour cette tâche.

J'ai ensuite vérifié les résultats des blasts pour les contigs retenus précédemment. J'ai appliqué le même filtre : je vérifie tout d'abord si la description du résultat est pour le blé, et ensuite si le E-value correspond à une valeur acceptable. Je prends la même valeur que pour les numéros d'accension genbank : 0.01.

Après avoir appliqué ce filtre, il me reste 70 contigs pour lesquels j'ai un numéro d'accension genbank et un numéro uniref.

## c Construction du tableau

Comme j'utilise latex pour la rédaction de mon rapport, j'ai écrit un script afin de combiner les informations de mes différents scripts dans un tableau que je peux insérer directement dans mon fichier latex (`q2_tableau.py`).

Le tableau créé rassemble les informations des 223 contigs ayant un résultat de blast pour le blé. Les informations pour les autres contigs n'est pas présenté. Pour ces contigs, j'indique aussi le uniref trouvé, si une valeur correspondante existe.

Contig	Accession	Description	Uniref - EBI
2	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
3	EF109232	Triticum aestivum strain CRB-INRA-CFD-13471 malate dehydrogenase (Mdh4B) gene, partial cds.	
4	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
5	AK331959	Triticum aestivum cDNA, clone : WT002.M17, cultivar : Chinese Spring.	UniRef100_M7YGL9
6	AK332278	Triticum aestivum cDNA, clone : WT003.J14, cultivar : Chinese Spring.	
7	AK335464	Triticum aestivum cDNA, clone : WT012.P12, cultivar : Chinese Spring.	
10	JQ240472	Triticum urartu clones BAC 70G09, BAC 169L13, and BAC 78P09, complete sequence.	
14	AK332744	Triticum aestivum cDNA, clone : WT004.M05, cultivar : Chinese Spring.	

Contig	Accession	Description	Uniref - EBI
16	AK332362	Triticum aestivum cDNA, clone : WT003.M19, cultivar : Chinese Spring.	
18	U73217	Triticum aestivum cold acclimation protein WCOR615 (Wcor615) mRNA, complete cds.	
21	DQ286562	Triticum aestivum putative lipid transfer protein mRNA, complete cds.	
22	KC816724	Triticum urartu cultivar G1812 clone BAC 288D18 chromosome 3AL, complete sequence.	
23	AK335482	Triticum aestivum cDNA, clone : WT013.A03, cultivar : Chinese Spring.	
24	AK330641	Triticum aestivum cDNA, clone : SET4.P05, cultivar : Chinese Spring.	UniRef100.M8ABV0
25	AK331680	Triticum aestivum cDNA, clone : SET1.K05, cultivar : Chinese Spring.	
26	AK332086	Triticum aestivum cDNA, clone : WT003.B19, cultivar : Chinese Spring.	UniRef100.M7YAN9
27	EU660894	Triticum turgidum subsp. durum clone BAC 1053F12+1054I5 cytosolic acetyl-CoA carboxylase (Acc-2) and putative amino acid permease genes, complete cds.	
29	BT008986	Triticum aestivum clone wdk2c.pk008.b17 :fis, full insert mRNA sequence.	
30	HQ596874	Triticum aestivum voucher AP212 trnH-psbA intergenic spacer, partial sequence; chloroplast.	
31	HQ391280	Triticum aestivum clone UCDDTA01731 genomic sequence.	UniRef100.M8A3H2
33	HE996560	Triticum aestivum cv. Arina SNP, chromosome 3B, clone Taes.arina.ctg_58725.	UniRef100.D9CJA9
34	HQ391329	Triticum aestivum clone UCDDTA01780 genomic sequence.	
35	EU159424	Triticum turgidum haplotype B DNA repair protein Rad50 gene, complete cds.	
36	DQ251490	Triticum aestivum cultivar Chinese Spring powdery mildew resistance protein PM3CS (Pm3) gene, Pm3-CS allele, complete cds.	
37	KC290909	Triticum aestivum clone pTa-s309 FISH-positive repetitive sequence.	UniRef100.T1L6W5
39	AJ318783	Triticum sp. partial mRNA for replication factor C, large subunit (rfc-1 gene).	UniRef100.Q8L6A5
40	AK330233	Triticum aestivum cDNA, clone : SET3.P11, cultivar : Chinese Spring.	UniRef100.T1N886
41	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
44	AJ784900	Triticum aestivum mRNA for type 1 non-specific lipid transfer protein precursor (ltp9.4 gene).	
46	AK330669	Triticum aestivum cDNA, clone : SET1.G08, cultivar : Chinese Spring.	
47	AK331428	Triticum aestivum cDNA, clone : WT007.H14, cultivar : Chinese Spring.	UniRef100.M8AEN7
48	HE996767	Triticum aestivum cv. Arina SNP, chromosome 3B, clone Taes.arina.ctg_66371.	UniRef100.M7Z3W8
49	AK332525	Triticum aestivum cDNA, clone : SET1.N11, cultivar : Chinese Spring.	
51	AK331813	Triticum aestivum cDNA, clone : WT002.G19, cultivar : Chinese Spring.	
53	HE996642	Triticum aestivum cv. Arina SNP, chromosome 3B, clone Taes.arina.ctg_60579.	UniRef100.T1M429
56	HQ390245	Triticum turgidum clone UCDDTA00696 genomic sequence.	
57	FJ345689	Triticum aestivum MITE Tourist-3 MITE Islay Tourist, complete sequence.	
58	JF758499	Triticum aestivum clone BAC 425P7, complete sequence.	UniRef100.M7Z3R4

Contig	Accession	Description	Uniref - EBI
59	FN645450	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0011b.	
60	FN645450	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0011b.	
61	AK333585	Triticum aestivum cDNA, clone : WT006.N11, cultivar : Chinese Spring.	UniRef100.M8A091
63	FN564428	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0091b.	
67	AK332970	Triticum aestivum cDNA, clone : WT005.F05, cultivar : Chinese Spring.	UniRef100.M7YP29
70	AK332566	Triticum aestivum cDNA, clone : WT004.E21, cultivar : Chinese Spring.	
71	AK334580	Triticum aestivum cDNA, clone : SET1.C02, cultivar : Chinese Spring.	
73	EU660896	Triticum urartu clone BAC 059G16 plastid acetyl-CoA carboxylase (Acc-1) gene, complete cds; nuclear gene for plastid product.	UniRef100.M7ZVV5
74	KC912694	Triticum aestivum chloroplast, complete genome.	
75	AB238931	Triticum monococcum TmABI1 gene for protein phosphatase 2C, complete cds.	UniRef100.M7YVM1
76	BT009089	Triticum aestivum clone wkm2c.pk0002.a3 :fis, full insert mRNA sequence.	
78	AK335897	Triticum aestivum cDNA, clone : SET2.L19, cultivar : Chinese Spring.	UniRef100.M7ZH67
80	AK330275	Triticum aestivum cDNA, clone : SET4.A24, cultivar : Chinese Spring.	
81	HE996341	Triticum aestivum cv. Arina SNP, chromosome 3B, clone Taes.arina_ctg.16989.	
84	JF758499	Triticum aestivum clone BAC 425P7, complete sequence.	
86	AK335765	Triticum aestivum cDNA, clone : WT013.L14, cultivar : Chinese Spring.	UniRef100.M7YZ42
88	FN564432	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0616b.	
91	AM932681	Triticum aestivum 3B chromosome, clone BAC TA3B63B13.	
92	U76215	Triticum aestivum NBS-LRR type protein pseudogene, complete sequence.	
94	HE996549	Triticum aestivum cv. Arina SNP, chromosome 3B, clone Taes.arina_ctg.58561.	
95	AY487917	Triticum aestivum Mla-like protein mRNA, partial cds.	UniRef100.Q6RW52
96	KC912694	Triticum aestivum chloroplast, complete genome.	UniRef100.T1MEW5
97	KC912694	Triticum aestivum chloroplast, complete genome.	
98	AK333621	Triticum aestivum cDNA, clone : WT006.O21, cultivar : Chinese Spring.	
99	KC912694	Triticum aestivum chloroplast, complete genome.	
100	AK333932	Triticum aestivum cDNA, clone : WT008.N17, cultivar : Chinese Spring.	UniRef100.T1N9G3
101	AK331581	Triticum aestivum cDNA, clone : SET1.J20, cultivar : Chinese Spring.	
102	KC912694	Triticum aestivum chloroplast, complete genome.	
104	HQ391318	Triticum aestivum clone UCDA01769 genomic sequence.	UniRef100.T1MZT0
105	HQ391224	Triticum aestivum clone UCDA01675 genomic sequence.	UniRef100.M7ZMV6
108	DQ286562	Triticum aestivum putative lipid transfer protein mRNA, complete cds.	
109	AK335062	Triticum aestivum cDNA, clone : WT011.P12, cultivar : Chinese Spring.	UniRef100.M7ZAY8
110	KC152455	Triticum aestivum clone BAC321B14 MATE1B gene, complete cds.	
112	AK332529	Triticum aestivum cDNA, clone : WT004.D08, cultivar : Chinese Spring.	UniRef100.M7ZA56
115	AY049041	Triticum aestivum 28S ribosomal RNA gene, partial sequence.	UniRef100.T1L6Y4

Contig	Accession	Description	Uniref - EBI
117	AK334519	Triticum aestivum cDNA, clone : WT010.C18, cultivar : Chinese Spring.	
119	AK333035	Triticum aestivum cDNA, clone : WT005.H19, cultivar : Chinese Spring.	UniRef100.Q9FT38
120	CT009735	Triticum aestivum.	
121	HE996280	Triticum aestivum cv. Arina SNP, chromosome 3B, clone Taes.arina.ctg_14118.	UniRef100.M8AZM6
122	EU626553	Triticum urartu clone BAC 261N5, complete sequence.	
123	HQ391007	Triticum aestivum clone UCDA01458 genomic sequence.	
125	KC912694	Triticum aestivum chloroplast, complete genome.	UniRef100.T1LKW9
127	AK330423	Triticum aestivum cDNA, clone : SET4.G18, cultivar : Chinese Spring.	UniRef100.M8AIQ8
131	HF541875	Triticum aestivum chromosome 3B specific BAC library, BAC clone TaaCsp3BFhA_0147D05.	UniRef100.M7ZGW4
132	CT009735	Triticum aestivum.	UniRef100.T1LKM3
135	HQ391329	Triticum aestivum clone UCDA01780 genomic sequence.	
136	KC912694	Triticum aestivum chloroplast, complete genome.	UniRef100.M7ZC27
137	AK332255	Triticum aestivum cDNA, clone : WT003.I14, cultivar : Chinese Spring.	UniRef100.M7YN64
144	AK332897	Triticum aestivum cDNA, clone : WT005.C09, cultivar : Chinese Spring.	
146	EF219468	Triticum aestivum translationally-controlled tumor protein mRNA, complete cds.	UniRef100.M7YF70
148	AJ001117	Triticum aestivum mRNA for sucrose synthase type I.	
150	AK330745	Triticum aestivum cDNA, clone : SET5.D06, cultivar : Chinese Spring.	UniRef100.M7YEA6
151	AK335219	Triticum aestivum cDNA, clone : WT012.F19, cultivar : Chinese Spring.	UniRef100.M7ZDI5
152	AK335725	Triticum aestivum cDNA, clone : SET2.K04, cultivar : Chinese Spring.	UniRef100.M7ZVF6
153	BT009622	Triticum aestivum clone wre1n.pk0137.c12 :fis, full insert mRNA sequence.	
154	HE774675	Triticum aestivum chromosome arm 3DS-specific BAC library, contig ctg1484.	
157	FN564428	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0091b.	
158	EU626553	Triticum urartu clone BAC 261N5, complete sequence.	
159	DQ862833	Triticum monococcum S-adenosylhomocysteine hydrolase mRNA, partial cds.	
161	AK330639	Triticum aestivum cDNA, clone : SET4.P03, cultivar : Chinese Spring.	
162	KC912694	Triticum aestivum chloroplast, complete genome.	UniRef100.T1LKW9
164	DQ432014	Triticum aestivum vacuolar proton-ATPase subunit A mRNA, complete cds.	
165	KC912694	Triticum aestivum chloroplast, complete genome.	UniRef100.T1LKW9
166	EU835980	Triticum aestivum clone BAC 502E09, complete sequence.	
167	FN564426	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0005b.	
168	JF439307	Triticum aestivum cultivar Yang Mai 158 serine/threonine protein kinase Stpk-D (Stpk-D) gene, complete cds.	
171	KC175605	Triticum aestivum calcium-dependent protein kinase 3-like 1 mRNA, partial cds.	UniRef100.M1NQF6
173	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
174	AK333177	Triticum aestivum cDNA, clone : WT005.N11, cultivar : Chinese Spring.	

Contig	Accession	Description	Uniref - EBI
175	AJ132439	Triticum aestivum mRNA for protein encoded by It1.1 gene, partial.	
176	AK331581	Triticum aestivum cDNA, clone : SET1_J20, cultivar : Chinese Spring.	
177	HQ390713	Triticum aestivum clone UCDA01164 genomic sequence.	
179	EU660895	Triticum aestivum clone BAC 1825J10 cytosolic acetyl-CoA carboxylase (Acc-2) and putative amino acid permeases genes, complete cds.	
180	FN564430	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0464b.	
181	FJ427399	Triticum turgidum clone BAC 738D05 chromosome 4B, partial sequence.	
182	DQ537336	Triticum aestivum clones BAC 1289J04 ; BAC 1001P20, complete sequence.	
184	AK330153	Triticum aestivum cDNA, clone : SET3_M02, cultivar : Chinese Spring.	
185	JX040632	Triticum turgidum subsp. durum x Secale cereale glutamine synthetase I (GSI) mRNA, complete cds.	UniRef100.M7ZP85
187	AY951945	Triticum monococcum TmBAC 60J11 FR-Am2 locus, genomic sequence.	
193	AK332664	Triticum aestivum cDNA, clone : WT004_I22, cultivar : Chinese Spring.	
194	AK330641	Triticum aestivum cDNA, clone : SET4_P05, cultivar : Chinese Spring.	UniRef100.T1MAM1
195	HE774676	Triticum aestivum chromosome arm 3DS-specific BAC library, contig ctg447.	
197	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
198	AK334078	Triticum aestivum cDNA, clone : WT009_E03, cultivar : Chinese Spring.	
199	AM502900	Triticum aestivum mRNA for MIKC-type MADS-box transcription factor WM30 (WM30 gene).	
201	AK331183	Triticum aestivum cDNA, clone : SET6_K07, cultivar : Chinese Spring.	
203	AK331090	Triticum aestivum cDNA, clone : SET6_A20, cultivar : Chinese Spring.	
206	FN564430	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0464b.	
208	FJ345689	Triticum aestivum MITE Tourist-3 MITE Islay Tourist, complete sequence.	
211	GU817319	Triticum aestivum clone BAC_2383A24 chromosome 3B, complete sequence.	
213	AK333846	Triticum aestivum cDNA, clone : WT008_O20, cultivar : Chinese Spring.	UniRef100.M7YLY4
216	AP013106	Triticum timopheevii mitochondrial DNA, complete sequence.	
217	AK332920	Triticum aestivum cDNA, clone : WT005_D07, cultivar : Chinese Spring.	
221	AK334145	Triticum aestivum cDNA, clone : WT009_O11, cultivar : Chinese Spring.	
222	KC912694	Triticum aestivum chloroplast, complete genome.	UniRef100.T1LKW9
224	GU817319	Triticum aestivum clone BAC_2383A24 chromosome 3B, complete sequence.	
225	FR820619	Triticum turgidum subsp. durum partial mRNA for td3ITN1 protein.	
226	AK334286	Triticum aestivum cDNA, clone : WT009_F09, cultivar : Chinese Spring.	
227	AK332238	Triticum aestivum cDNA, clone : WT003_H22, cultivar : Chinese Spring.	UniRef100.M7ZLU3
228	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	

Contig	Accession	Description	Uniref - EBI
229	HQ391114	Triticum aestivum clone UCDA01565 genomic sequence.	
231	GU211169	Triticum aestivum clone 09d3 gliadin/avenin-like seed protein mRNA, complete cds.	UniRef100.D2KFH0
233	AF532601	Triticum aestivum multidrug resistance associated protein MRP2 mRNA, complete cds.	UniRef100.M7ZK96
235	AF389882	Triticum aestivum clone PAAC-SCGCA5 AFLP sequence.	UniRef100.T1LCX9
236	AK333949	Triticum aestivum cDNA, clone : WT008_P23, cultivar : Chinese Spring.	
238	FN564428	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0091b.	
239	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
240	HQ435325	Triticum aestivum clone BAC 1J9 Tmemb.185A domain-containing protein (1J9.1), EamA domain-containing protein (1J9.2), and Rht-D1b (Rht-D1b) genes, complete cds, complete sequence.	UniRef100.M7ZYV2
242	HQ390774	Triticum aestivum clone UCDA01225 genomic sequence.	UniRef100.M7ZSC0
243	AK330263	Triticum aestivum cDNA, clone : SET4.A13, cultivar : Chinese Spring.	
244	HQ391044	Triticum aestivum clone UCDA01495 genomic sequence.	
245	FN564430	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0464b.	
247	KC912694	Triticum aestivum chloroplast, complete genome.	
250	FN564430	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0464b.	
251	DQ537335	Triticum aestivum clones BAC 1031P08; BAC 754K10; BAC 1344C16, complete sequence.	
252	AK335757	Triticum aestivum cDNA, clone : WT013.L07, cultivar : Chinese Spring.	
253	FJ225148	Triticum aestivum ferritin 2A gene, complete cds.	
254	HE774676	Triticum aestivum chromosome arm 3DS-specific BAC library, contig ctg447.	
255	AP013106	Triticum timopheevii mitochondrial DNA, complete sequence.	
256	FN645450	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0011b.	
257	AK332440	Triticum aestivum cDNA, clone : WT003_P20, cultivar : Chinese Spring.	
258	AK333064	Triticum aestivum cDNA, clone : WT005_I23, cultivar : Chinese Spring.	
259	AK332804	Triticum aestivum cDNA, clone : WT004.O17, cultivar : Chinese Spring.	
260	FN645450	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0011b.	
261	AK335863	Triticum aestivum cDNA, clone : WT013.P13, cultivar : Chinese Spring.	UniRef100.M7YYG6
262	AB646974	Triticum aestivum PRR gene for pseudo-response regulator, complete cds, allele : Ppd-B1a.1.	
263	GU817319	Triticum aestivum clone BAC_2383A24 chromosome 3B, complete sequence.	
264	AK332413	Triticum aestivum cDNA, clone : WT003.O18, cultivar : Chinese Spring.	UniRef100.M7ZIU6
265	FJ427399	Triticum turgidum clone BAC 738D05 chromosome 4B, partial sequence.	
266	DQ154924	Triticum turgidum RAB7 (RAB7) gene, exons 1, 2 and partial cds; and delta-1-pyrroline-5-carboxylate dehydrogenase (P5CDH) gene, complete cds.	

Contig	Accession	Description	Uniref - EBI
267	FN564433	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0661b.	
268	AK334924	Triticum aestivum cDNA, clone : WT011_I02, cultivar : Chinese Spring.	
269	AK334063	Triticum aestivum cDNA, clone : WT009_A23, cultivar : Chinese Spring.	UniRef100.M7YKC3
270	AY465427	Triticum turgidum subsp. durum putative C3H2C3 RING-finger protein (6G2) gene, complete cds.	
271	DQ167201	Triticum aestivum eukaryotic translation initiation factor 5A1 gene, complete cds.	
274	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
275	GU817319	Triticum aestivum clone BAC_2383A24 chromosome 3B, complete sequence.	
276	AK330938	Triticum aestivum cDNA, clone : SET5_K22, cultivar : Chinese Spring.	UniRef100.M7ZJN7
277	GQ409824	Triticum turgidum subsp. durum cultivar Langdon clone BAC 406B11, complete sequence.	
278	JF946486	Triticum aestivum transposon TREP 3040_Harbinger, complete sequence; pseudo-response regulator (Ppd-B1) gene, Ppd-B1a allele, complete cds; and retrotransposon Gypsy TREP 3457_Danae, complete sequence.	UniRef100.M8B455
279	JF701619	Triticum aestivum cultivar Chinese Spring clone BAC CS12224M17_A, complete sequence.	
280	AK334173	Triticum aestivum cDNA, clone : WT009_C16, cultivar : Chinese Spring.	
281	KF282629	Triticum aestivum cultivar Chinese Spring clone BAC 351D1 chromosome 4A DELLA protein (Rht-A) gene, complete cds, complete sequence.	
284	AK332097	Triticum aestivum cDNA, clone : WT003_C06, cultivar : Chinese Spring.	
286	EU626553	Triticum urartu clone BAC 261N5, complete sequence.	
287	BT009432	Triticum aestivum clone wlmk1.pk0037.b8 :fis, full insert mRNA sequence.	UniRef100.M7YEM7
289	FN564432	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0616b.	
290	AK335883	Triticum aestivum cDNA, clone : SET2_L04, cultivar : Chinese Spring.	
292	HE774676	Triticum aestivum chromosome arm 3DS-specific BAC library, contig ctg447.	
293	AB238931	Triticum monococcum TmABI1 gene for protein phosphatase 2C, complete cds.	
294	JQ269664	Triticum aestivum cultivar WL 711 betaine aldehyde dehydrogenase-like protein mRNA, partial cds.	UniRef100.H9NAU5
296	AK335270	Triticum aestivum cDNA, clone : WT012_H16, cultivar : Chinese Spring.	
297	AY968588	Triticum aestivum ice recrystallization inhibition protein 1 precursor, mRNA, complete cds.	
300	AK332508	Triticum aestivum cDNA, clone : WT004_C11, cultivar : Chinese Spring.	UniRef100.M7ZMZ7
301	DQ537335	Triticum aestivum clones BAC 1031P08; BAC 754K10; BAC 1344C16, complete sequence.	
302	KC912694	Triticum aestivum chloroplast, complete genome.	
303	GU211251	Triticum aestivum pyruvate dehydrogenase E1 component alpha subunit (PDHA1) gene, partial cds.	
306	JF261156	Triticum monococcum cultivar DV92 Mla1 gene, complete cds.	



Contig	Accession	Description	Uniref - EBI
307	AK335953	Triticum aestivum cDNA, clone : SET1.C22, cultivar : Chinese Spring.	UniRef100.M8A0S9
308	KC912694	Triticum aestivum chloroplast, complete genome.	
309	HQ821868	Triticum aestivum cultivar Jasna glutamate dehydrogenase mRNA, complete cds.	UniRef100.E9NX12
311	JF946486	Triticum aestivum transposon TREP 3040.Harbinger, complete sequence; pseudo-response regulator (Ppd-B1) gene, Ppd-B1a allele, complete cds; and retrotransposon Gypsy TREP 3457.Danae, complete sequence.	
312	AK335209	Triticum aestivum cDNA, clone : WT012.F09, cultivar : Chinese Spring.	UniRef100.Q41591
313	AK336081	Triticum aestivum cDNA, clone : SET3.C24, cultivar : Chinese Spring.	UniRef100.M7YMK8
314	AM932685	Triticum aestivum 3B chromosome, clone BAC TA3B95F5.	
316	GQ169688	Triticum aestivum plastid glutamine synthetase 2 (GS2) gene, GS2-D1a allele, complete cds; nuclear gene for plastid product.	
318	FN564428	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0091b.	
320	KC573058	Triticum monococcum subsp. monococcum cultivar DV92 Sr35 region, genomic sequence.	
321	HE996762	Triticum aestivum cv. Arina SNP, chromosome 3B, clone Taes.arina.ctg.66287.	UniRef100.M8A6Z7
324	GQ409824	Triticum turgidum subsp. durum cultivar Langdon clone BAC 406B11, complete sequence.	
326	KC573058	Triticum monococcum subsp. monococcum cultivar DV92 Sr35 region, genomic sequence.	UniRef100.M7Z2V6
327	BT009452	Triticum aestivum clone wlmk8.pk0022.f7 :fis, full insert mRNA sequence.	
328	HE774675	Triticum aestivum chromosome arm 3DS-specific BAC library, contig ctg1484.	
329	AM502905	Triticum aestivum mRNA for MIKC-type MADS-box transcription factor WM32B (WM32B gene).	UniRef100.T1LUM5
330	AK334989	Triticum aestivum cDNA, clone : WT011.L15, cultivar : Chinese Spring.	
333	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
334	AK333292	Triticum aestivum cDNA, clone : WT006.B19, cultivar : Chinese Spring.	UniRef100.M7ZHG4
336	BT009004	Triticum aestivum clone wdk2c.pk018.c16 :fis, full insert mRNA sequence.	UniRef100.Q8S9G0
337	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	UniRef100.M7YVM1
339	AK335226	Triticum aestivum cDNA, clone : WT012.G01, cultivar : Chinese Spring.	
340	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	
341	HQ435325	Triticum aestivum clone BAC 1J9 Tmemb.185A domain-containing protein (1J9.1), EamA domain-containing protein (1J9.2), and Rht-D1b (Rht-D1b) genes, complete cds, complete sequence.	
342	HQ390713	Triticum aestivum clone UCDA01164 genomic sequence.	
344	AK336242	Triticum aestivum cDNA, clone : SET1.E02, cultivar : Chinese Spring.	UniRef100.M8B455
346	FN564434	Triticum aestivum chromosome 3B-specific BAC library, contig ctg0954b.	

## 4 Identifiez les contigs qui coderaient pour des protéines et donnez une table de ceux-ci, contenant contig, numéros d'Accession, Uniref, séquences protéiques

Pour identifier les contigs qui coderaient en protéines, j'ai examiné de nouveau les résultats de blast choisis en numéro 2. Je vais chercher la position des hits dans les résultats retenus, pour ensuite vérifier dans les fichiers genbank si ces hits correspondent à une région codante d'un gène.

Tout d'abord, j'ai créé un script biopython permettant d'extraire les régions des hits pour le résultat choisi à la question précédente (q3.hits.py). Comme certains hits contiennent plusieurs séquences différentes, j'ai gardé chaque partie du résultat ayant un E-value plus petit que 0.01. Les résultats de ce script sont enregistrés dans le fichier hit\_locations.txt.

J'ai ensuite vérifié si ces hits correspondent à une région codante dans les fichiers genbank obtenus à la question précédente. Pour ce faire, j'ai utilisé le script getCDS.py. Après un premier essai, 55 contigs des 223 restants feraient partie d'une région codante. Toutefois, pour 2 des résultats, la région codante trouvée ne contient pas d'information à propos de la séquence protéique obtenue. Par exemple, pour le contig 120, le CDS observé indique que la région codante correspond à un pseudogène qui n'est pas encore identifié.

Ce résultat est étonnant aussi car le blast des contigs sur la base de données Uniref100 a trouvé 70 résultats, donc on devrait s'attendre à un résultat similaire.

J'ai essayé d'autres approches afin d'identifier les gènes qui coderaient pour les protéines. J'ai tenté de faire des blastx des contigs sur chaque contig retenu à la question précédente, toutefois j'ai rencontré des problèmes techniques lors des blasts. Certains blasts prenaient trop de temps à effectuer, et NCBI retournait un message d'erreur. J'ai utilisé le script python q3.blastx.py pour tenter cette approche.

Finalement, j'ai préféré utilisé les résultats de la question précédente pour identifier les contigs qui coderaient pour des protéines. Comme j'ai déjà fait un blastx sur la base de données Uniref100, je considère donc que les contigs qui rencontrent les critères pour les résultats du blast à la question précédente sont ceux qui coderaient pour des protéines.

Dans le tableau des résultats, la séquence protéique donnée est celle correspondante au hit obtenu de blastx. Donc, la séquence représentée est seulement une partie de la séquence représentative du cluster identifié par l'identifiant Uniref. Il serait possible de convertir l'identifiant uniref en identifiant Uniprot afin d'obtenir la séquence complète.

Voici le tableau des contigs qui coderaient pour des protéines :

Contig	Accession	Uniref	Séquence protéique
5	AK331959	UniRef100_M7YGL9	LMMQLLIRNEKDGILVPIQYPLYAS
24	AK330641	UniRef100_M8ABV0	DTSTAESGSEAEDVTSPKALRSYISHPKLTPVRE
26	AK332086	UniRef100_M7YAN9	VITDFMSQVGQGKRRALATNEWLRVPECD
31	HQ391280	UniRef100_M8A3H2	TYDSYAREKQIGGQLLQTYKT
33	HE996560	UniRef100_D9CJA9	RKTMRIQALRCHVLYSHDGSKLNFPV
37	KC290909	UniRef100_T1L6W5	LHRRPLRPGSRPGFCSGRRALL
39	AJ318783	UniRef100_Q8L6A5	GMSAGDRGGVADLIASIKISKIPI
40	AK330233	UniRef100_T1N886	ASIVFISSVSGVVAISSGSYAMTKGAMNQL
47	AK331428	UniRef100_M8AEN7	TVIARSAIRQDAVNKAKSFDER
48	HE996767	UniRef100_M7Z3W8	ITDFALYLVDPADILKRRIALAAVDKLCISKLSDNFFAI
53	HE996642	UniRef100_T1M429	QEDDLQLIDGAMEYHDLVTP
58	JF758499	UniRef100_M7Z3R4	FSKYYSLRSELLVSNMDVSR TKIHLDTISISAT
61	AK333585	UniRef100_M8A091	GLHFLHSIPLIHMDLKPQNILLDDNMTPKIS
67	AK332970	UniRef100_M7YP29	PRLVEIFQRHNVLPNAILSAGSANCAC TAGGGQLYMWGKM KTTGDDTMY
73	EU660896	UniRef100_M7ZVV5	HSGTLNESTNVGVKTTGGPRIGGPEL
75	AB238931	UniRef100_M7YVM1	KRDVSR TNICLDTSRFIHF DNKYFQTD

Contig	Accession	Uniref	Séquence protéique
78	AK335897	UniRef100_M7ZH67	GFCSRKLGGSSALQEHDLLD
86	AK335765	UniRef100_M7YZ42	LKQTARLVYQ TALMESGFNLDPDPKDFASSIYRSV
95	AY487917	UniRef100_Q6RW52	WVAEGFVHHGNQGTSLFLLGLNYFNQLNR
96	KC912694	UniRef100_T1MEW5	KNYGRACYECLRGGLDFTKDDENVNSQPFMRWRDR
100	AK333932	UniRef100_T1N9G3	VFGDNYGDETTWNFDDQDTESVWGSNAMNEPGHHGS
104	HQ391318	UniRef100_T1MZT0	LLENGEDGFIYVGNAVNPATLEQIFGFSSLAGAPNLLALEQFD NALSRK
105	HQ391224	UniRef100_M7ZMV6	ATRIFSNASGSYSSNVNLAVENASWTDEKQLQDM
109	AK335062	UniRef100_M7ZAY8	ELHALIIGINFEEIDFDKNDVVDKIMDDFD
112	AK332529	UniRef100_M7ZA56	EMAATFNVNAEAGLQKLDGYLLSRS
115	AY049041	UniRef100_T1L6Y4	PRTRRLSADCSSCSRGESGSPRAGR
119	AK333035	UniRef100_Q9FT38	NGTPLAPNRIKDCRSYPLYQFVREVCGTEYLTGEKTRSPGEE LNKV
121	HE996280	UniRef100_M8AZM6	FYIAGESYGGHYVPQL
125	KC912694	UniRef100_T1LKW9	AGVFGGSLFSAMHGSLVTSSLIRETTENESAN
127	AK330423	UniRef100_M8AIQ8	NELILSDEDVVRFQIGEVFAHMPVDDVEA
131	HF541875	UniRef100_M7ZGW4	RAQQRLQEEGCVVDIKLFSGAVAGELLSAAY
132	CT009735	UniRef100_T1LKM3	GYMAPERIDEGITPKSDIFSLGVIIMEI
136	KC912694	UniRef100_M7ZC27	ANRVALEACVQARNEGRDLAREGNEIIRAACKWSPEL
137	AK332255	UniRef100_M7YN64	ATGKTIMTAAQMVKPVSLLEGGKSPLVIFDDVAD
146	EF219468	UniRef100_M7YF70	NLSAKLEGDDLDFAFKKNVESATKYLLSKLKDQLFFVGES
150	AK330745	UniRef100_M7YEA6	SFYTMKAVNNNSRVSKLTT
151	AK335219	UniRef100_M7ZDI5	RHTIEGSDDMPAHIKSSMFGCALT
152	AK335725	UniRef100_M7ZVF6	LCTDDIPISSATEEDRQL
162	KC912694	UniRef100_T1LKW9	AAWPVVGIWFTALGIST
165	KC912694	UniRef100_T1LKW9	FQYASFNNRSRLHFFLAAPVVGIIWFTALG
171	KC175605	UniRef100_M1NQF6	PLDITVISRMKQFRMTMNLKKVALKIVAESLSEEEIVG
185	JX040632	UniRef100_M7ZP85	VYRVLSAACEDGDLSIQEIDAVIDIFRRN
194	AK330641	UniRef100_T1MAM1	PTRHDDYHMLLRFLKARKFDIEKAKQMWTDMLQWRKEYGT DTI
213	AK333846	UniRef100_M7YLY4	PPCGKPASSRTRRCDSVQRDMVFITGEFQMMQAFIKAERVEN
222	KC912694	UniRef100_T1LKW9	LGISTMAFNLGNFNFNQSVVDSQGRVINTW
227	AK332238	UniRef100_M7ZLU3	TERAYKYRPLKVVEFDQPYPQCIAYLDLKRE
231	GU211169	UniRef100_D2KFH0	SRCLAINSVAAHILHEQQHQHQQQQYSWG
233	AF532601	UniRef100_M7ZK96	DEVRRKELKLDSPVVENGENWSVGQRQLVCLG
235	AF389882	UniRef100_T1LCX9	KGICEGLHYLHENHIVHLDLKPANILLDDNMVPKI
240	HQ435325	UniRef100_M7ZYV2	NPVLKVMLLDHDDEPTNYEEAMMSPSDKWLEAMKSEIG
242	HQ390774	UniRef100_M7ZSC0	LSALAKYTQGFSGADITEICQRACKYAIEN
261	AK335863	UniRef100_M7YYG6	LLSFMMDDALTTGSIRSTDGEK
264	AK332413	UniRef100_M7ZIU6	FIAVIVCWIKEGDSKLFFLATIYALLGIPLSYLMWYRPLYRAM R
269	AK334063	UniRef100_M7YKC3	KPDNILLDDNMVPKIADFGLSKYFRAGLSFQNLDEH
276	AK330938	UniRef100_M7ZJN7	VDPVDVVSCLRKGWSASIDSVGPAKEP
278	JF946486	UniRef100_M8B455	YSIRLEILVLEMIVSRLILVIDTSILFNF
287	BT009432	UniRef100_M7YEM7	GSYGNLFRVFGSTPGSTEVT'TLEASRNPMRRQ
294	JQ269664	UniRef100_H9NAU5	AVIKVSEHASWSGCFYSRIIQAALLAV
300	AK332508	UniRef100_M7ZMZ7	DTAIALALRESKPVYLSISCNLPGLPHPTF
307	AK335953	UniRef100_M8A0S9	DNRINKAEILFTGVACFLVAVILGSASVHASN
309	HQ821868	UniRef100_E9NX12	TMAWILDEYSKFHGYSPAVVTGKPVDLGGSLG
312	AK335209	UniRef100_Q41591	LLTTFTVDEFATPGLKSILSLVVP
313	AK336081	UniRef100_M7YMK8	REAYDRGKLVPEPNDVSEARRKLVELMLLR
321	HE996762	UniRef100_M8A6Z7	DLEDSTASEAPDAYKAAWTLKGA
326	KC573058	UniRef100_M7Z2V6	MKNKGLASLNSVVELLSEIVNRSMIQPIDINVDKGMEKSYCIHD MVIDSIC

Contig	Accession	Uniref	Séquence protéique
329	AM502905	UniRef100_T1LUM5	LWQREAASLRQQLHDLQESHK
334	AK333292	UniRef100_M7ZHG4	VKQPYNRLRDKFPAASFSGRPNLSEAGFDLLNKLLTY
336	BT009004	UniRef100_Q8S9G0	SPNYAAPEVISGKLYAGPEVDVWSCGVIL
337	FN564434	UniRef100_M7YVM1	MDKRDVSRNICLDTSRFIHFDNKYFQTD
344	AK336242	UniRef100_M8B455	IELVSYISIRLEILVLEMIVSRLILVIDTSILFNF

## 5 Question 4

- a Alignez ces CDS en utilisant ClustalW, dialign et Mavid
- b Discutez de la performance en terme de facilité d'utilisation et de rapidité des différents programmes.
- c Analysez et discutez de la qualité de l'alignement donné par chaque méthode.
- d Quel est votre meilleur alignement ? Justifiez votre choix.

## 6 Annexes

### 1 Tableau de la taille et du taux de GC des contigs

Contig	Taille	Taux GC	Contig	Taille	Taux GC
1	91	51.65	2	182	31.87
3	183	42.62	4	103	39.81
5	83	42.17	6	90	42.22
7	100	46.00	8	88	38.64
9	122	33.61	10	110	37.27
11	140	43.57	12	96	37.50
13	120	37.50	14	114	39.47
15	79	64.56	16	238	41.60
17	143	55.24	18	97	51.55
19	143	53.85	20	82	45.12
21	209	45.45	22	105	38.10
23	150	48.67	24	102	47.06
25	101	36.63	26	88	43.18
27	151	35.76	28	137	59.85
29	97	44.33	30	112	25.89
31	70	38.57	32	125	44.80
33	83	40.96	34	95	32.63
35	201	38.31	36	203	30.05
37	69	79.71	38	153	36.60
39	75	44.00	40	96	46.88
41	52	44.23	42	74	45.95
43	96	63.54	44	115	43.48
45	87	44.83	46	96	48.96
47	82	45.12	48	125	40.00
49	131	35.88	50	102	49.02
51	70	47.14	52	80	43.75
53	99	42.42	54	124	40.32
55	108	45.37	56	142	40.85
57	87	36.78	58	133	40.60
59	112	33.04	60	90	40.00
61	94	44.68	62	95	44.21
63	103	44.66	64	185	59.46
65	142	41.55	66	99	40.40
67	150	49.33	68	84	34.52
69	46	50.00	70	110	34.55
71	80	47.50	72	134	50.00
73	113	53.10	74	93	45.16
75	179	36.87	76	82	46.34
77	106	42.45	78	61	45.90
79	157	31.21	80	111	69.37
81	92	47.83	82	183	45.90
83	51	39.22	84	143	34.97
85	87	40.23	86	102	62.75
87	51	29.41	88	116	46.55
89	107	42.99	90	76	46.05
91	73	47.95	92	104	36.54
93	201	45.27	94	85	31.76
95	99	41.41	96	108	38.89
97	63	33.33	98	131	45.80
99	62	25.81	100	111	40.54

101	121	38.84	102	109	43.12
103	70	37.14	104	148	43.92
105	105	46.67	106	103	70.87
107	105	37.14	108	104	42.31
109	92	41.30	110	80	38.75
111	120	22.50	112	101	47.52
113	69	44.93	114	76	38.16
115	92	76.09	116	69	36.23
117	94	47.87	118	55	34.55
119	138	67.39	120	112	38.39
121	78	47.44	122	111	32.43
123	104	31.73	124	136	42.65
125	104	46.15	126	72	34.72
127	88	47.73	128	76	38.16
129	120	42.50	130	114	42.98
131	105	49.52	132	100	39.00
133	114	42.11	134	69	42.03
135	112	41.07	136	117	49.57
137	106	45.28	138	79	54.43
139	117	51.28	140	100	67.00
141	146	43.15	142	113	46.02
143	120	44.17	144	155	40.65
145	88	40.91	146	121	46.28
147	98	47.96	148	100	50.00
149	81	44.44	150	60	41.67
151	80	46.25	152	56	41.07
153	109	44.04	154	89	55.06
155	179	44.69	156	107	43.93
157	99	38.38	158	101	35.64
159	125	46.40	160	80	41.25
161	93	56.99	162	57	47.37
163	79	50.63	164	221	32.58
165	96	41.67	166	94	45.74
167	119	32.77	168	89	35.96
169	81	51.85	170	94	39.36
171	115	46.09	172	92	41.30
173	117	43.59	174	215	41.40
175	83	30.12	176	111	37.84
177	104	32.69	178	76	46.05
179	204	33.82	180	159	41.51
181	123	35.77	182	87	27.59
183	100	40.00	184	90	46.67
185	93	41.94	186	67	40.30
187	138	39.13	188	83	36.14
189	81	32.10	190	77	45.45
191	101	46.53	192	78	42.31
193	82	46.34	194	131	38.93
195	96	34.38	196	69	47.83
197	115	47.83	198	90	51.11
199	154	45.45	200	77	49.35
201	93	41.94	202	189	42.86
203	84	39.29	204	96	37.50
205	109	56.88	206	81	46.91
207	106	51.89	208	131	29.77
209	94	45.74	210	75	36.00

211	109	36.70	212	120	55.83
213	128	57.03	214	77	38.96
215	121	36.36	216	72	50.00
217	139	48.20	218	71	49.30
219	106	44.34	220	84	39.29
221	81	35.80	222	91	35.16
223	98	43.88	224	104	33.65
225	84	46.43	226	130	31.54
227	94	46.81	228	166	41.57
229	138	41.30	230	90	45.56
231	95	47.37	232	94	45.74
233	98	55.10	234	92	41.30
235	130	33.08	236	79	43.04
237	127	46.46	238	203	43.84
239	92	40.22	240	127	47.24
241	120	39.17	242	95	49.47
243	174	35.06	244	115	46.09
245	155	36.13	246	84	47.62
247	82	52.44	248	103	39.81
249	94	48.94	250	101	33.66
251	90	37.78	252	73	64.38
253	110	42.73	254	125	36.80
255	82	53.66	256	152	24.34
257	80	43.75	258	95	38.95
259	61	49.18	260	106	36.79
261	68	39.71	262	95	35.79
263	114	41.23	264	133	44.36
265	93	40.86	266	97	42.27
267	99	43.43	268	133	40.60
269	118	33.90	270	157	38.85
271	188	42.55	272	94	52.13
273	126	26.98	274	247	34.82
275	101	31.68	276	104	71.15
277	94	34.04	278	149	34.90
279	111	32.43	280	139	47.48
281	95	32.63	282	87	27.59
283	107	72.90	284	115	30.43
285	133	48.87	286	112	33.93
287	100	48.00	288	128	41.41
289	90	41.11	290	92	40.22
291	137	48.18	292	90	41.11
293	153	41.83	294	84	45.24
295	80	40.00	296	82	28.05
297	92	51.09	298	108	70.37
299	101	40.59	300	94	64.89
301	173	35.26	302	112	41.96
303	107	37.38	304	106	38.68
305	81	40.74	306	118	33.05
307	95	47.37	308	90	44.44
309	96	51.04	310	94	43.62
311	99	40.40	312	121	35.54
313	89	49.44	314	94	39.36
315	99	56.57	316	70	42.86
317	90	51.11	318	233	38.63
319	118	48.31	320	118	33.05

321	76	48.68	322	87	37.93
323	105	47.62	324	117	39.32
325	88	47.73	326	179	37.43
327	262	36.64	328	166	37.35
329	200	50.00	330	128	34.38
331	111	36.94	332	119	36.97
333	150	40.00	334	113	42.48
335	94	43.62	336	91	47.25
337	174	33.33	338	86	41.86
339	187	34.76	340	73	36.99
341	92	39.13	342	114	37.72
343	80	43.75	344	153	35.29
345	77	54.55	346	92	43.48

346 contigs, taille moyenne : 109.176300578 Taux GC moyen : 42.9628288283

## 2 Script biopython de calcul des fréquences nucléotidiques

```

1  # -*- coding: utf-8 -*-#
2  from Bio import SeqIO
3  from Bio.SeqRecord import SeqRecord
4  handle = open("NC_000002.202564986-202645895.gb", "r")
5  seq_record = SeqIO.parse(handle, 'gb')
6  for seq in seq_record:
7      dist_a = seq.seq.count("A")
8      dist_c = seq.seq.count("C")
9      dist_g = seq.seq.count("G")
10     dist_t = seq.seq.count("T")
11     print "A: count: " + str(dist_a) + " % " + \
12         str(float(dist_a)/len(seq)*100)
13     print "C: count: " + str(dist_c) + " % " + \
14         str(float(dist_c)/len(seq)*100)
15     print "G: count: " + str(dist_g) + " % " + \
16         str(float(dist_g)/len(seq)*100)
17     print "T: count: " + str(dist_t) + " % " + \
18         str(float(dist_t)/len(seq)*100)
19     print "total: " + str(dist_a+dist_c+dist_g+dist_t)

```

## 3 Script biopython Pour choisir 5 contigs au hasard à partir du résultat de CAP3, et d'effectuer un blast sur ces contigs

```

1  # -*- coding: utf-8 -*- #
2  import random
3  from Bio.Blast import NCBIWWW
4
5  contigs = {}
6  contig_no = None
7  contig_seq = ""
8  contig_size = 0
9
10 with open("seq.data.cap.contigs", "r") as f:
11     for line in f:

```



```

12         # on regarde d'abord si c'est un contig ou non
13         if line[0] == '>':
14             if contig_no == None:
15                 contig_no = int(line[7:])
16             if contig_seq != "":
17                 contigs.update({contig_no: contig_seq})
18                 contig_seq = ""
19                 contig_no = int(line[7:])
20                 contig_size += 1
21         else :
22             contig_seq = contig_seq + line.replace("\n", "")
23         contigs.update({contig_no: contig_seq})
24         contig_size += 1
25
26     # Maintenant, on a nos contigs, on en choisit 5 au hasard
27     random_contig = []
28
29     # Je m'assure ici de ne pas avoir de doublon
30     for i in range(5):
31         random_c = random.randint(1, contig_size)
32         while random_c in random_contig:
33             random_c = random.randint(1, contig_size)
34         random_contig.append(random_c)
35
36     # On blast maintenant les contigs choisis :
37     for i in random_contig:
38         result_handle = NCBIWWW.qblast("blastn", "nr", contigs[i])
39
40         # on enregistre le resultat
41         nom_fichier = "blast_contig_" + str(i) + ".xml"
42         save_file = open(nom_fichier, "w")
43         save_file.write(result_handle.read())
44         save_file.close()
45         result_handle.close()
46
47     print "5_contigs_cherches"

```

#### 4 Script Biopython pour obtenir les fichiers d'accension des 10 premiers résultats du blast pour un contig donné.

```

1  # -*- coding: utf-8 -*- #
2
3  # Parser pour un fichier XML de resultat blast
4  # Specifique a la question 2 du devoir 1. Je sais
5  # ici que chaque hit a seulement un hsp, donc en
6  # specifiant le # d'accension et le subjct_start et end,
7  # j'obtiens ce que je cherche
8
9  import sys
10 import os
11 from Bio.Blast import NCBIXML
12 from Bio import SeqIO
13 from Bio.SeqRecord import SeqRecord
14 from Bio import Entrez
15
16 # On choisit une E-VALUE

```

```

17 E_VALUE_THRESH = 0.04
18 Entrez.email = "glahaie@gmail.com"
19 path = "annexes/question.2/"
20
21 path_fichier = path + "blast_contig_"+sys.argv[1] + ".xml"
22
23 with open(path_fichier) as fichier:
24     blast_record = NCBIXML.read(fichier)
25     i = 0
26     path_result = path + "contig_"+sys.argv[1]+"/"
27     if not os.path.exists(path_result):
28         os.makedirs(path_result)
29     for alignment in blast_record.alignments:
30         for hsp in alignment.hsps:
31             if hsp.expect < E_VALUE_THRESH:
32 #On obtient alors le fichier genbank
33                 handle = Entrez.efetch(db="nucleotide", rettype="gb",
34                                         retmode="text", id=alignment.accession,
35                                         seq_start=hsp.sbjct_start, seq_stop=hsp.sbjct_end)
36                 seq_record= SeqIO.read(handle, "gb")
37                 handle.close()
38                 nom_fichier = path_result + alignment.accession + ".gb"
39                 SeqIO.write(seq_record, nom_fichier, "gb")
40
41         i += 1
42         if i > 10:
43             break

```

## 5 Log de l'exécution de clustalw2.

### CLUSTAL 2.1 Multiple Sequence Alignments

```

Sequence format is Pearson
Sequence 1: lc1|XM_518463.3_cdsid_XP_518463.2      2058 bp
Sequence 2: lc1|XM_003833312.1_cdsid_XP_003833360.1  2004 bp
Sequence 3: lc1|XM_004043991.1_cdsid_XP_004044039.1  2004 bp
Sequence 4: lc1|XM_002816867.2_cdsid_XP_002816913.1  2043 bp
Sequence 5: lc1|XM_003266293.1_cdsid_XP_003266341.1  2004 bp
Sequence 6: lc1|XM_005553053.1_cdsid_XP_005553110.1  2004 bp
Sequence 7: lc1|NM_001266091.1_cdsid_NP_001253020.1  2043 bp
Sequence 8: lc1|XM_003922988.1_cdsid_XP_003923037.1  2004 bp
Start of Pairwise alignments
Aligning...

Sequences (1:2) Aligned. Score: 98
Sequences (1:3) Aligned. Score: 97
Sequences (1:4) Aligned. Score: 97
Sequences (1:5) Aligned. Score: 97
Sequences (1:6) Aligned. Score: 96
Sequences (1:7) Aligned. Score: 96
Sequences (1:8) Aligned. Score: 96
Sequences (2:3) Aligned. Score: 99
Sequences (2:4) Aligned. Score: 98
Sequences (2:5) Aligned. Score: 98
Sequences (2:6) Aligned. Score: 97

```

```

Sequences (2:7) Aligned. Score: 97
Sequences (2:8) Aligned. Score: 97
Sequences (3:4) Aligned. Score: 98
Sequences (3:5) Aligned. Score: 98
Sequences (3:6) Aligned. Score: 98
Sequences (3:7) Aligned. Score: 97
Sequences (3:8) Aligned. Score: 96
Sequences (4:5) Aligned. Score: 98
Sequences (4:6) Aligned. Score: 98
Sequences (4:7) Aligned. Score: 98
Sequences (4:8) Aligned. Score: 97
Sequences (5:6) Aligned. Score: 98
Sequences (5:7) Aligned. Score: 98
Sequences (5:8) Aligned. Score: 96
Sequences (6:7) Aligned. Score: 99
Sequences (6:8) Aligned. Score: 96
Sequences (7:8) Aligned. Score: 96
Guide tree file created: [foxp4_ortho.dnd]

```

There are 7 groups  
Start of Multiple Alignment

```

Aligning...
Group 1: Sequences: 2      Score:37737
Group 2: Sequences: 2      Score:37091
Group 3: Sequences: 3      Score:37148
Group 4: Sequences: 4      Score:37047
Group 5: Sequences: 5      Score:37481
Group 6: Sequences: 7      Score:37220
Group 7: Sequences: 8      Score:36779
Alignment Score 440506

```

CLUSTAL-Alignment file created [foxp4\_ortho.aln]

## 6 Log du travail de RepeatMasker sur le fichier foxp4\_ortho.fa.

There were no repetitive sequences detected in /usr/local/rmsserver/tmp/RM2\_foxp4\_ortho.fa\_1383262617

## 7 Log de l'exécution de Mavid sur foxp4\_ortho.fa.

```

./utils/randtree/randtree foxp4_ortho.fa
./mavid ./mavid.ph foxp4_ortho.fa

```

```

*****
*                                     *
*           Welcome to MAVID.         *
*           (version 2.0, build 4)     *
*                                     *
*****

```

```

Aligning 1 versus 1
Aligning [0,2003] to [0,2057]
Aligning 1 versus 2

```

```

Aligning [0,2003] to [0,2058]
Aligning 1 versus 1
Aligning [0,2003] to [0,2042]
Aligning 1 versus 1
Aligning [0,2003] to [0,2003]
Aligning 1 versus 2
Aligning [0,2042] to [0,2003]
Aligning 2 versus 3
Aligning [0,2042] to [0,2042]
Aligning 3 versus 5
Aligning [0,2058] to [0,2042]
MAVID worked!

```

```

clustalw2 ./mavid.mfa -tree

```

## CLUSTAL 2.1 Multiple Sequence Alignments

```

Sequence format is Pearson
Sequence 1: lc1|XM_004043991.1_cdsid_XP_004044039.1 2059 bp
Sequence 2: lc1|XM_005553053.1_cdsid_XP_005553110.1 2059 bp
Sequence 3: lc1|XM_518463.3_cdsid_XP_518463.2 2059 bp
Sequence 4: lc1|XM_003266293.1_cdsid_XP_003266341.1 2059 bp
Sequence 5: lc1|NM_001266091.1_cdsid_NP_001253020.1 2059 bp
Sequence 6: lc1|XM_002816867.2_cdsid_XP_002816913.1 2059 bp
Sequence 7: lc1|XM_003833312.1_cdsid_XP_003833360.1 2059 bp
Sequence 8: lc1|XM_003922988.1_cdsid_XP_003923037.1 2059 bp

```

```

Phylogenetic tree file created: [./mavid.ph]

```

```

../utils/root_tree/root_tree ./mavid.ph
./mavid ./mavid.ph foxp4_ortho.fa

```

```

*****
*
*           Welcome to MAVID.           *
*           (version 2.0, build 4)       *
*
*****

```

```

Aligning 1 versus 1
Aligning [0,2003] to [0,2042]
Aligning 1 versus 1
Aligning [0,2057] to [0,2003]
Aligning 1 versus 2
Aligning [0,2003] to [0,2003]
Aligning 3 versus 1
Aligning [0,2003] to [0,2042]
Aligning 1 versus 4
Aligning [0,2003] to [0,2042]
Aligning 2 versus 5
Aligning [0,2042] to [0,2042]
Aligning 1 versus 7
Aligning [0,2003] to [0,2042]
MAVID worked!

```

```
clustalw2 ./mavid.mfa -tree
```

## CLUSTAL 2.1 Multiple Sequence Alignments

Sequence format is Pearson

```
Sequence 1: lc1|XM_003922988.1_cdsid_XP_003923037.1 2098 bp
Sequence 2: lc1|XM_005553053.1_cdsid_XP_005553110.1 2098 bp
Sequence 3: lc1|NM_001266091.1_cdsid_NP_001253020.1 2098 bp
Sequence 4: lc1|XM_003266293.1_cdsid_XP_003266341.1 2098 bp
Sequence 5: lc1|XM_004043991.1_cdsid_XP_004044039.1 2098 bp
Sequence 6: lc1|XM_518463.3_cdsid_XP_518463.2 2098 bp
Sequence 7: lc1|XM_003833312.1_cdsid_XP_003833360.1 2098 bp
Sequence 8: lc1|XM_002816867.2_cdsid_XP_002816913.1 2098 bp
```

Phylogenetic tree file created: [./mavid.ph]

```
../utils/root_tree/root_tree ./mavid.ph
```

## 8 Script biopython pour identifier la composition du vecteur pANNE

```
1  #- coding:utf-8 -*- #
2
3  # Script pour le num ro 3 du devoir 1: Cette partie ne fait
4  #qu'envoyer la requ te blast au serveur du NCBI, et ensuite
5  #enregistre le r sultat dans un fichier.
6
7  from Bio.Blast import NCBIWWW
8  from Bio.Blast import NCBIXML
9
10 path_fichier = "annexes/question_3/"
11 nom_resultat = "blast_fichier"
12 LEN_THRESH = 100
13 E_THRESH = 1e-50
14 # Tout d'abord on ouvre le fichier
15
16 sequence = ""
17 with open(path_fichier+"pANNE.txt", 'r') as f:
18     for line in f:
19         sequence = sequence + line.strip()
20
21 #On enl ve les retour de chariot du fichier
22 i = 1
23 while len(sequence) > LEN_THRESH:
24
25 #Maintenant, on fait le blast
26     print "i=" + str(i)
27     print "on fait un blast sur la s quence de longueur "
28         + str(len(sequence))
29     result_handle = NCBIWWW.qblast("blastn", "nr",
30         sequence, megablast=True)
31
32 #on enregistre le r sultat
33     save_file = open(path_fichier+nom_resultat+str(i)+".xml", "w")
```

```

34     save_file.write(result_handle.read())
35     save_file.close()
36     result_handle.close()
37
38     list_start = []
39     list_end = []
40     sequences = []
41 #Maintenant on enlève de la séquence les zones identifiées
42     with open(path_fichier+nom_resultat+str(i)+".xml", "r") as result:
43         blast_record = NCBIXML.read(result)
44         alignment = blast_record.alignments[0]
45         for alignment in blast_record.alignments:
46             for hsp in alignment.hsps:
47 #On met jour la séquence pour enlever ce résultat
48                 if hsp.expect < E_THRESH:
49                     list_start.append(hsp.query_start)
50                     list_end.append(hsp.query_end)
51
52             break
53 #On a les points à enlever
54 #sort sur les listes
55         list_start.sort()
56         list_end.sort()
57         start = -1
58         end = -1
59         for s_start, s_end in zip(list_start, list_end):
60             if end < 0:
61                 sequences.append(sequence[: s_start - 1])
62                 end = s_start - 1
63             else:
64                 end = s_start - 1
65                 sequences.append(sequence[start: end])
66                 start = s_end - 1
67             sequences.append(sequence[start:])
68             sequence = ""
69             for fragment in sequences:
70                 sequence += fragment
71 #Pour vérifier les résultats, j'enregistre la nouvelle
72 séquence dans un fichier
73             print "On écrit le reste de la séquence avec le " + str(i)
74             with open(path_fichier+"pANNE"+str(i)+".txt", "w") as f:
75                 f.write(sequence)
76             i += 1

```

## 9 Temps d'exécution des alignements multiples

Programme	Temps d'exécution
ClustalW	real 0m4.840s
dalign	real 0m18.424s
Mavid	real 0m0.296s

## 10 Script Biopython pour obtenir les fichiers d'accèsion des 10 premiers résultats du blast pour un contig donné.

```

1  #- coding:utf-8 -#
2
3  from reportlab.lib import colors
4  from reportlab.lib.units import cm
5  from Bio.Graphics import GenomeDiagram
6  from Bio.Graphics.GenomeDiagram import CrossLink
7  from reportlab.lib import colors
8
9  gd_diagram = GenomeDiagram.Diagram("Composition_du_vecteur_pANNE.txt")
10 gd_track_for_features = gd_diagram.new_track(1, name="Annotated_Features",
11      start=0, end=6627)
12 gd_feature_set = gd_track_for_features.new_set()
13 from Bio.SeqFeature import SeqFeature, FeatureLocation
14
15 colors = [colors.green, colors.lightgreen, colors.teal, colors.darkgreen,
16     colors.seagreen, colors.lawngreen, colors.olivedrab]
17
18 #Essai la main, la lecture des fichiers XML
19 #blast #1: pHT2
20 feature = SeqFeature(FeatureLocation(1, 1056), strand = +1)
21 gd_feature_set.add_feature(feature, name="pHT2", label=True, color=colors[0],
22     sigil="ARROW", arrowhead.length=0.5, arrowshaft.height=0.1)
23 feature = SeqFeature(FeatureLocation(5342, 5798), strand = +1)
24 gd_feature_set.add_feature(feature, name="pHT2", label=True, color=colors[1],
25     sigil="ARROW", arrowhead.length=1, arrowshaft.height=0.1)
26 feature = SeqFeature(FeatureLocation(3495, 5341), strand = +1)
27 gd_feature_set.add_feature(feature, name="pHT2", label=True, color=colors[2],
28     sigil="ARROW", arrowhead.length=1, arrowshaft.height=0.1)
29 feature = SeqFeature(FeatureLocation(5798, 6627), strand = +1)
30 gd_feature_set.add_feature(feature, name="pHT2", label=True, color=colors[3],
31     sigil="ARROW", arrowhead.length=1, arrowshaft.height=0.1)
32 feature = SeqFeature(FeatureLocation(3039, 3494), strand = +1)
33 gd_feature_set.add_feature(feature, name="pHT2", label=True, color=colors[4],
34     sigil="ARROW", arrowhead.length=1, arrowshaft.height=0.1)
35 feature = SeqFeature(FeatureLocation(1742, 2057), strand = +1)
36 gd_feature_set.add_feature(feature, name="pHT2", label=True, color=colors[5],
37     sigil="ARROW", arrowhead.length=1, arrowshaft.height=0.1)
38 feature = SeqFeature(FeatureLocation(2776, 3039), strand = +1)
39 gd_feature_set.add_feature(feature, name="pHT2", label=True, color=colors[6],
40     sigil="ARROW", arrowhead.length=1, arrowshaft.height=0.1)
41
42 #blast 2:
43 feature = SeqFeature(FeatureLocation(1057, 1741), strand = +1)
44 gd_feature_set.add_feature(feature, name="PGeneClip", label=True, color="blue",
45     sigil="ARROW", arrowhead.length=1)
46
47 #blast 3
48 feature = SeqFeature(FeatureLocation(2058, 2770), strand = +1)
49 gd_feature_set.add_feature(feature, name="Cloning_vector_EN.Cherry", label=True,
50     color="red", sigil="ARROW", arrowhead.length=1)
51
52 #On essaie d'ajouter d'autres track pour repr senter la position des blasts
53 gd_track_for_features = gd_diagram.new_track(1, name="pHT2", start=0, end=4924)
54 gd_feature_set = gd_track_for_features.new_set()
55 feature = SeqFeature(FeatureLocation(2246, 4092), strand = None)
56 gd_feature_set.add_feature(feature, name="pHT2", label=False, color=colors[2],
57     sigil="ARROW", arrowhead.length=0.2, arrowshaft.height=0.1)
58 feature = SeqFeature(FeatureLocation(1, 1058), strand = None)

```

```

59 gd_feature_set.add_feature(feature, name="pHT2", label=False, color=colors[0],
60     sigil="ARROW", arrowhead_length=0.2, arrowshaft_height=0.1)
61 feature = SeqFeature(FeatureLocation(4093, 4922), strand = +1)
62 gd_feature_set.add_feature(feature, name="pHT2", label=False, color=colors[3],
63     sigil="ARROW", arrowshaft_height=0.1)
64 feature = SeqFeature(FeatureLocation(2795, 2339), strand = -1)
65 gd_feature_set.add_feature(feature, name="pHT2", label=False, color=colors[1],
66     sigil="ARROW", arrowshaft_height=0.1)
67 feature = SeqFeature(FeatureLocation(2340, 2795), strand = +1)
68 gd_feature_set.add_feature(feature, name="pHT2", label=False, color=colors[4],
69     sigil="ARROW", arrowshaft_height=0.1)
70 feature = SeqFeature(FeatureLocation(743, 1058), strand = +1)
71 gd_feature_set.add_feature(feature, name="pHT2", label=False, color=colors[5],
72     sigil="ARROW", arrowshaft_height=0.1)
73 feature = SeqFeature(FeatureLocation(1984, 2246), strand = +1)
74 gd_feature_set.add_feature(feature, name="pHT2", label=False,
75     color=colors[6], sigil="ARROW", arrowshaft_height=0.1)
76
77 #On essaie d'ajouter d'autres track pour repr senter la position des blasts
78 gd_track_for_features = gd_diagram.new_track(1, name="PGeneClip", start=0,
79     end=5267)
80 gd_feature_set = gd_track_for_features.new_set()
81 feature = SeqFeature(FeatureLocation(1879, 2563), strand = +1)
82 gd_feature_set.add_feature(feature, name="PGeneClip", label=False, color="blue",
83     sigil="ARROW", arrowhead_length=1, arrowshaft_height=0.1)
84
85 gd_track_for_features = gd_diagram.new_track(1,
86     name="Cloning_vector_EN.Cherry", start=0, end=10649)
87 gd_feature_set = gd_track_for_features.new_set()
88 feature = SeqFeature(FeatureLocation(7102, 7813), strand = +1)
89 gd_feature_set.add_feature(feature, name="Cloning_Vector_EN.Cherry", label=False,
90     color="red", sigil="ARROW", arrowhead_length=1, arrowshaft_height=0.1)
91
92 gd_diagram.draw(format='linear', pagesize="LETTER", orientation="portrait",
93     fragments=1, start=0, end=10649)
94 gd_diagram.write("GD_labels_default.eps", "eps")

```

## 11 Fichiers genbank utilisés pour ce rapport

Nom	Numéro d'accension	Nom	Numéro d'accension
Cloning Vector EN.Cherry, complete sequence	HM771696.1	PGeneClip hMGFP Vector, complete sequence	AY744386.1
Expression vector pHT2, complete sequence	AY773970.1	Homo sapiens chromosome 6, GRCh37.p13 Primary Assembly	NC_000006.11
SARS coronavirus MA15 ExoN1 isolate d3om5, complete genome	JF292906.1	SARS coronavirus MA15 isolate d2ym4, complete genome	JF292909.1
SARS coronavirus MA15 isolate d4ym5, complete genome	JF292915.1	SARS coronavirus HKU-39849 isolate recSARS-CoV HKU-39849, complete genome	JN854286.1
SARS coronavirus HKU-39849 isolate UOB, complete genome	JQ316196.1	SARS coronavirus isolate Tor2/FP1-10912, complete genome	JX163923.1
SARS coronavirus isolate Tor2/FP1-10851, complete genome	JX163924.1	SARS coronavirus isolate Tor2/FP1-10895, complete genome	JX163925.1



Nom	Numéro d'accension	Nom	Numéro d'accension
SARS coronavirus isolate Tor2/FP1-10912, complete genome	JX163926.1	SARS coronavirus isolate Tor2/FP1-10851, complete genome	JX163927.1
SARS coronavirus isolate Tor2/FP1-10895, complete genome	JX163928.1	SARS coronavirus SinP3, complete genome	AY559090.1
SARS coronavirus HKU-39849 isolate TCVSP-HARROD-00001, complete genome	GU553363.1	SARS coronavirus HKU-39849 isolate recSARS-CoV HKU-39849, complete genome	JN854286.1
SARS coronavirus HKU-39849 isolate TCVSP-HARROD-00002, complete genome	GU553364.1	SARS coronavirus HKU-39849 isolate TCVSP-HARROD-00003, complete genome	GU553365.1
SARS coronavirus Sin850, complete genome	AY559096.1	SARS coronavirus MA15 isolate P3pp3, complete genome	FJ882948.1
SARS coronavirus MA15 ExoN1 isolate P3pp3, complete genome	FJ882951.1	SARS coronavirus MA15 isolate P3pp4, complete genome	FJ882952.1
SARS coronavirus MA15, complete genome	FJ882957.1	SARS coronavirus MA15 isolate P3pp7, complete genome	FJ882958.1
SARS coronavirus MA15 ExoN1 isolate P3pp6, complete genome	FJ882959.1	SARS coronavirus MA15 isolate P3pp5, complete genome	FJ882961.1
SARS coronavirus ExoN1 isolate c5P1, complete genome	JF292922.1	SARS coronavirus ExoN1 isolate c5P10, complete genome	JX162087.1
SARS coronavirus ExoN1 strain	KF514407.1	PREDICTED : Pan troglodytes forkhead box P4, transcript variant 2 (FOXP4), mRNA	XM.518463.3
PREDICTED : Pan paniscus forkhead box P4, transcript variant 2 (FOXP4), mRNA.	XM.003833312.1	PPREDICTED : Gorilla gorilla gorilla forkhead box P4, transcript variant 2 (FOXP4), mRNA.	XM.004043991.1
PREDICTED : Pongo abelii forkhead box P4, transcript variant 1 (FOXP4), mRNA.	XM.002816867.2	PREDICTED : Nomascus leucogenys forkhead box P4, transcript variant 2 (FOXP4), mRNA.	XM.003266293.1
PREDICTED : Macaca fascicularis forkhead box P4 (FOXP4), transcript variant X3, mRNA.	XM.005553053.1	Macaca mulatta forkhead box P4 (FOXP4), mRNA.	NM.001266091.1
PREDICTED : Saimiri boliviensis boliviensis forkhead box P4, transcript variant 2 (FOXP4), mRNA	XM.003922988.1	Homo sapiens chromosome 2, GRCh37.p13 Primary Assembly	NC.000002.11
Homo sapiens amyotrophic lateral sclerosis 2 (juvenile) (ALS2), transcript variant 1, mRNA	NM.020919.3	Homo sapiens amyotrophic lateral sclerosis 2 (juvenile) (ALS2), transcript variant 2, mRNA	NM.001135745.1
Pan troglodytes chromosome 2B, Pan.troglodytes-2.1.4	NC.006470.3	Macaca mulatta chromosome 12, Mmul.051212, whole genome shotgun sequence	NC.007869.1
Canis lupus familiaris breed boxer chromosome 37, CanFam3.1, whole genome shotgun sequence	NC.006619.3	Bos taurus breed Hereford chromosome 2, Bos.taurus.UMD.3.1, whole genome shotgun sequence	AC.000159.1
Mus musculus strain C57BL/6J chromosome 1, GRCm38.p1 C57BL/6J	NC.000067.6	Rattus norvegicus strain BN/SsNHsdMCW chromosome 9, Rnor.5.0	NC.005108.3
Gallus gallus isolate #256 breed Red Jungle fowl, inbred line UCD001 chromosome 7, Gallus.gallus-4.0, whole genome shotgun sequence	NC.006094.3	Danio rerio strain Tuebingen chromosome 6, Zv9	NC.007117.5

Nom	Numéro d'accension	Nom	Numéro d'accension
Homo sapiens chromosome 2 genomic contig, GRCh37.p13 Primary Assembly	NT.005403.17	alsin isoform 1 [Homo sapiens]	NP_065970.2
alsin [Pan troglodytes]	NP_001073389.1	forkhead box protein P4 isoform 1 [Homo sapiens]	NP_001012426.1

## 12 Fichiers de gène de NCBI utilisé pour ce rapport

Nom	Gene ID	Nom	Gene ID
ALS2 amyotrophic lateral sclerosis 2 (juvenile) [ Homo sapiens (human) ]	57679	ALS2 amyotrophic lateral sclerosis 2 (juvenile) [ Pan troglodytes (chimpanzee) ]	470613
ALS2 amyotrophic lateral sclerosis 2 (juvenile) [ Macaca mulatta (Rhesus monkey) ]	703263	ALS2 amyotrophic lateral sclerosis 2 (juvenile) [ Canis lupus familiaris (dog) ]	100856109
ALS2 amyotrophic lateral sclerosis 2 (juvenile) [ Bos taurus (cattle) ]	535750	Als2 amyotrophic lateral sclerosis 2 (juvenile) [ Mus musculus (house mouse) ]	74018
Als2 amyotrophic lateral sclerosis 2 (juvenile) [ Rattus norvegicus (Norway rat) ]	363235	FOXP4 forkhead box P4 [ Homo sapiens (human) ]	116113

## Références

- [1] Triticum aestivum (ID 11) - Genome - NCBI (2013). Retrieved December 17, 2013 from <http://www.ncbi.nlm.nih.gov/genome/11>.
- [2] Ling HQ, Zhao S, Liu D, Wang J, Sun H, Zhang C, Fan H, Li D, Dong L, Tao Y, et al. Draft genome of the wheat A-genome progenitor Triticum urartu. Nature. 2013 Apr 4;496(7443) :87-90. doi : 10.1038/nature11997. Epub 2013 Mar 24. PubMed PMID : 23535596.
- [3] Whole Chromosome Survey Sequencing (2013). Retrieved December 17, 2013 from <http://www.wheatgenome.org/Projects/IWGSC-Bread-Wheat-Projects/Sequencing/Whole-Chromosome-Survey-Sequencing>
- [4] Sequencing Projects (2013). Retrieved December 17, 2013 from <http://www.wheatgenome.org/Projects/IWGSC-Bread-Wheat-Projects/Sequencing>
- [5] Wilkinson, P.A., Winfield, M.O., Barker, G.L.A., Allen, A.M., BurrIDGE, A, Coghill, J.A., BurrIDGE, A. and Edwards, K.J. 2012. CerealsDB 2.0 : an integrated resource for plant breeders and scientists. BMC Bioinformatics 13 : 219.
- [6] GC content. In Wikipedia. Retrieved December 17, 2013, from <http://en.wikipedia.org/wiki/GC-content>
- [7] Kent WJ, Sugnet CW, Furey TS, Roskin KM, Pringle TH, Zahler AM, Haussler D. The human genome browser at UCSC. Genome Res. 2002 Jun ;12(6) :996-1006.
- [8] Basic Local Alignment Search Tool (Altschul et al., J Mol Biol 215 :403-410; 1990).
- [9] Suzek BE, Huang H, McGarvey P, Mazumder R, Wu CH. UniRef : comprehensive and non-redundant UniProt reference clusters. Bioinformatics. 2007 May 15 ;23(10) :1282-8. Epub 2007 Mar 22. PubMed PMID : 17379688.