Univerzita Pardubice

Fakulta ekonomicko-správní

Bioinformatika

Simona Smejkalová

Bakalářská práce

2009

Univerzita Pardubice
Fakulta ekonomicko-správní
Ústav systémového inženýrství a informatiky
Akademický rok: **2008/2009**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Simona SMEJKALOVÁ**

Studijní program: **B6209 Systémové inženýrství a informatika**

Studijní obor: **Informatika ve veřejné správě**

Název tématu: **Bioinformatika**

Zásady pro vypracování:

1. Popis a problematika bioinformatiky
2. Databáze biologických dat
3. Vyhledávání v biologických datech
4. Porovnání databází a způsobu vyhledávání v nich

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

[1] **CVRČKOVÁ, Fatima. Úvod do praktické bioinformatiky. 2006. vyd. [s.l.] : Academia, 2006. 150 s. ISBN 80-200-1360-1.**

[2] **TROUBSKÁ, Daniela. Bioinformatika [online]. 2007 [cit. 2007-10-19]. Dostupný z WWW: <http://xxx.enemy.cz/desahruza/kulamolekula/Bioinformatika.pdf>.**

[3] **European Bioinformatics Institute [online]. 2006-2007 [cit. 2007-10-19]. Dostupný z WWW: <http://www.ebi.ac.uk/>.**

[4] **National Center for Biotechnology Information [online]. 2007 [cit. 2007-10-19]. Dostupný z WWW: <http://www.ncbi.nlm.nih.gov/>.**

[5] **The Science Creative Quarterly [online]. 2007 [cit. 2007-10-19]. Dostupný z WWW: <http://www.scq.ubc.ca/?p=385>.**

[6] **PAČES, Jan, Genomika a bioinformatika [online]. 2007 [cit. 2007-10-19]. Dostupný z WWW: <http://www.open-science.cz/ov/users/Image/default/C1Kurzy/Biolog/7paces.pdf>.**

Vedoucí bakalářské práce: **Ing. Milan Tomeš**
Ústav systémového inženýrství a informatiky

Datum zadání bakalářské práce: **6. října 2008**

Termín odevzdání bakalářské práce: **1. května 2009**

L.S.

doc. Ing. Renáta Myšková, Ph.D.
děkanka

doc. Ing. Jiří Křupka, Ph.D.
vedoucí ústavu

V Pardubicích dne 6. října 2008

Prohlašuji:

Tuto práci jsem vypracovala samostatně. Veškeré literární prameny a informace, které jsem v práci využila, jsou uvedeny v seznamu použité literatury.
Byla jsem seznámena s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 26. 4. 2009

ANOTACE

V této práci bude uveden popis a problematika bioinformatiky, budou zde vyjmenovány důležité databáze biologických dat, porovnání těchto databází a některé algoritmy, které prohledávají biologická data.

KLÍČOVÁ SLOVA

Bioinformatika, základy molekulární biologie, databáze biologických dat, GenBank, EMBL, digitální sekvence, přiřazení sekvencí, vyhledávací algoritmy, BLAST, FASTA, SSEARCH, evoluční stromy.

TITLE

Bioinformatics

ANNOTATION

This work will contain description and problems of bioinformatics. Then there will be named the most important biological databases, comparison this databases and some searching algorithms those biological data.

KEYWORDS

Bioinformatics, the principals of molecular biology, biological databases, GenBank, EMBL, digital sequence, sequence alignment, searching algorithms, BLAST, FASTA, SSEARCH evolutional trees.

OBSAH

1 Úvod

V dnešní době, kdy nám téměř s veškerou prací pomáhají stroje, přístroje, počítače, díky nimž je práce zkrácena na desetinu a přírůstek dat je exponenciální, tomu není jinak ani při získávání biologických dat. Tím však vyvstává problém, jak získaná data uchovávat a jak v nich dolovat. V této práci bude hlavně popsáno, v jakých databázích data získat a jakým způsobem.

Cílem práce bude jednak čtenáři objasnit základní pojmy BI a vytvořit tak alespoň běžný přehled nad touto problematikou. Dále bude snaha vyjmenovat nejdůležitější databáze a zhodnotit, odkud by mohl uživatel nejlépe čerpat data s důrazem na přehlednost nebo také např. možnost odkazů na další informace. Následně také vysvětlit uživateli principy těch nástrojů, které umožňují vyhledávání a porovnávání biologických dat v databázích, hlavně sekvencí aminokyselin a nukleových kyselin, a pomoct tak tím s výběrem programů, které se nejvíce hodí na prohledávání té určité databáze tou určitou sekvencí.

V první kapitole bude pojem bioinformatika (BI) zařazen do souvislostí tím, že bude uvedena historie, tj. od jaké doby je možno mluvit o bioinformatice, kdy byl poprvé použit tento výraz a díky čemu došlo k rozmachu BI.

V další kapitole bude následovat vysvětlení pojmu BI pomocí definice, a co BI zahrnuje. Toto je rovněž důležité, neboť se lze setkat ve zdrojích proměnlivé důvěryhodnosti (web) někdy i s poněkud bizardními vysvětleními pojmu BI, které nemají mnoho společného s BI, tak jak ji alespoň chápe vědecká společnost zabývající se tímto tématem.

Následuje potřebná kapitola, která zastává první slovo ze složeného výrazu BIOinformatika, neboť se v ní budou popisovat základní principy a pojmy z molekulární biologie. Těmito třemi kapitolami by měl být čtenář již obeznámen s problematikou spojenou s pojmem BI.

Dále se v práci bude věnováno de facto základnímu stavebnímu kamenu BI a to jsou veřejně dostupné zdroje dat. Budou zde vyjmenovány ty nejzákladnější a zároveň nejobsáhlejší databáze biologických dat, ale i druhotné a například i úzce specializované databáze

Poté následuje oddíl, který se zaměřuje na dolování dat v biologických databázích. Toto již souvisí s praktickou BI a s činností bioinformatika. V této šesté a sedmé kapitole je zaměření na práci se sekvenčními daty, píše se zde, co jsou to sekvenční data, jak se s nimi pracuje při hledání v databázích, jaké metody a algoritmy se používají k prohledávání databází a konkrétní příklady algoritmů.

V následné osmé kapitole a podkapitolách bude nastíněna další práce s biologickými daty a tj. mnohočetného přiřazení a s tím související úkony jako přiřazování trojrozměrných struktur a základy modelování struktury proteinů. Budou zde uvedeny i některé příslušné programy pracující s těmito strukturálními daty.

V neposlední řadě také budou zmíněny evoluční stromy, které v podstatě shrnují předešlou práci se strukturálními daty.

V předposlední desáté kapitole se budou porovnávat dvě databáze, resp. jejich webové rozhraní, a doporučovat, které by bylo například vhodné pro začátečníka.

A v poslední jedenácté bude zachycena konkrétní práce se sekvenčními daty. Bude zde doporučeno, jaký program se hodí na jaké prohledávání. Budou se zde prohledávat tři databáze třemi různými algoritmy, vyhodnocovat výsledky prohledávání.

2 Historie bioinformatiky

Bioinformatika je mladá vědecká disciplína, jejíž počátky lze nalézt někdy v 60. letech 20. století. Za zakladatelku je považována Margaret Oakley Dayhoffová (1925-1983), která poprvé použila počítač pro výpočty ve své doktorské práci už v roce 1947. Termín bioinformatika se poprvé objevil až v roce 1991.[3] Za svůj současný rozmach bioinformatika možná vděčí i tomu, že základy oboru byly položeny v době, která se dnes může jevit jako idylická a idealistická. Vědecké úsilí – s výjimkou vojenského výzkumu a bezprostředně komerčních aplikací – bylo ještě počátkem 80. let minulého století vnímáno jako záležitost především veřejná (a bylo do značné míry financováno z veřejných rozpočtů nebo aspoň z neziskových zdrojů) a měřítkem kvality výzkumu byly publikace, nikoli patenty. Neochota volně sdílet primární data, na nichž publikace stojí, mohla svědčit pouze o pochybné kvalitě oněch dat, a i týmu, který je vyprodukoval, nikoli snad o tom, že si firma, která výzkum financuje, chce udržet kontrolu nad oběhem výsledků, které by mohly mít komerčně zajímavý dopad. Asi jen díky této atmosféře bylo vůbec možné, že se volné sdílení primárních sekvenčních dat stalo standardní praxí, od níž snad již nelze ustoupit. Významnější časopisy dnes vyžadují, aby autoři, kteří publikují sekvenční data, tato data volně zpřístupnili odborné veřejnosti.[1]

3 Definice

Jak již bylo řečeno bioinformatika, jakožto nová vědní disciplína, nemá jednu danou definici, proto zde bude uvedeno pár nejvýstižnějších.

Definice 1

Bioinformatika je nová disciplína na rozhraní počítačových věd, informačních technologií, matematiky a biologie.[3]

Definice 2- „klasická bioinformatika“

Oblast na pomezí biologie a informatiky, která se zabývá především zpracováváním, prohledáváním a analýzou dat o sekvenci (pořadí monomerů) a struktuře biologických makromolekul.[1]

Definice 3

Bioinformatika zkoumá metody shromaždováním analýzy a vizualizace hromadných biologických dat., hlavně z oblasti molekulární biologie.[2]

Definice 4 – širší pojetí

Využití počítačů k hledání odpovědí na biologické otázky.[1]

Bioinformatika zahrnuje

- studium,
- praktické uchovávání,
- vyhledávání,
- zobrazování,
- manipulaci,
- a modelování biologických dat[4]

Bioinformatika představuje spojení technologií z oblastí

- molekulární biologie a
- informačních technologií. [4]

Pod pojmem biologické daty si lze představit následující výčet: [9]

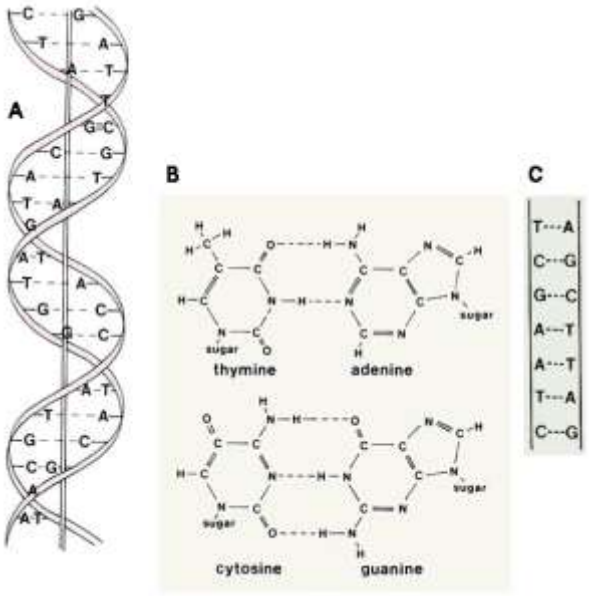
- Sekvence DNA a RNA
- Sekvence proteinů
- Struktura proteinů
- Údaje o aktivitě genů – DNA čip, „microarray“
- Údaje o expresi proteinů
- Mapy interakcí mezi proteiny a DNA
- Mapy interakcí mezi proteiny navzájem

4 Základní pojmy z molekulární biologie

V této kapitole bude čtenář seznámen se základními pojmy z oblasti molekulární biologie. Protože v bioinformatice se pracuje převážně s daty, týkajícími se DNA nebo proteinů, jsou zde pouze uvedeny základní informace z této oblasti. Bez znalosti těchto pojmů je velmi složité porozumět bioinformatickým textům.

4.1 Struktura DNA

Molekula DNA má tvar dvojité pravotočivé šroubovice, tzn. šroubovice, která je tvořena dvěma vlákny. Každé vlákno tvoří sekvence nukleotidů – jde tedy o polynukleotidový řetězec.



Obrázek 1 DNA, zdroj [10]

Nukleotidy tvoří základní stavební bloky DNA. V DNA se mohou vyskytovat pouze 4 druhy nukleotidů: guanin (G), adenin (A), thymin (T) a cytosin (C). (viz obr. 1) Genetická informace, kterou DNA nese, je určena právě a pouze pořadím těchto nukleotidů v řetězci. Pořadí nukleotidů se označuje jako primární struktura DNA a má zásadní význam pro přenos genetické informace. Polynukleotidový řetězec má dva různé konce, které se označují jako 5´konec a 3´konec. Označení konců řetězce je velmi významné, protože většina procesů na těchto řetězcích probíhá od 5´konce ke 3´konci. Ve stejném pořadí se obvykle i zapisují a čtou řetězce nukleotidů. Jak již bylo uvedeno, molekula DNA je tvořena 2 polynukleotidovými řetězci. Pořadí nukleotidů na obou vláknech není stejné, ale je komplementární. Vzhledem k tomu, že z posloupnosti nukleotidů na jednom vlákně můžeme odvodit posloupnost nukleotidů na druhém vlákně, můžeme informaci obsaženou v molekule DNA označit jako redundantní. V praktické BI se druhé vlákno odvozuje od prvního. [2]

Molekula DNA má schopnost replikace, tzn. z jedné mateřské molekuly DNA mohou vzniknout dvě dceřiné molekuly DNA. Replikace probíhá tak, že na určitém místě dojde k uvolnění obou vláken původní molekuly a na ně se naváží nové nukleotidy (na základě komplementarity). Tím vzniknou 2 dceřiné dvouřetězcové molekuly DNA. Jednotka odpovědná za vznik dědičné vlastnosti se označuje jako gen. Touto jednotkou je souvislý úsek molekuly DNA. Geny se mohou vyskytovat na obou vláknech DNA. Soubor všech genů organismu se označuje jako genom. Lidský genom obsahuje přibližně 30 000 genů a tvoří ho asi 3*109 bází. Kromě kódujících oblastí (genů) se v DNA vyskytují i nekódující oblasti. Funkce těchto oblastí ještě není zcela známá, některé úseky mají význam pro buněčnou regulaci (ovlivňují tvorbu proteinů). DNA je uložená v jádře buňky a je rozdělená na několik chromozómů.[2]

4.2 Centrální dogma molekulární biologie

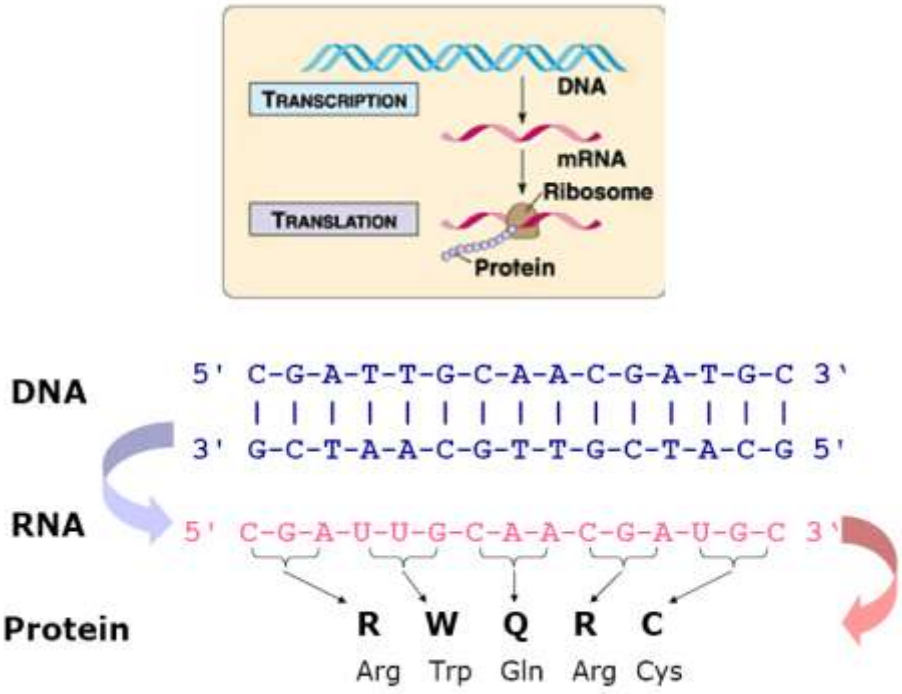
K tomu, aby mohlo dojít k projevení genetické informace uložené v DNA, musí dojít k expresi genu. Při expresi genů dochází k vytváření proteinů, které umožňují projevení informace uložené v DNA.(viz obr. 2) Proces, při kterém dochází k vyzvednutí informace uložené v sekvenci nukleotidů a k vytvoření proteinů (bílkovin, peptidů) na základě této informace, se označuje jako centrální dogma molekulární biologie. Tento proces se skládá ze dvou kroků: transkripce a translace. [2]

4.2.1 Transkripce

Transkripce označuje krok, kdy dochází k vytvoření kopie genu do molekuly RNA (ribonukleová kyselina) za působení enzymu RNA-polymeráza. RNA má podobnou strukturu jako DNA. Molekula RNA je však pouze jednovláknová a obvykle mnohem kratší než molekula DNA. Navíc místo nukleotidu thymin se v RNA vyskytuje nukleotid uracil. Při transkripci se dočasně uvolní vlákna DNA a podle matice se na základě komplementarity vytvoří molekula RNA. Hotové vlákno RNA se uvolní z matice a vycestuje z jádra buňky. Většina genů je rozdělena do několika úseků (exony), které jsou odděleny nekódujícími oblastmi (introny). Při transkripci se vytvořená RNA ještě upraví vystřižením intronů (splicing). Vznikne tak funkční molekula RNA.[2] (viz obr. 2)

4.2.2 Translace

Proces, při kterém dochází k převedení informace ze sekvence nukleotidů v RNA do sekvence aminokyselin, která tvoří protein, se označuje jako translace. Molekula RNA, která vznikla při transkripci, se připojuje na ribozomy. K jednotlivým kodonům (trojice po sobě následujících nukleotidů) této RNA se napojuji molekuly mRNA. mRNA jsou malé molekuly RNA, které slouží k transportu aminokyselin v buňce. Aminokyseliny nesené na mRNA se naváží na vznikající peptidický řetězec a molekula mRNA se uvolní zpět do cytoplazmy. Primární struktura RNA určuje primární strukturu bílkovin (pořadí aminokyselin). [2](viz obr. 2)



Obrázek 2 Centrální dogma molekulární biologie, zdroj [2]

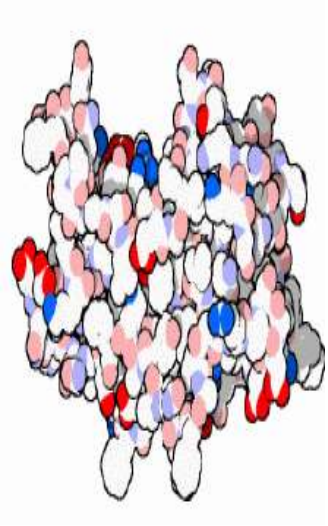
4.3 Genetický kód

Během translace se využívá tzv. genetický kód. Tři po sobě následující nukleotidy RNA (kodón, triplet) kódují jednu aminokyselinu. V RNA se mohou vyskytovat pouze 4 různé nukleotidy, tzn. existuje celkem 4^3 = 64 různých kodónů. V přírodě se vyskytuje pouze 20 různých aminokyselin, které mohou být využity k tvorbě bílkoviny. Z uvedených čísel vyplývá, že jedna aminokyselina může být kódována několika kodóny. Ve skutečnosti pro některé aminokyseliny existuje více kódů než pro jiné aminokyseliny. Genetický kód je tedy redundantní. Některé kodóny mají speciální význam. Jsou to tzv. start a stop kodóny, které označují začátek a konec peptidického řetězce. [2]

4.4 Struktura a funkce proteinů

Proteiny představují molekulární aparát, který řídí a vykonává téměř všechny biologické funkce. Některé proteiny podporují a posilují naše pojivové tkáně, jiné proteiny obsažené ve svalech umožňují pohyb. Další proteiny (enzymy) katalyzují nejrůznější chemické reakce, umožňují a kontrolují trávení a metabolismus, reprodukci a umožňují činnost imunitního systému. Interakce proteinů s RNA a DNA umožňují tvorbu nových proteinů a regulují jejich hladinu v závislosti na změnách vnitřního a vnějšího prostředí. Proteiny jsou syntetizovány jako lineární řetězec aminokyselin, ale v buněčném prostředí rychle vytvoří kompaktní kulovitý (globulární) tvar (protein folding). Tento tvar představuje přirozenou strukturu proteinu, která je nezbytná pro biologickou

funkci proteinu. Pouze v této přirozené globulární struktuře může většina proteinů zcela plnit svoji biologickou funkci. [2] (viz obr. 3)



Obrázek 3 Globulární tvar proteinu, zdroj [3]

4.5 Aminokyseliny

Aminokyseliny tvoří základní stavební bloky proteinů. Na rozdíl od DNA a RNA, které jsou tvořeny ze 4 různých nukleotidů, proteiny jsou tvořeny z 20 aminokyselin, které mají různou velikost, tvar a chemické vlastnosti. Všechny aminokyseliny mají stejnou kostru, kterou tvoří aminoskupina, alfa uhlík a karboxylová skupina. Schopnost jednotlivých aminokyselin vytvářet různé vazby (vodíkové, disulfidické) s jinými aminokyselinami nebo okolními molekulami má zásadní vliv na prostorovou strukturu proteinu.(viz obr. 2) Fyzikální a chemické vlastnosti proteinu, jeho biologická funkce a prostorový tvar jsou určeny pouze sekvencí aminokyselin, které protein vytváří. Primární struktura proteinů se zapisuje jako řetězec znaků - posloupnost jednoznakových kódů.[2](viz tab. 1, 2)

Tabulka 1 Kódy pro zápis sekvencí nukleotidů a aminokyselin, zdroj [1]

DNA, RNA	
Kód	Báze
A	adenin
C	cystosin
G	guanin
T	thymin
U	uracil
R	A,G(purin)
Y	C,T(pirimidin)
S	G,C(strong)
W	A,T (teak)
K	G,T (keto)
M	A,C (amino)
B	C,G,T (not A)
D	A,G,T (not C)
H	A,C,T (not G)
V	A,C,G (not
N	cokoli(any)

Protein	
kód	aminokyselina
A	alanin
B	asparát/asparagin
C	cystin
D	aspartát
E	glutamát
F	fenylalanin
G	glycin
H	histidine
I	isoleucin
K	lysin
L	leucin
M	metionin
N	asparagin
P	prolin
Q	glutamin
R	arginin
S	serin
T	threonin
U	selenocystein
V	valin
W	tryptophan
Y	tyrosin
X	cokoliv
Z	glutamát/glutamin
*	translační stop
-	Mezera neurčená délky

4.6 *Polypeptidy*

Řetězec aminokyselin se nazývá peptid. Delší řetězce se často označují jako polypeptidy nebo proteiny. Při kovalentním spojení (druh chemické vazby) dvou aminokyselin dochází ke vzniku dipeptidu (2 aminokyseliny spojené peptidickou vazbou), a dalších produktů. U molekul proteinů se podobně jako u molekul DNA a RNA rozlišují 2 různé konce řetězce – N-konec (aminokyselina s volnou aminoskupinou) a C-konec (aminokyselina s volnou karboxylovou skupinou). Posloupnost aminokyselin v řetězci se obvykle uvádí od N-konce směrem k C-konci.[2]

5 *Veřejně dostupné zdroje dat*

V této kapitole budou vyjmenovány některé databáze biologických dat: Jelikož v poslední době došlo k velkému rozvoji v oblasti molekulární biologie a genomického[1] výzkumu, který produkuje obrovské množství dat a tudíž i velké množství databází (až několik set), vyvstala potřeba nějak tyto data schraňovat. Proto začaly vznikat databáze. (viz kap. 2) Zde budou vyjmenovány pouze ty významné, nejvíce navštěvované.

K nejdůležitějším institucím zabývajícím se správou biologických dat a vývojem nástrojů pro jejich analýzu a poskytováním informací patří:[3]

- Evropský institut pro bioinformatiku (EBI) se sídlem v Hinxtonu ve Velké Británii

[1] genomika je specializovaný vědní obor, který usiluje o komplexní a úplnou identifikaci a analýzu dědičné informace organismu (genomu)

- Národní centrum pro biotechnologické informace (NCBI) založené původně v rámci Národní lékařské knihovny (NLM) v USA
- Centrum pro informační biologii (CIB) založené jako oddělení Národního genetického institutu (NIG) v Mishimě, Japonsko

Aktuálnost dat v databázích je zajištěna systémem, který neumožní vědcům publikovat výsledky jejich výzkumu dříve, než data, kterých se výzkum týká, nejsou zveřejněna v příslušných databázích. Databáze biologických dat můžeme podle obsažených informací rozdělit do následujících skupin:[2]

Primární:

- databáze sekvencí nukleotidů (EMBL, GenBank, DDBJ) neboli „Velká trojka“(viz kap. 5.1)

Druhotné:

- databáze sekvencí proteinů (Swissprot, TrEMBL)
- databáze struktur proteinů (PDB, MSD)
- genomové databáze (TIGR)
- databáze obsahující informace o expresi genů (ArrayExpress) (viz kap. 5.2)

Kromě těchto základních a poměrně obsáhlých databází existuje celá řada dalších databází biologických dat, jak již bylo řečeno v úvodu. Všechny databáze zahrnují kromě vlastních dat i další informace týkající se např. struktury genů, funkcí proteinů. Všechny informace jsou integrovány z mnoha různých zdrojů. Mezi jednotlivými databázemi existuje velké množství odkazů, což umožňuje snadné hledání vztahů mezi molekulami různých typů.[3]

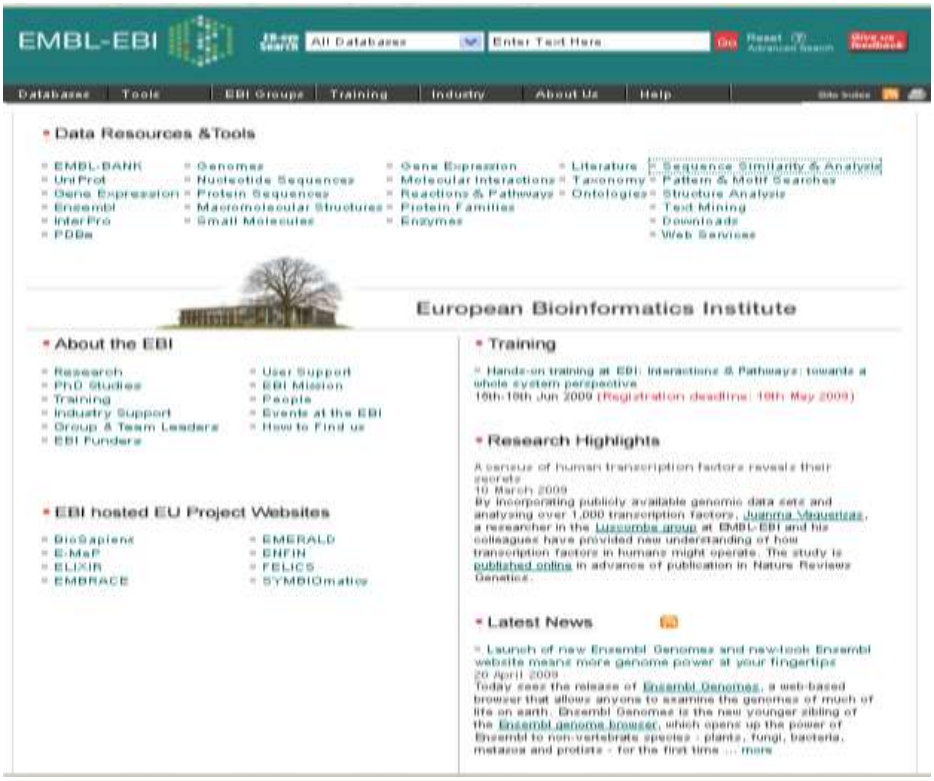
Dále se rozlišují databáze moderované a nemoderované. Do nemoderovaných databází může přispívat kdokoli – jedinou podmínkou je, že data jsou vhodného typu a ve správném strojově čitelném formátu. Správce nemoderované databáze do dat nijak nezasahuje ani nekontroluje kvalifikaci a kompetenci přispěvatele. A naopak moderované databáze správce přijímají data podle výběrových kritérií a kontrolují správnost záznamu. [1]

5.1 Primární databáze sekvencí nukleových kyselin

V každém ze tří hlavních bioinformatických center (viz kap. 5) je spravována genomová databáze biologických dat, hlavně nukleových kyselin:[4]

- EMBL Nucleotide Sequence Database (v rámci institutu EBI) – vytvořena v roce 1980 jako první databanka na světě schraňující nukleotidové sekvence, cílem vytvoření databáze

bylo přimět autory ukládat sekvence přímo v databázi namísto publikování v časopisech, ze kterých byly původně získávány (prvních 2427 sekvencí bylo převedeno do databáze z časopisů), k 20. 04. 2009 obsahuje 248,327,537,740 sekvencí a 156,591,695 nukleotidových bází. [6] (viz obr. 4)



Obrázek 4 Webové stánky databáze EMBL, zdroj [6]

- GenBank (v rámci institutu NCBI) – založeno v roce 1988 jako národní zdroj informací pro molekulární biologii, NCBI vytváří veřejné databáze, provádí výzkum ve výpočetní biologii, vyvíjí softwarové nástroje pro analýzu genomových dat, a šíří biomedicínské informace – vše pro lepší pochopení molekulárních procesů ovlivňujících lidské zdraví a choroby. [5] (viz obr. 5)

Obrázek 5 Webové stránky databáze GenBank, zdroj [5]

- DDBJ (The DNA Data Bank of Japan) - činnost zahájila v roce 1984, shromažďuje data především z japonských výzkumů a úzce spolupracuje s ostatními databázemi[4](viz obr. 6)

Obrázek 6 Webové stránky databáze DDBJ, zdroj [8]

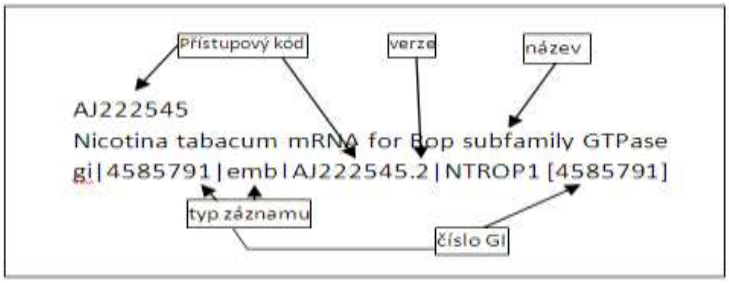
Tyto tři databáze Velké trojky si denně vyměňují veškeré změny v datech a prakticky se neustále vzájemně zálohují. Jejich obsah je tedy téměř totožný až na poslední přírůstky několika hodin. Avšak nevýhoda obrovského objemu dat procházejících databázemi Velké trojky je ta, že databáze nutně musejí být nemoderované, krom toho tyto databáze fungují jako depozitář (repository) nukleotidových sekvencí, což znamená, že právo upravovat záznamy v databázi nebo dokonce záznam odstranit má pouze původní autor, a to i v případě, že bylo prokázáno, že data jsou mylná.[1]

Primární databáze jsou přístupné prostřednictvím webových rozhraní (viz kap. 10), kromě toho lze prostřednictvím FTP bezplatně získat a lokálně instalovat celou databázi ve stavu odpovídajícím určitému datu. Změny v databázi od posledního vydání jsou označovány jako přírůstky (updates).[1]

5.1.1 Orientace v primárních databázích

Orientaci v primárních databázích umožňuje jednotný systém jedinečných identifikátorů databázových záznamů, sdílený mezi všemi třemi databázemi Velké trojky.[1]

Každý datový záznam obdrží okamžitě po zařazení do databáze tzv. přístupový kód, který se skládá z proměnlivého počtu písmen a číslic (podle toho kdy a přes kterou primární databázi byl záznam přijat). Přístupový kód je jakýmsi ekvivalentem rodného čísla záznamu, zůstává nezměněn po celou dobu jeho existence a umožňuje kdykoliv příslušný databázový záznam vyhledat. Současně s publikací v GenBank pak záznam získá rovněž jedinečné číslo GI (GenBank Identifier). Mezi přístupovým kódem a číslem GI není zjevný vztah. [1]



Obrázek 7 Příklad identifikátoru databázového záznamu formát GenBank, zdroj [1]

„Hlavička“ záznamů v databázi GenBank obsahuje také informaci o typu dat a cestě, kudy se do databáze dostala – tak emb znamená záznam původem z EMBL, gb GenBank, dbj záznam z DDBJ, dbest záznam typu EST a podobně. Původ dat je obdobě vyznačen i u odvozených databází proteinových, které bývají pro účely prohledávání často spojovány i s jinými specializovanými databázemi, nepř. Swissprot či PIR. (viz kap. 5.2) [1]

5.1.2 Typy záznamů v primárních databázích

V primárních databázích se lze setkat s několika typy datových záznamů. Následující výčet shrnuje ty nejběžnější: [1]

- standardní originální nukleotidové sekvence získané pokud možno kvalitním sekvenováním fragmentů genomové DNA
- EST (expressed sequence tags) – částečné sekvence, které jsou obvykle nižší kvality než standardní sekvence

- dosud neposkládané a neanotované „surové“ sekvence ze sekvenování genomu (HTGS – high throughput genome sequencing)
- referenční sekvence již poskládaných a více či méně anotovaných kompletních genomů
- sekvence anotované jinými než původními autory (TPA – third party annotation)

5.2 Druhotné databáze

Tyto druhotné databáze jsou proteinového charakteru, tedy obsahují většinou aminokyselinové sekvence, globulární tvar proteinu (viz kap. 4.4), funkce proteinu, odkaz na literaturu a další.

5.2.1 Databáze sekvencí proteinů

Mezi hlavní databáze proteinových sekvencí patří databáze PIR (Protein Information Resource), MIPS (Martinsried Institute for Protein Sequences), Swissprot a TrEMBL. Snahou databáze Swissprot je poskytnout kromě vlastní sekvence i další kvalitní anotaci, která zahrnuje popis funkcí proteinů, strukturu jeho domén, post-translační modifikace atd. Díky této anotaci, která je na vysoké úrovni, a díky jednoduché struktuře se tato databáze stala nejčastěji využívanou databází proteinových sekvencí. Databáze Swissprot se stejně jako databáze EMBL skládá ze záznamů o sekvencích. Formát databáze je navržen tak, aby byl co nejvíce podobný formátu databáze EMBL.(viz kap.6.1) Každý záznam obsahuje tato data:[2]

- vlastní sekvence aminokyselin
- informace o citacích v literatuře
- taxonomická data (informace o organismu, z kterého byla sekvence získána.)

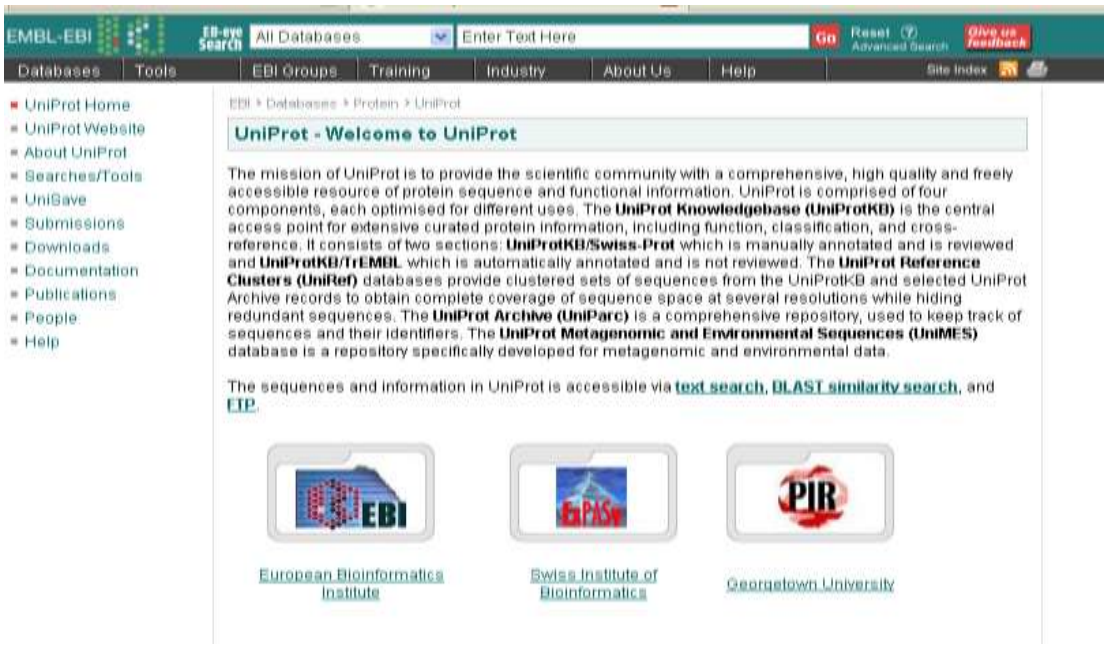
Kromě těchto základních dat mohou záznamy obsahovat informace o:[2]

- funkci proteinu
- post-translačních modifikacích
- zajímavých oblastech nebo místech sekvence (např. vazební místa)
- sekundární struktuře sekvence
- podobnosti s dalšími proteiny
- chorobách spojených s odchylkami proteinu
- variantách sekvence

Databáze TrEMBL obsahuje hypotetické proteinové sekvence, které byly odvozené z kódujících sekvencí uložených v databázích Velké trojky nebo získané z literatury, a ještě nebyly integrované do databáze Swissprot. Většina záznamů v TrEMBL byla vytvořena počítačem, přeložením sekvencí uložených v databázích Velké trojky na základě genetického kódu. Společně se sekvencí nukleotidů jsou v záznamech veškeré informace, které lze odvodit ze záznamů o sekvencích nukleotidů. Záznamy

zůstávají v databázi TrEMBL, dokud nejsou manuálně popsány a integrovány do databáze Swissprot.[2]

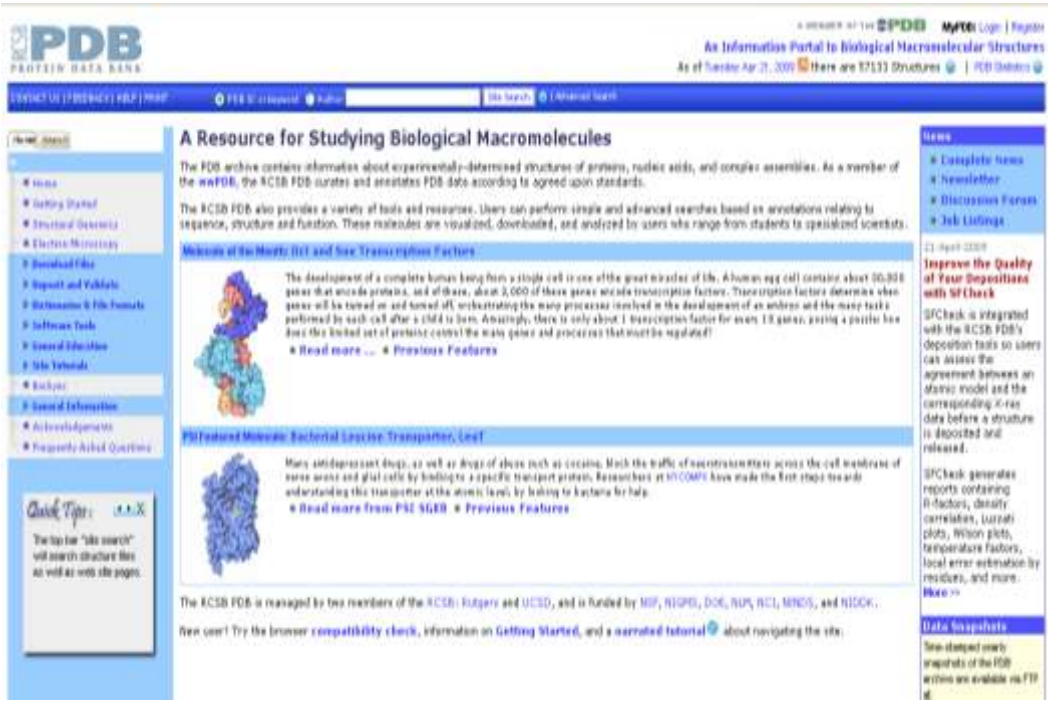
V roce 2002 se Swissprot, TrEMBL a americká databáze PIR propojily v rámci iniciativy Uniprot a sdílejí nyní data podobným způsobem jako primární databáze Velké trojky.(viz obr. 8)[1]



Obrázek 8 Webové stránky konsorcia Uniprot, zdroj [6]

5.2.2 Databáze struktur proteinů

Největší databáze, která obsahuje informace o 3D struktuře proteinů je databáze PDB (Protein Data Base).(viz obr. 9) Databáze je složena ze záznamů o jednotlivých proteinech, které mají podobu textového souboru. Každý záznam může kromě prostorových souřadnic jednotlivých atomů (*x*, *y*, *z*) proteinu obsahovat i další informace o proteinu. Mohou zde být informace o sekundární struktuře proteinu,(viz kap. 4.4) o aktivních místech proteinu, o disulfidických vazbách, které se v proteinu nachází, o nestandardních (modifikovaných) aminokyselinách, které se v proteinu nachází, o tom, jakou metodou byla prostorová struktura proteinu zjištěna. Dále může záznam obsahovat informace o organismu, z něhož byl protein izolován, a odkazy na záznamy v dalších databázích biologických dat, které s daným proteinem souvisí. Existují programy, které umožňují zobrazit informace obsažené v PDB souborech v grafické formě. Tyto programy umožňují zobrazení modelu proteinu a jeho prohlížení z různých úhlů.[2]



Obrázek 9 Databáze 3D struktury proteinů PDB, zdroj [10]

5.2.3 Geonomové databáze

Genomové databáze vznikly jako odpověď na rychle se množící projekty, které se zabývají sekvenováním celých genomů (přečtení všech bází genomu). Cílem těchto databází je posbírat dohromady všechny informace o genomech jednotlivých organismů, které budou jasně odděleny od informací o genomech jiných organismů. Díky těmto databázím je možné snadněji analyzovat všechny geny určitého organismu. Tyto databáze opět obsahují informace o sekvencích nukleotidů jednotlivých genů, ale tyto sekvence jsou doplněny informacemi o poloze genů na jednotlivých chromozómech. Genomické databáze obvykle poskytují možnost zobrazování v několika úrovních (od zobrazení celých chromozómů po zobrazení sekvencí jednotlivých genů).[2] Ale asi nejvýznamnějším přínosem jsou přehledné grafické mapy genomů, ve kterých se snadno vyhledává jako příklad databáze, jejíž obsah překračuje pouhý uspořádaný výčet sekvencí, lze uvést databázovou sekci webových stránek Venterova Ústavu pro výzkum genomů TIGR (The Institude for Genomic Research) (viz obr. 10) s podsekcemi věnovanými především genomům prokaryontních[2] mikroorganismů, ale i genomových či EST projektům některých eukaryot[3].[1]

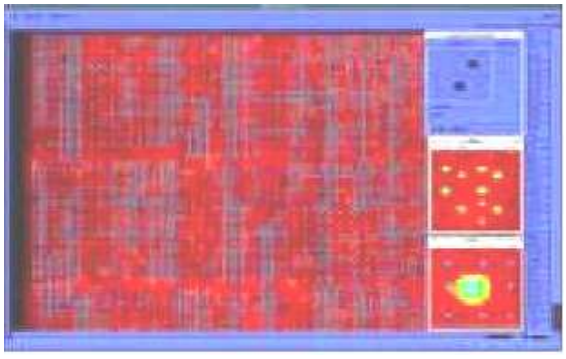
[2] Prokaryota-bakterie, rozmnožující se nepohlavně

[3] Eukaryota-velká skupina jednobuněčných a mnohobuněčných organizmů, jako jsou například živočichové, rostliny, houby nebo prvoci

Obrázek 10 Webové stránky Venterova Ústavu pro výzkum genomů TIGR, zdroj [11]

5.2.4 Databáze obsahující informace o expresi genů

Úroveň exprese jednotlivých genů v tkáních se určuje pomocí mikročipů. Mikročipy jsou destičky se sondami, na které se váže mRNA. Sondy jsou uspořádány do matice a tvoří je polynukleotidy charakteristické pro jednotlivé geny. Na tyto mikročipy se nanášejí vzorky obarvených tkání, mRNA obsažená v tkáních se váže na sondy a pomocí laseru se určuje intenzita jednotlivých bodů matice sond. Intenzita bodů určuje, kolik mRNA se na jednotlivá místa matice navázalo. Velice často se mikročipy používají pro porovnání genové exprese dvou tkání (zdravé a nemocné). V těchto případech se na mikročip nanesou vzorky obou tkání, které jsou různě zbarvené. (viz obr. 11) Informace z těchto experimentů se uchovávají ve specializovaných databázích. Tyto databáze potom obsahují informace o expresi genů, které jsou uložené v rozsáhlých hustě zaplněných maticích reálných čísel. Tyto matice mohou mít několik tisíc řádků a sloupců. Vzhledem k množství uložených informací představují tyto databáze druhý největší informační zdroj, hned po databázích sekvencí nukleotidů a proteinů



Obrázek 11 Ukázka microarray čipů, zdroj [4]

6 Práce se sekvenčními daty

V této kapitole bude uvedeno, jak vypadá porovnávání dvou sekvencí a zjišťování míry jejich podobnosti. Toto porovnávání je centrálním tématem praktické bioinformatiky. Můžou se porovnávat, jak nukleoidové sekvence, tak proteinové, z toho důvodu, že je potřeba:[1]

- hledat homology[4] oblastí genomu nekódujících proteiny,
- vyhledávat sekvence DNA identických či téměř identických dotazu. Může to být třeba proto, že je potřeba zjistit, zda už zkoumaný fragment někdo nesekvenoval,
- zabývat se evoluční historií genů kódujících proteiny a další.

Nyní je potřeba si říct, co jsou to sekvenční data. Sekvenční data jsou jakékoli formy zápisu lineární posloupnosti monomerů v molekule biologické makromolekuly - nejčastěji DNA nebo proteinu. Do práce se sekvenčními daty především patří:[3]

- úpravy zápisu sekvencí do podoby srozumitelné programům pro další analýzy nebo do úhledné podoby pro prezentaci,
- odvození komplementárního vlákna DNA, (viz. Kap. 4.2)
- identifikace kódujících oblastí a překlad DNA sekvence,
- tvorba grafických map ze sekvence.

Dále je potřeba pochopit, co je míněno pod pojmem sekvence. Sekvence je digitální zápis posloupnosti jednoznačných znaků odpovídajících jednotlivým typům monomerů po řadě tak, jak se s nimi v molekule setkáváme při postupu ve směru odpovídajícím směru biosyntézy daného typu molekuly. Pro DNA či RNA je to od 5´ k 3´ konci, pro proteiny od N-konce k C-konci. [1] (viz kap 4.1 a kap. 4.6)

6.1 Zápis sekvencí a běžné formáty datových souborů

V nejjednodušší digitální podobě je sekvence zapsána jako prostý řetězec IUPAC[5] znaků, které jsou současně znaky ASCII(viz tab. 1 a tab. 2), v textovém souboru. Tento formát označujeme jako surová data. [1]Nutno podotknout, že tento formát umí přijímat většina online programů. (viz kap. 10.3) A však není v této „surové“ formě uvedeno nic o původu sekvence.

Proto pro uchovávání sekvenčních dat spolu s dalšími průvodními informacemi byla vyvinuta celá řada formátů. Některý program může být naprogramován na příjem sekvencí v určitém typu

[4] Homolog souhlasnost, shodnost; vztah mezi členy řady sloučenin, jejichž složení se vzájemně liší o jednu nebo více stejných stavebních jednotek

[5] IUPAC je nejvyšší mezinárodní autoritou v oboru chemického názvosloví

formátu. Nejběžnější a nejpoužívanější je však formát FASTA, nazvaný podle stejnojmenného algoritmu. (viz kap. 7.2.1)

6.1.1 Formát FASTA

Vhodný jak pro sekvence nukleotidové, tak i pro sekvence aminokyselin. Je to textový soubor, jehož první řádka začíná znakem > (větší než), za nímž následuje „hlavička“ obsahující název sekvence, anotaci a případné další údaje, které nejsou součástí sekvence samotné. První skupina alfanumerických znaků v hlavičce (až po první mezeru) představuje jedinečný identifikátor sekvence (je vhodné sem umisťovat např. název genu či klonu), na zbytek můžeme pohlížet jako na volitelný komentář. Na dalších řádkách pak následuje surová sekvence. (bez mezer a cizích znaků). Neřeší však určení typu sekvence.[1]

Příklad formátu FASTA dle [4] je např.:

```
>sekvence187 [org=Staphylococcus phage 187] ORF1(partial)
CTGAGATATAAAAGCGTATAGAATATTCGAAGTAGTATAGTGTAGCCTGCAATAGCATCAGTGTAGTTCTCA
GCATAGAGAGGCTAAGATAAATGAAGATTGTGTCAGCAGCTACAATGTTTAGATATGGGAATGCTTTAAGGC
TGACTCGAAAGGTATAAAGTCGGTAAATAAGTTGCTTGAACACTGTTTTTACACCTGACAAAGNNNNNNNNN
NNNNNNNNNGCATAGTCATTAGTAATGATGGCGGAAAGACCTTT
```

6.1.2 Formát PIR/NBRF

Příkladem poměrně jednoduchého formátu, který elegantně řeší problém určení typu sekvence, je formát PIR/NBRF. Přijímá ho jen málo programů, ale stojí zde za zmínku. První řádka začíná rovněž znakem >, za nímž následuje dvouznakový kód určující typ sekvence (např. P1 je protein, DL lineární DNA – DNA linear), středník a zkrácený název sekvence. Druhá řádka obsahuje plný název a anotaci, od třetí řádky následuje sekvence ukončená hvězdičkou.[1]

Příklad formátu PIR dle [4] je např.:

```
>P1;FOSB_MOUSE
sw:fosb_mouse => FOSB_MOUSE
MFQAFPGDYD SGSRCSSSPS AESQYLSSVD SFGSPPTAAA SQECAGLGEM
PGSFVPTVTA ITTSQDLQWL VQPTLISSMA QSQGQPLASQ PPAVDPYDMP
GTSYSTPGLS AYSTGGASGS GGPSTSTTTS GPVSARPARA RPRRPREETL
TPEEEEKRRV RRERNKLAAA KCRNRRRELT DRLQAETDQL EEEKAELESE
IAELQKEKER LEFVLVAHKP GCKIPYEEGP GPGPLAEVRD LPGSTSAKED
GFGWLLPPPP PPPLPFQSSR DAPPNLTASL FTHSEVQVLG DPFPVVSPSY
TSSFVLTCPE VSAFAGAQRT SGSEQPSDPL NSPSLLAL*
```

6.1.3 Formát GenBank

Jako dalším formátem je GenBank z amerického institutu NCBI. Sekvence je rozdělená do řádků po 60 znacích a mezi každým desátým a jedenáctým znakem je udělaná mezera pro

přehlednost. V hlavičce jsou údaje o původu sekvence, druh sekvence a datum vložení do databáze.

Příklad formátu GenBank dle [4] je např.:

```
LOCUS       sekvence187    262 bp    DNA             UNA       30-Jan-2003
DEFINITION  sekvence187
ORIGIN
        1 CCTGAGATAT AAAAGCGTAT AGAATATTCG AAGTAGTATA GTGTAGCCTG CAATAGCATC
       61 AGTGTAGTTC TCAGCATAGA GAGGCTAAGA TAAATGAAGA TTGTGTCAGC AGCTACAATG
      121 TTTAGATATG GGAATGCTTT AAGGCTGACT CGAAAGGTAT AAAGTCGGTA AATAAGTTGC
      181 TTGAACACTG TTTTTACACC TGACAAAGNN NNNNNNNNNN NNNNNNGCAT AGTCATTAGT
      241 AATGATGGCG GAAAGACCTT TG
```

6.1.4 Formát EMBL

Ne zcela správný formát pro své specifické formátování sekvence. Ale tento formát používají například některé programy jako výstup. (viz kap. 8.2)[1]

Příklad formátu EMBL dle [4] je např.

```
ID  sekvence187 standard; DNA; UNC; 262 BP.
DE  sekvence187
SQ  Sequence 262 BP;
    CCTGAGATAT AAAAGCGTAT AGAATATTCG AAGTAGTATA GTGTAGCCTG CAATAGCATC   60
    AGTGTAGTTC TCAGCATAGA GAGGCTAAGA TAAATGAAGA TTGTGTCAGC AGCTACAATG  120
    TTTAGATATG GGAATGCTTT AAGGCTGACT CGAAAGGTAT AAAGTCGGTA AATAAGTTGC  180
    TTGAACACTG TTTTTACACC TGACAAAGNN NNNNNNNNNN NNNNNNGCAT AGTCATTAGT  240
    AATGATGGCG GAAAGACCTT TG                                           262
```

6.1.5 Formát CLUSTAL

Pro mnohočetná přiřazení se vžil formát CLUSTAL podle stejnojmenného programu. (viz kap. 9) Od prostého zápisu se liší pouze definovanou délkou názvu sekvencí a rozestupem řádků, hlavičkou (která musí začínat slovem CLUSTAL a může pokračovat krátkým komentářem do jednoho řádku). [1]

Příklad formátu CLUSTAL dle [4] je např.:

```
CLUSTAL W (1.8) multiple sequence alignment
sekvence187 CTGAGATATAAAAGCGTATAGAATATTCGAAGT-GTATAGTGTAGCCTGCAATAGCATCA
sekvence289 ATCAGAGAT-AAAGCGGATAGAATATTCG-AGTAGTATAGTGTAGCCTGCAATAGCATCA
sekvence187 GTGTAGTTCTCAGCATAGAGAGGCTAAGATAAATGAAGATTGTGTCAGCAGCTACAATGT
sekvence289 GTGTAGTTCTCAGCATAGAGAGGCTAAGATAAATGAAGATTGTGTCAGCAGCTACAATGT
sekvence187 TTAGATATGGGAATGCTTTAAGGCTGACTCGAAAGGTATAAAGTCGGTAAATAAGTTGCT
sekvence289 TTAGATATGGGAATGCTTTAAGGCTGACTCGAAAGGTATAAAGTCGGTAAATAAGTTGCT
sekvence187 TGAACACTGTTTTTACACCTGACAAAGNNNNNNNNNNNNNNNNNNGCATAGTCATTAGTA
sekvence289 TGAACACTGTTTTTACACCTGACAAAGNNNNNNNNNNNNNNNNNNGCATAGTCATTAGTA
sekvence187 ATGATGGCGGAAAGACCTTTG
sekvence289 ATGATGGCGGAAAGACCTTTG
```

6.1.6 Formát PHYLIP

Na rozdíl od předchozího formátu CLUSTAL tento programy, které soubory ve formátu PHYLIP přijímají, kontrolují správnost uvedených údajů o počtu a délce sekvencí.[1]

Příklad formátu PHYLIP dle [4] je např.:

```
     3 100
ABC-1 ---ATTGCGT TATGGAAATT CGAAACTGCC AAATACTATG TCACCATCAT
ABC-2 GATATTGCTT TATGGAAATT CGAAACTGCC AAATACTATG TCACCATCAT
ABC-3 ---ATTGCTT TATGGAAATT CGAAACTGCC AAATACTATG TTA-------

       TGATGCACCT GGACACAGAG ATTTCATCAA GAACATGATC ACTGGTACTT
       TGATGCACCT GGACACAGAA ATTTCATCAA GAACATGATC ACTGGTACTT
       TGATGCACCT GGACACAGAG ATTTCATCAA AAACATGATC ACTGGTACTT

>[org=Saccharomyces cerevisiae][strain=ABC][clone=1]
>[org=Saccharomyces cerevisiae][strain=ABC][clone=2]
>[org=Saccharomyces cerevisiae][strain=ABC][clone=3]
```

Poněkud méně přehlednou, ale v některých situacích výhodnější variantou mnohočetného přiřazení právě uvedených „prostřídaných“ formátů, jsou formáty, v nichž jsou jednotlivé sekvence i s mezerami uváděny odděleně za sebou. Takový zápis totiž usnadňuje např. výběr pouze některých sekvencí pro další zpracování nebo postupné přidávání sekvencí do přiřazení. Asi nejjednodušší z těch to formátů je již zmiňovaný formát FASTA.(viz kap. 7.2.1)[1]

Původní data do primárních databází se můžou uložit, pokud se dodrží formální požadavky na formát datových souborů. Jak EMBL tak GenBank používají stejné vstupní nástroje – webové formuláře, kterými lez odeslat data do databáze. A však tato činnost je však typická spíše úzké aktivní výzkumné komunitě, která se zabývá právě sekvenováním DNA. Mnohem častěji se však v datech vyhledává. Jakým způsobem se hledají podobné sekvence, bude nastíněn v následujících podkapitolách a dále v konkrétních prohledávacích algoritmech v 7. kapitole.[1]

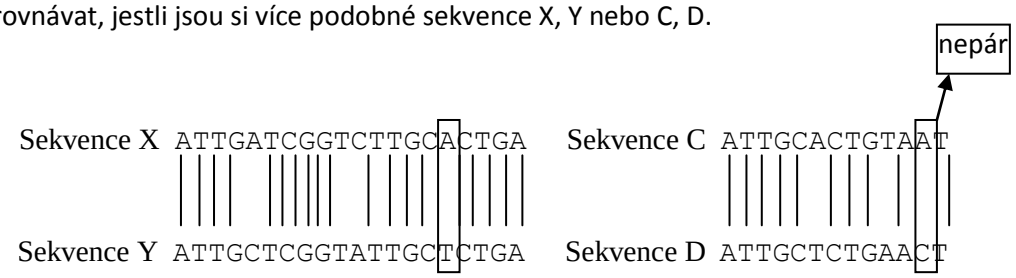
6.2 Postup stanovení podobnosti dvou sekvencí

Obecný postup stanovení míry podobnosti dvou sekvencí je následující:[1]

- Sekvence po celé délce se přiloží k sobě, tj. zapíše se do dvou pod sebou umístěných řádků tak, aby identické pozice (báze či aminokyseliny) ležely pod sebou (tudíž neproporcionální písmo). Takovému zápisu se říká přiřazení (aligment)
- Vypočte se celková hodnota (score) podobnosti tak, že se sečtou hodnoty podobnosti všech jednotlivých pozic přiřazení. Hodnoty podobnosti se stanoví podle předem zvolených kritérií. V nejjednodušším případě se např. může jakékoli identické dvojici pozic (pár – match) přidělit hodnota 1 a jakékoli neidentické dvojici (mismatch) hodnota 0.

- zda-li se porovnává míra podobnosti různě dlouhých dvojic sekvencí, vydělí se celková hodnota podobnosti délkou přiřazení (normalizace hodnoty podobnosti)

Nyní zde bude uveden konkrétní případ zjištění míry podobnosti čtyř sekvencí, bude se porovnávat, jestli jsou si více podobné sekvence X, Y nebo C, D.



Výpočet normalizované hodnoty podobnosti bude stanoven podle předcházejícího výčtu, tedy sekvence se přiloží k sobě, aby identické pozice ležely pod sebou, spočítá se kolik je párů(tedy stejných identických dvojic(písmen) spojených svislou čárou) a spočítá se kolik je neidentických dvojic (nespojených čárou). Páry se vynásobí jedničkou a nepáry nulou. Tyto dvě hodnoty se sečtou, jelikož jsou to nestejně dlouhé sekvence(X a Y sekvence mají délku 20 a sekvence C a D délku 13), vyděli se součet délkou sekvence.

S(X,Y) = (17 x 1 + 3 x 0)/20 = 17 + 3 = 0,85 S(C,D) = (10 x 1 + 3 x 0)/13 = 10 + 3 = 0,77

Výsledek je, že si jsou podobnější sekvence X,Y. Pro výpočet podobnosti nukleových kyselin (hlavně) se též může použít substituční matice. (viz kap. 6.3, obr. 14)

Tento výpočet je celkem snadná záležitost, netriviální se však stává to, pokud budou sekvence buď nestejné nebo i stejné délky, které se budou významně lišit a nebude tak jednoznačné k sobě přiřadit stejné báze nebo aminokyseliny. Vyvstává tu otázka, zda vnášet mezery nebo ne. Je zde na výběr ze dvou typů přiřazení: [1]

- Globální přiřazení – přiřazení sekvencí po celé délce i za cenu vnášení mezer do jedné z nich i do obou; je základem některých metod pro konstrukci mnohočetných přiřazení. (viz kap. 8)
- Lokálního přiřazení – přiřazení jednoznačných úseků a skončilo by se tam, kde se sekvence rozcházejí nad únosnou míru.

Globální přiřazení

```
SLAV----------APATNIK-------PIQNYR-R------AKSETQRYMVIE
SLAVYTYIEFVRANAPATNIKSECVRAAPIQNYRRVEHVRATAKSETQRYMVIE
```

Lokální přiřazení

```
 SLAVYTYIEFVRANAPATNIKSECVRAAPIQNYRRVEHVRATAKSETQRYMVIE
-------------NAPATNIKSECVRA-PIQNYRRVEHVRA--------------
```

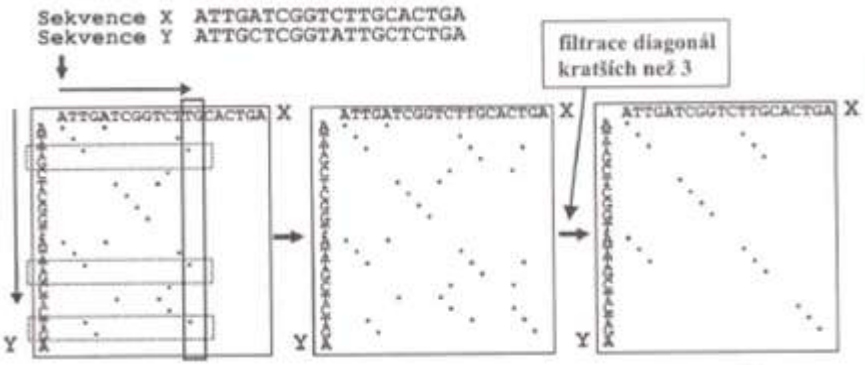
Obrázek 12 Příklad globálního a lokálního přiřazeni, zdroj [1]

Pokoušet se o globální přiřazení má cenu jen u sekvencí, které jsou si vzájemně blízké – tj. nepodlehly v průběhu evoluce přestavbám a lze je vzájemně přiřadit s vnesením poměrně malého počtu nedlouhých mezer. Naproti tomu lokální přiřazení je vhodné i pro sekvence složené ze „stěhovavých“ domén (viz obr. 12). [1]

Dobrou pomůckou pro volbu typu přiřazení, a vůbec pro posouzení, zda danou dvojici sekvencí má smysl přiřazovat, je grafická mapa vzájemné podobnosti sekvencí čili bodový diagram (dotplot). Bodový diagram vyjadřující podobnost dvou sekvencí X a Y se může získat následujícím postupem: [1]

- Souřadnice (pořadová čísla) bází či aminokyselin v sekvencích se vynesou na dvě vzájemně kolmé osy (X na osu x, Y na osu y) tak, že pozici 1 v sekvenci X odpovídá pozice 1 v sekvenci Y
- Vezme se „okénko“ zvolené délky (např. 2 písmena) a posouvá se po sekvenci X od polohy 1 na konec v krocích po 1 písmenu
- Pro každou pozici okénka se udělá tečka všude, kde okénko „našlo“ stejnou dvojici písmen v sekvenci Y
- V následném kroku se můžou pro přehlednost odfiltrovat (tj. smazat) všechny tečky, které nejsou součástí diagonál delších než nějaká předem stanovená minimální hodnota, např. hodnota 3)

  Celý postup je znázorněn na obrázku 13.



Obrázek 13 Konstrukce bodového diagramu, zdroj [1]

Podobnost sekvencí X a Y se v diagramu projeví jako diagonála (přerušovaná v místě nepárů). Z pohledu na diagram se může okamžitě odhalit přítomnost a poloha oblastí vzájemné podobnosti, ale i inzercí, delecí, translokací a opakování v rámci jednotlivých sekvenci. [2]

inserce – vložení jednoho nebo více symbolů

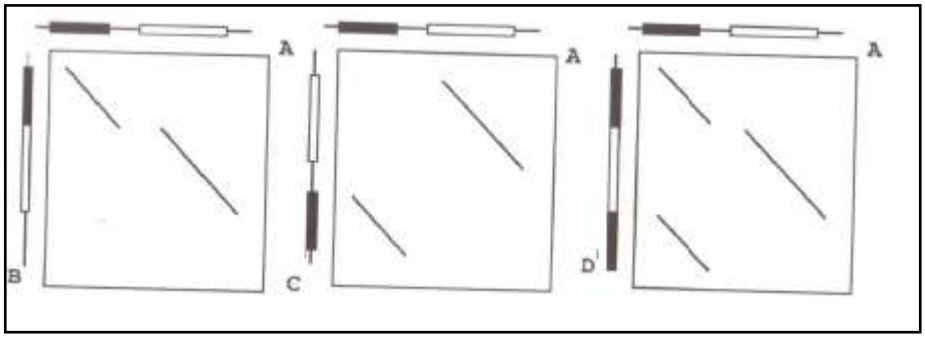
delece – odstranění symbolu z jedné nebo více pozic.

translokace – přehození dvou symbolů

Algoritmy, které se používají pro praktickou konstrukci přiřazení, dovedou vytvořit buď přiřazení lokální, nebo globální. Rozhodnutí mezi lokálním a globálním přiřazením tedy musí uživatel učinit již na úrovni volby programu. U párového přiřazení se prakticky používá pouze přiřazení lokální. Strategii nejjednoduššího možného postupu konstrukce lokálního přiřazení může být zhruba následovná:[1]

- Shora uvedeným postupem se sestrojí bodový diagram (viz obr. 13)
- Zvolí se diagonála, která se má stát jádrem přiřazení
- Tato diagonála se postupně rozšiřuje po 1 zbytku, a v každém kroku se vypočte hodnota podobnosti
- Pokračujeme tak dlouho, dokud s přidáváním dalších zbytků hodnota podobnosti buď roste, nebo klesá méně než o předem stanovený mezní rozdíl.

  Na obrázku 14 je zobrazen příklad bodových diagramů pro sekvence, které prošly přestavbami doménové struktury. Lze na něm vidět, že jen pro dvojici A a B lze sestrojit podle předcházejícího výčtu smysluplné lokální přiřazení.



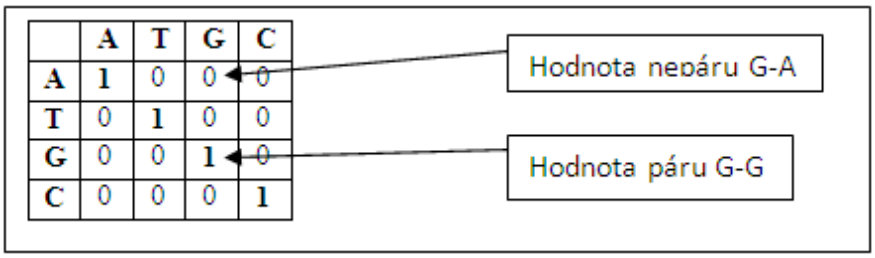
Obrázek 14 Bodové diagramy s určením nejvhodnějšího lokálního přiřazení, zdroj [1]

Praktickým příkladem obecně dostupného programu, který dokáže vytvořit bodový diagram a lokální přiřazení dvou uživatelsky zadaných sekvenci je program BLAST2 a je variantou dnes již klasického prohledávacího postupu BLAST (viz kap. 6.2.2)[1]

6.3 Substituční matice

Volba druhu přiřazení není jediným krokem, kde do hry vstupují uživatelská rozhodnutí. Dojde na ně i při vlastním výpočtu hodnoty podobnosti, kde musí nejprve zvolit „váhu“ všech možných kombinací (párů i nepárů) aminokyselin, popřípadě nukleotidů. Pokud se při konstrukci přiřazení zaváděly mezery, musí se rovněž rozhodnout, jakou „váhu“ či „cenu“ mezerám přisoudit. Od toho se pak bude odvíjet další rozhodování, totiž, zda-li se vůbec zavádění mezer vyplatí, nebo zda raději místo zavedení mezery lokální přiřazení ukončit.[1]

Váhy všech možných párů i nepárů aminokyselin či nukleoidů udává substituční matice (scoring matrix). Substituční matice je čtvercová tabulka, jejíž řádky a sloupce odpovídají jednotlivým symbolů v sekvenci. Číselná hodnota na průsečíku řádku a sloupce odpovídá příspěvku příslušné kombinace symbolů k celkové hodnotě podobnosti. (viz obr. 15) Hodnoty přisouzené jednotlivým nepárům nezávisí na pořadí symbolů (nepár G-A má stejnou hodnotu jako nepár A-G), a matice je tudíž symetrická podle diagonály.[1]



Obrázek 15 Nejjednodušší substituční matice, použitá pro výpočet podobnosti nukleotidových sekvencí, zdroj [1]

Konkrétní hodnoty v matici odrážejí pozorování ze skutečných sekvencí. V případě proteinů jsou to jednak pozorované frekvence záměn konkrétní aminokyseliny v rámci známých souborů homologních sekvencí, jednak frekvence výskytu jednotlivých aminokyselin. Spárovaná vzácná aminokyselina, např. cystein nebo tryptofan, má větší váhu než aminokyselina častá, popřípadě aminokyselina kódovaná mnoha kodony, jako třeba serin. Číselné hodnoty v maticích jsou stanoveny empiricky z „reprezentativních“ či „typických“ souborů sekvencí tak, aby jejich aplikace vedla k výsledkům, které pokud možno dávají smysl. Především pro proteiny neexistuje „jediná správná“ substituční matice. Pro sekvence nukleových kyselin se však prakticky používají varianty jediné matice, tzv. matice IUPAC čili matice identity (identity matrix), která všem párům přiřazuje konstantní kladnou hodnotu a všem nepárům rovněž konstantu (nulovou či zápornou).[1]

U proteinů je situace složitější, protože tam bylo až dosud sestaveno několik sérií matic. Historicky nejstarší, a dosud poměrně často používané, jsou matice PAM (point accepted station), vytvořené Margaret Dayhoffovou a spolupracovníky již na sklonku 70. let minulého století na základě analýzy globálních přiřazení sady blízce příbuzných sekvencí. Konstrukce matice PAM vychází z empirického stanovení frekvence jednotlivých specifických záměn. V rámci modelové sady sekvencí byly pro každou aminokyselinu zjištěny frekvence všech možných záměnových mutací v situaci, kdy mutace postihují 1 % zbytku v sekvenci. Na základě takto změřených hodnot specifických mutačních rychlostí byla sestrojena matice PAM1, vynásobením této matice jí samotnou PAM2, atd. až po PAM250, která odpovídá stavu, kdy na stoaminokyselinový úsek připadá již 250 mutací. Novější obdobou série PAM, sestrojenou z většího souboru výchozích dat, jsou matice Jones-Taylor-Thorntonovy (JTT) a Gonnetovy. Pro číslování Gonnetových matic platí totéž co pro PAM – GONNET250 odpovídá PAM250.[1]

PAM matice pro vzdálenější sekvence (tj. s vyššími čísly) byly tedy získány extrapolací. Druhá často používaná série matic BLOSUM byla naopak odvozena přímo z empirických dat na základě lokálního přiřazení konzervativních domén méně blízce příbuzných sekvencí. Také matice BLOSUM jsou číslovány (viz tab. 2) – čím vyšší číslo, tím vyšší frakce konzervativních zbytků ve výchozím souboru sekvencí. Pro praktickou aplikaci je důležité pamatovat si pravidlo, že čím podobnější jsou si sekvence proteinů, tím nižší hodnoty P'AM (případně vyšší hodnoty BLOSUM) bychom měli volit – a na opak. Matice s nízkým PAM a s vysokým BLOSUM nejsou ale volně zaměnitelné, protože byly odvozeny na základě zcela jiných empirických dat. Používat by se měly pokud možno na data podobná těm původním, proto nízké PAM je preferována v situacích, kde za podobnost „může" krátká evoluční vzdálenost (malý počet mutací) oddělující zkoumané sekvence, kdežto vysoké BLOSUM tam, kde se předpokládá významnou podobnost „ostrůvků" v jinak divergentních sekvencích např. v důsledku selekce či evolučních omezení. V tabulce 2 je příklad reálné substituční matice pro proteiny - BLOSUM80. Kódy aminokyselin v tabulce 1.

Tabulka 2 Příklad reálné substituční matice pro proteiny - BLOSUM80, zdroj [1]

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	B	Z	X	*
A	5	-2	-2	-2	-1	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0	-2	-1	-1	-6
R	-2	6	-1	-2	-4	1	-1	-3	0	-3	-3	2	-2	-4	-2	-1	-1	-4	-3	-3	-2	0	-1	-6
N	-2	-1	6	1	-3	0	-1	-1	0	-4	-4	0	-3	-4	-3	0	0	-4	-3	-4	4	0	-1	-6
D	-2	-2	1	6	-4	-1	1	-2	-2	-4	-5	-1	-4	-4	-2	-1	-1	-6	-4	-4	4	1	-2	-6
C	-1	-4	-3	-4	9	-4	-5	-4	-4	-2	-2	-4	-2	-3	-4	-2	-1	-3	-3	-1	-4	-4	-3	-6
Q	-1	1	0	-1	-4	6	2	-2	1	-3	-3	1	0	-4	-2	0	-1	-3	-2	-3	0	3	-1	-6
E	-1	-1	-1	1	-5	2	6	-3	0	-4	-4	1	-2	-4	-2	0	-1	-4	-3	-3	1	4	-1	-6
G	0	-3	-1	-2	-4	-2	-3	6	-3	-5	-4	-2	-4	-4	-3	-1	-2	-4	-4	-4	-1	-3	-2	-6
H	-2	0	0	-2	-4	1	0	-3	8	-4	-3	-1	-2	-2	-3	-1	-2	-3	2	-4	-1	0	-2	-6
I	-2	-3	-4	-4	-2	-3	-4	-5	-4	5	1	-3	1	-1	-4	-3	-1	-3	-2	3	-4	-4	-2	-6
L	-2	-3	-4	-5	-2	-3	-4	-4	-3	1	4	-3	2	0	-3	-3	-2	-2	-2	1	-4	-3	-2	-6
K	-1	2	0	-1	-4	1	1	-2	-1	-3	-3	5	-2	-4	-1	-1	-1	-4	-3	-3	-1	1	-1	-6
M	-1	-2	-3	-4	-2	0	-2	-4	-2	1	2	-2	6	0	-3	-2	-1	-2	-2	1	-3	-2	-1	-6
F	-3	-4	-4	-4	-3	-4	-4	-4	-2	-1	0	-4	0	6	-4	-3	-2	0	3	-1	-4	-4	-2	-6
P	-1	-2	-3	-2	-4	-2	-2	-3	-3	-4	-3	-1	-3	-4	8	-1	-2	-5	-4	-3	-2	-2	-2	-6
S	1	-1	0	-1	-2	0	0	-1	-1	-3	-3	-1	-2	-3	-1	5	1	-4	-2	-2	0	0	-1	-6
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-2	-1	-1	-2	-2	1	5	-4	-2	0	-1	-1	-1	-6
W	-3	-4	-4	-6	-3	-3	-4	-4	-3	-3	-2	-4	-2	0	-5	-4	-4	11	2	-3	-5	-4	-3	-6
Y	-2	-3	-3	-4	-3	-2	-3	-4	2	-2	-2	-3	-2	3	-4	-2	-2	2	7	-2	-3	-3	-2	-6
V	0	-3	-4	-4	-1	-3	-3	-4	-4	3	1	-3	1	-1	-3	-2	0	-3	-2	4	-4	-3	-1	-6
B	-2	-2	4	4	-4	0	1	-1	-1	-4	-4	-1	-3	-4	-2	0	-1	-5	-3	-4	4	0	-2	-6
Z	-1	0	0	1	-4	3	4	-3	0	-4	-3	1	-2	-4	-2	0	-1	-4	-3	-3	0	4	-1	-6
X	-1	-1	-1	-2	-3	-1	-1	-2	-2	-2	-2	-1	-1	-2	-2	-1	-1	-3	-2	-1	-2	-1	-1	-6
*	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	1

7 Prohledávání databází podle podobnosti se známou sekvencí

Nyní, když už bylo vysvětleno, jak k sobě dvě sekvence přiřazovat a které databáze použít, se může přikročit již k samotnému prohledávání, neboli k tomu, které sekvence v prohledávané databázi jsou nejpodobnější zvolené sekvenci (od této chvíle už dotaz – query). Zároveň je vhodné, kdyby bylo možné posoudit, zda ty nejpodobnější sekvence jsou dotazu vskutku podobné víc než jen náhodně[1] – tedy statisticky vyhodnotit výsledky vyhledávání.

Při prohledávání databází je na výběr de facto ze dvou typů algoritmů:

- Přesný algoritmus
- Heuristické algoritmy

7.1 Přesný algoritmus

U přesného algoritmu se postupuje přibližně takto:[1]

- sestrojí se bodový diagram podobnosti,
- odvodí se z něj co nejdelší lokální přiřazení a
- pro něj se vypočítá celková hodnota podobnosti s dotazem,(viz kap. 6.2)
- všechny sekvence v databázi se seřadí podle hodnot podobnosti s dotazem (tím se zjistí, která sekvence je dotazu nejbližší),
- zaznamená se distribuci zjištěných hodnot podobnosti dotazu s jednotlivými sekvencemi z databáze a
- zjistí se, jakou křivku rozdělení tyto hodnoty sledují.
- Hodnoty podobnosti, které leží výrazně mimo křivku odpovídající nahodilé distribuci, odpovídají sekvencím, jejichž míra shody s dotazem je zřejmě větší než nahodilá.

Uvedený postup je poněkud naivním a zejména ve statistické části značně zjednodušeným příkladem přesného prohledávacího algoritmu, který by databázi prohledal způsobem vskutku vyčerpávajícím. Nevýhodou takového algoritmu je nutnost sestrojovat přiřazení pro všechny sekvence v databázi a z toho plynoucí pomalost. Proto byly přesné algoritmy v praxi do značné míry vytlačeny postupy heuristickými.(viz kap. 7.2)[1]

I přes hardwarovou a časovou náročnost existují prakticky použitelné implementace přesných prohledávacích algoritmů. Příkladem může být program SSEARCH, určený k prohledávání proteinových databází proteinovým dotazem a nukleotidových databází nukleotidovým dotazem.[1] (viz kap. 11 tab. 5)

7.2 Heuristický algoritmus

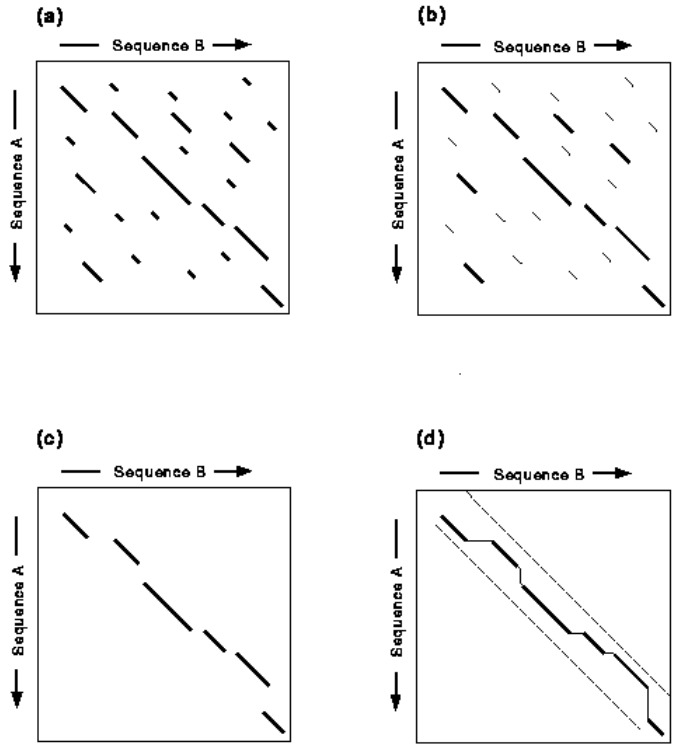
Heuristické algoritmy pro porovnávání podobnosti dvou sekvencí jsou nejčastěji používané pro případy lokálního přiřazení sekvencí. (viz kap. 6.2) Modelovým příkladem heuristického prohledávání databáze je FASTA (o který se ji hovořilo jakožto s „dárcem“ jména jednoho z nejběžnějších formátů) [1] a dalším, který doznal ještě lepších výsledků je algoritmus BLAST. Oba dva principy těchto algoritmů budou v následujících podkapitolách vysvětleny.

7.2.1 Algoritmus FASTA

Klíčovým krokem ke zrychlení bylo připustit si, že k prvnímu hrubému odhadu míry podobnosti sekvencí v databázi s dotazem vlastně ani není potřeba sestrojovat přiřazení. K první filtraci databáze, tedy k vyloučení sekvencí, kterými nestojí za to se zabývat, může stačit samotný bodový diagram.[1]

Algoritmus FASTA pracuje tak, že:[1]

- Pro každou sekvenci z databáze a pro dotaz je nejprve sestrojen bodový diagram (viz obr. 16 a))s velikostí „okénka“ (ktuple) o předem zvolené délce. V něm je pak nalezen předem určený počet „nejlepších“ diagonál (viz obr. 16 b)), tedy diagonál o co největším počtu identit
- Pro tyto diagonály je s využitím předem zvolené substituční matice vypočtena nenormalizovaná hodnota podobnosti (init1). Nedosahuje-li hodnota podobnosti nejlepší diagonály alespoň předem stanovené mezní hodnoty (cutoff), program se danou sekvencí z databáze přestane zabývat
- Pokud hodnota podobnosti init1 pro nejlepší diagonálu překročí mezní hodnotu, program prozkoumá, zda s touto diagonálou nesousedí diagonála jiná. Jestliže ano, program obě diagonály propojí (viz obr. 16 c))a vypočte novou hodnotu podobnosti initn jako součet hodnot init1 pro výchozí diagonály, o něhož byla za každou mezeru odečtena konstanta. Není-li s čím propojovat, pak initn =init1
- Jestliže hodnota podobnosti initn přesáhne zvolenou mezní hodnotu (threshold), program teprve pro příslušnou diagonálu poctivě sestrojí co nejdelší lokální přiřazení (viz obr. 16 d)) a znovu vypočte optimalizovanou hodnotu podobnosti opt za použití zvolené substituční matice a ceny mezer



Obrázek 16 Algoritmus FASTA, zdroj [13]

Hodnoty podobnosti opt jsou průběžně zaznamenávány a je stanovena jejich distribuce, označovaných jako Z-score.(viz kap. 7.3) [1]

Výstup se pak skládá ze seznamu nalezených frekvencí, série přiřazení mezi dotazem a nejbližšími nalezenými sekvencemi a údaji o statistickém hodnocení a pro nalezené „významné“ sekvence i hodnota „očekávanosti“ E, vyjadřující počet sekvencí vykazujících stejnou podobnost s dotazem.(viz Příloha 2)[1]

U algoritmus FASTA je nevýhodou, že se někdy může dopustit falešně negativních výsledků (tj. zmeškání sekvence významně podobné dotazu), ale bohužel i falešně pozitivních výsledků (tj. najde sekvence, které dotazu příbuzné nejsou).[1]

Existují i různé modifikace pro prohledávání nukleotidových sekvencí proteinovým dotazem a pro prohledávání proteinových databází nukleoidovým dotazem, jsou to například:[1]

TFASTX (dokáže prohledávat proteinovým dotazem nukleotidovou databázi) a FASTX, FASTY (dokážou prohledávat nukleotidovým dotazem proteinovou databázi). (viz tab. 5)

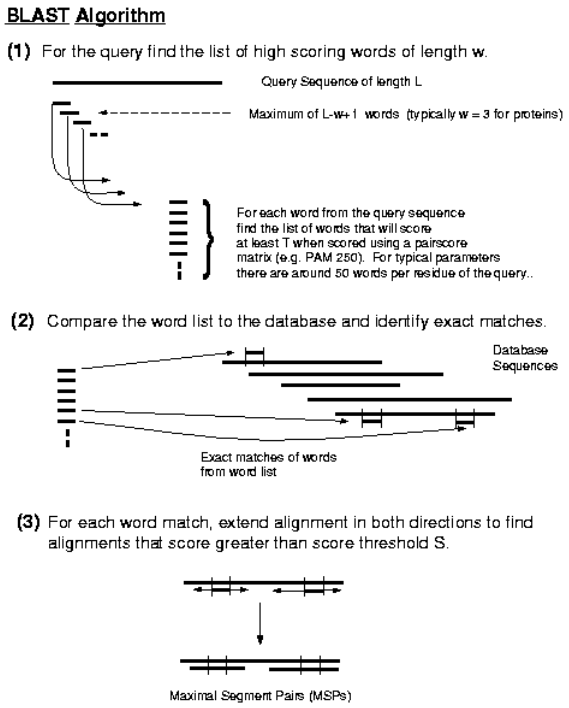
7.2.2 Algoritmus BLAST

V současné době je zřejmě nejrozšířenějším heuristickým algoritmem. Uvádí se, že je až 6x rychlejší než algoritmus FASTA. (viz kap. 7.2.1)

Algoritmus BLAST pracuje tak, že: [1]

- pro sekvenci dotazu programu nejprve vytvoří „slovník“ (index) výskytu všech přítomných kombinací symbolů (slov) o předem zvolené délce
- pro každé slovo (word) ze slovníku nejprve zjistí hodnoty podobnosti za použití zvolené substituční matice. V dalším kroku pak ze slovníku vypustí všechny slova, jejichž výskyt v páru přiřazených sekvencí přispívá k výsledné hodnotě podobnosti hodnotou menší než je jistá předem zvolená mezní hodnota T (threshold) (viz obr. 17 (1))
- V následujícím kroku program prohledává všechny sekvence databáze a eviduje pro každou sekvenci výskyt slov ze slovníku (viz obr. 17 (2))
- Pro každou takto nalezenou dvojici „významných“ slov v těsné vzájemné blízkosti program slova propojí a použije jako jádro lokálního přiřazení, které pak oběma směry rozšiřuje, dokud hodnota podobnosti nezačne klesat. Pokud zjištěná hodnota podobnosti (score) dosáhne alespoň předem zvolené mezní hodnoty (cutoff) program přiřazení zaznamená jako tzv. HSP (high scoring pair), jinak přiřazení opustí. (viz obr. 17 (3))
- Pro každou databázovou sekvenci obsahující HSP pak program vypočte úhrnnou normalizovanou hodnotu podobnosti ze všech HSP, která je následně použita ve statistickém hodnocení podobně jako u agutimu FASTA.

Výstup programu obsahuje seznam sekvencí s nejlepšími HSP spolu s hodnotami očekávatelnosti E (viz kap. 7.3), které odrážejí předpokládaný počet sekvencí o stejné nebo lepší podobnosti s dotazem ve stejně velké databázi složené z náhodných sekvencí.[1]



Obrázek 17 Algoritmus BLAST, zdroj [13]

Hlavním zdrojem zrychlení oproti algoritmu FASTA je první krok, rychlosti a relativní nenáročnosti programu však přispívá i vlastní algoritmus prohledávání databáze ve 2. kroku. V konečném důsledku BLAST dokáže „růst s databází“ podstatně lépe než FASTA a je mimořádně vhodný i k provozování jako veřejně dostupná vyhledávací služba. Prakticky všechny veřejně přístupné databáze sekvencí dávají k dispozici server s programem BLAST provozovaná prostřednictvím webového rozhraní. Verze programu BLAST provozovaná a nadále vyvíjená na americkém Národním centru pro biotechnologické informace, tzv. NCBI BLAST se dnes stala již standardním webovým nástrojem k prohledávání rozsáhlých databází. NCBI BLAST je opatřen přehledným grafickým rozhraním, přístupným prostřednictvím webového prohlížeče,(viz obr. 21) a obsáhlou dokumentací.[1]

Zajímavým rozšířením možností algoritmu BLAST, přístupným například právě prostřednictvím serveru NCBI, je PSI-BLAST- pozičně specifický interovaný BLAST (position-specific iterated BLAST) dostupný pouze ve verzi pro proteinové sekvence, umožňuje nacházet vzdálenější homology dotazu a tedy posunout horizont směrem k hlubší evoluční minulosti.[1] Dalšími modifikacemi stejně jako u algoritmu FASTA jsou:[1]

BLASTN (k prohledávání nukleotidové databáze nukleotidovým dotazem), BLASTP (k prohledávání proteinových databází proteinovým dotazem), BLASTX (k prohledávání proteinové databáze nukleotidovým dotazem) TBLASTX (určený k prohledávání všech možných překladů nukleotidové databáze všemi možnými překlady nukleotidového dotazu; v podstatě rozšíření BLASTX o přístup k překladům neanotovaných nukleotidových sekvencí).(viz tab. 5)

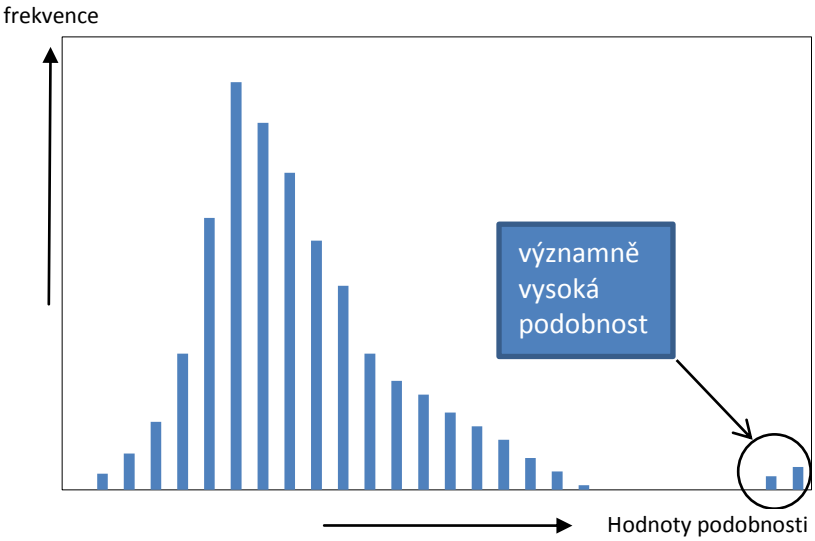
7.3 Statistické skórovací hodnoty

Zde je vysvětlení statistických hodnot, které se zaznamenávají při prohledávání a někdy se zobrazují ve výsledcích:[3]

- Z-score – míra, jak nepravděpodobná je nalezená shoda; čím větší číslo, tím větší pravděpodobnost, že srovnání není dílem náhody
- P-value – pravděpodobnost, že pozorovaná shoda je dílem náhody
- E-value – počet podobných záznamů se stejnými hodnotami skóre jako pozorovaný záznam, které mohou vzniknout v dané databázi náhodně (E=P*N ; N –velikost databáze)

Když hodnota E bude menší než 0.02, sekvence jsou pravděpodobně homologní. Když bude E mezi hodnotami 0.02 < E< 1 homologie není vyloučena a pokud E bude větší 1, shoda je zřejmě výsledkem náhody. [3]

Další statistická hodnota, která se objevuje ve výsledcích prohledávání, příklad hypotetického histogramu je na grafu 1.



Graf 1 Hypotetický příklad distribuce hodnot podobnosti, zdroj [1]

8 Mnohonásobné přiřazení proteinových sekvencí

Mnohonásobné přiřazení (multiple alignment) je další záležitostí, kterou se zabývá praktická BI. Používá se například při definici evolučně konzervovaných či konvergentních sekvenčních motivů. Sekvenčními motivy se označují oblasti aminokyselinových sekvencí sdílené nějakou skupinou proteinů. Ve většině prakticky významných případů je skupinou míněna evolučně spřízněná rodina homologních proteinu spjatých společným původem, jde tedy o motivy evolučně konzervované. Lze však i setkat s jednoduchými motivy vzniklými často konvergencí (místo určující buněčnou lokalizaci proteinů). Mnohočetné přiřazení se sestavuje zpravidla buď proto, aby se na základě evoluční konzervace usuzovalo na funkční význam konkrétních pozic (např. identifikovali aktivní místa enzymů nebo regulačně významná místa post-translačních modifikací), nebo aby se z nich vyvozovaly hypotézy o evoluční minulosti studované genové rodiny. (viz kap. 9) [1]

8.1 Mnohonásobné srovnání sekvencí

I řešení přiřazení dvou sekvencí není triviální záležitostí. U mnohočetného přiřazení pak přistupují další problémy, zejména dramatický nárůst výpočetní náročnosti. Zejména s ohledem na tento jev mají prakticky všechny reálně použitelné algoritmy heuristickou povahu, podobně jako v případě prohledávání databází sekvenčním dotazem.[1]

Mnohonásobné srovnání je seřazení tří a více sekvencí nukleových kyselin nebo proteinů s mezerami vloženými do sekvencí tak, že úseky sekvencí s úplnou nebo částečnou homologií jsou seřazeny nad sebou ve stejném sloupci. Obdobně jak tomu bylo u dvou sekvencí. (viz kap. 6.2)

Obecně používaným programem pro mnohonásobné seřazení sekvencí je Clustal W pracující na heuristickém principu vyjmenovaným v následné kap. 8.1.1.[4]

8.1.1 Algoritmus CLUSTAL

Konkrétní příklad strategie pro plně automatickou konstrukci mnohočetného přiřazení si lze ukázat na příkladu algoritmu CLUSTAL. Postup používaný při konstrukci mnohočetného přiřazení metodou CLUSTAL lze shrnout do následujících kroků:[1]

- Program nejprve vzájemně porovná všechny možné dvojice vložených sekvencí a z výsledků sestrojí matici párových normalizovaných hodnot vzdálenosti. Pro vlastní stanovení párových hodnot vzdálenosti může uživatel zvolit buď rychlou, ale méně přesnou metodu, která vychází z hodnot podobnosti (score) odpovídajících nejdelším diagonálám bodového diagramu, nebo pomalejší, avšak přesnější postup, který zahrnuje optimalizaci párového přiřazení pro každou dvojici sekvencí. Tento krok je z celého postupu nejpomalejší a jeho trvání roste s počtem sekvencí kvadraticky.
- Na základě matice podobností program provede shlukovou analýzu a sestrojí „vůdčí strom“ – tedy dendrogram, který postupně sdružuje sekvence podle míry vzájemné podobnosti od nejbližších ke vzdálenějším. (viz kap. 9)
- Nakonec program sestrojí vlastní přiřazení, tak, že nejprve vytvoří párová globální přiřazení těch sekvencí, které jsou ve vůdčím stromu bezprostředními sousedy. V následujících krocích postupně přiřazuje další sekvence v pořadí dané dendrogramem.

Obrázek 18 Výstup z programu Clustal W, zdroj [4]

Konstrukce dobrého mnohočetného přiřazení je, dokonce i za použití vhodných heuristik, úkolem náročným na čas (uživatelsky i hardwarově) a pokud se této činnosti chce někdo věnovat systematicky, stojí za to si pořídit lokální software. Přesto však existují veřejné servery, poskytující tuto službu. Jsou to například: [1]

- na serveru databáze BLOCKS program BLOCKMAKER poskytuje nástroj pro vyhledávání krátkých konzervativních úseků v uživatelem zadaných sekvencích a tvorbu lokálního přiřazení;
- algoritmus T-Coffee dosažitelný například prostřednictvím odkazu ze stránek Evropského bioinformatického institutu, představuje zajímavou alternativu CLUSTALu – totiž plně automatizovaný postup pro konstrukci globálního mnohočetného přiřazení, sice pomalejší než CLUSTAL, avšak odolnější vůči „mezerovému“ artefaktu
- K srovnatelným výsledkům jako T-Coffee při významně menších nárocích na strojový čas lze dospět pomocí novějšího programu MUSCLE
- Zvláště pro konstrukci přiřazení vzdáleně příbuzných sekvencí byl v laboratořích firmy IBM vyvinut algoritmus MUSCA

8.2 Přiřazování trojrozměrných struktur a modelování struktury proteinů

Predikce trojrozměrné struktury proteinu založené na mnohočetném přiřazení dvou a více sekvencí představuje poměrně zásadní problém, který v obecné rovině dosud není uspokojivě vyřešen. Proto je zde zmínka o jednom možném postupu. [1]

Výrazné usnadnění predikce trojrozměrných struktur proteinů je v případech, kdy je již známa experimentálně zjištěná prostorová struktura proteinu o podobné sekvenci. V takovém případě se totiž může použít experimentálně zjištěná struktura jako „forma“ – templát, podle kterého se nejprve virtuálně „poskládá“ řetězec odpovídající zkoumané sekvenci a pak pouze drobnými úpravami se vyřeší případné strukturní či energetické konflikty (např. elektrostatické pnutí či prostorovou kolizi aminokyselinových zbytků). Postupy založené na právě popsaném principu bývají označovány jako navlékací algoritmy (treading algorithms). Na navlékání samotné se lze dívat jako na speciální typ konstrukce přiřazení- totiž „přiřazení“ mezi sekvencí a strukturou. Popisovaný postup je založen na využití modelovací služby SwissModel ve spojitosti se zdarma stažitelným klientským programem Deep View, který umožňuje uskutečnit první – časově poměrně náročné – kroky postupu na uživatelském počítači. [2]

9 Studium příbuzenských vztahů biologických sekvencí

Sekvence biologických makromolekul poskytují bohatý zdroj informací o příbuzenských vztazích mezi organismy i jejich geny. [1] Neboli v této kapitole bude zmíněno o evoluční interpretaci sekvencí. Budou zde nastíněny kroky tvorby „genealogického stromu“ čili dendrogramu.

Dendrogram je graf, který spojuje operační taxonomické jednotky (OTU, Operational Taxonomic Units, reprezentované sekvencemi vybraných vzájemně ortologních genů z vybraných jedinců) prostřednictvím větví (branches) s uzly (nodes), které reprezentují hypotetické společné předky propojených OTU a jsou navzájem dalšími spojnicemi propojeny s předchozí „generací“ společných předků, reprezentovaných uzly hlouběji uvnitř stromu. Je předpokládáno, že geny se zmnožují duplikací a taxony[6] odštěpováním populací; větvení dendrogramu je tedy binární. [1]

Všechny spojnice vycházejí z uzlu a všechny uzly na nich ležící tvoří „větev vyššího řádu“ čili klad (clade). Délka větví odráží „evoluční vzdálenost“ uzlů, kterou si lze představit jako počet mutací, který odděluje uzly propojené danou spojnicí. Z praktických důvodů se dendrogram často uvádí v pravoúhlé grafické reprezentaci. Přehlednější, avšak na výpočetní grafické provedení náročná je „hvězdicovitá“ forma dendrogramu.(viz obr. 19) Má-li délka větve odrážet čas uplynulý od oddělení linií vycházejících z uzlu, musí být splněny další předpoklady – např. stálá a pro všechny sekvence stejná frekvence mutací, což je zejména u sekvencí kódujících proteiny a podléhajících selekčnímu tlaku málokdy splněno. Proto se někdy v dendrogramech délka větví neuvádí a je pak podstatná pouze topologie „stromu“. [1]

Umístění společného hypotetického předka všech OTU čili kořene stromu (root) představuje závažný problém. Nejjednodušším řešením je začlenit do analýzy tzv. outgroup – sekvenci, která je sice homologní, avšak vzdálená ostatním studovaným vzorkům. Není-li k dispozici vhodná sekvence, nezbývá než sestrojit, případně interpretovat strom jakožto nezakořeněný (unrooted).[1]

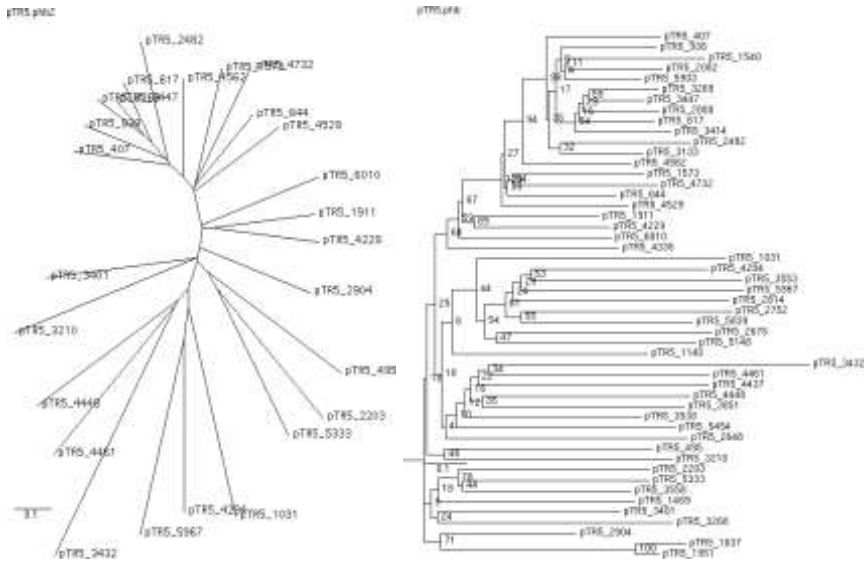
Grafická forma dendrogramu se příliš nehodí k tomu, aby se na ní či z ní cokoli počítalo; podstatně snazší je pracovat s daty zapsanými pomocí standardní sady znaků. Všechny běžné fylogenetické programy dovedou zacházet s daty v tzv. newickovském formátu (Newick Format), který zachycuje vzájemné propojení uzlů pomocí závorek a umožňuje i zápis délky větví. Např. ((((A : 4, B:8) : 2, (C : 6, D:1) : 6) : 3, E : 9):1, F : 16). [1]

K výslednému dendrogramu může být založen na jednom ze dvou principů. Buď [1]

[6] Taxon neboli systematická jednotka neboli taxonomická jednotka je skupina konkrétních (žijících nebo vymřelých) organizmů, které mají společné určité znaky (nejčastěji jsou příbuzné) a tím se odlišují od ostatních taxonů

- pomocí nějaké algoritmické metody se sestrojí jediný „nejlepší možný“ strom, nebo
- se vyhledá nejlepší z množiny všech možných dendrogramů, které by se na základě vstupních dat daly postavit.

Prakticky se proto často, podobně jako v případě prohledávání velkých databází, používají heuristické metody.



Obrázek 19 Evoluční strom pTR5 rodiny lidských endogenních retrovirů, zdroj [3]

10 Porovnání konkrétních databází

V této kapitole dojde k porovnání dvou ze tří primárních databází, resp. jejich online uživatelskému rozhraní, o kterém by se dalo říci, že dává databázím polidštěnou tvář a je v podstatě vstupní branou do nepřeberného množství informací.

10.1 Uživatelská webová rozhraní

Webové uživatelské rozhraní slouží k vyhledávání, ale i stahování záznamů z databází. Na rozdíl od obsahu databází, který je v rámci Velké trojky téměř zcela sjednocen, uživatelské rozhraní je pro každou databázi specifické. Pro uživatele je velmi důležité, aby bylo co nejpřehlednější, protože už samotné dolování informací z internetu není někdy snadné a u biologických dat to platí taktéž.

V poměrech evropských a amerických typický uživatel (neznalý japonštiny) přistupuje k databázím Velké trojky prostřednictvím:[1]

- **SRS** (Sequence Retrieval Systém) databáze EMBL na Evropském ústavu pro bioinformatiku (viz obr. 20)

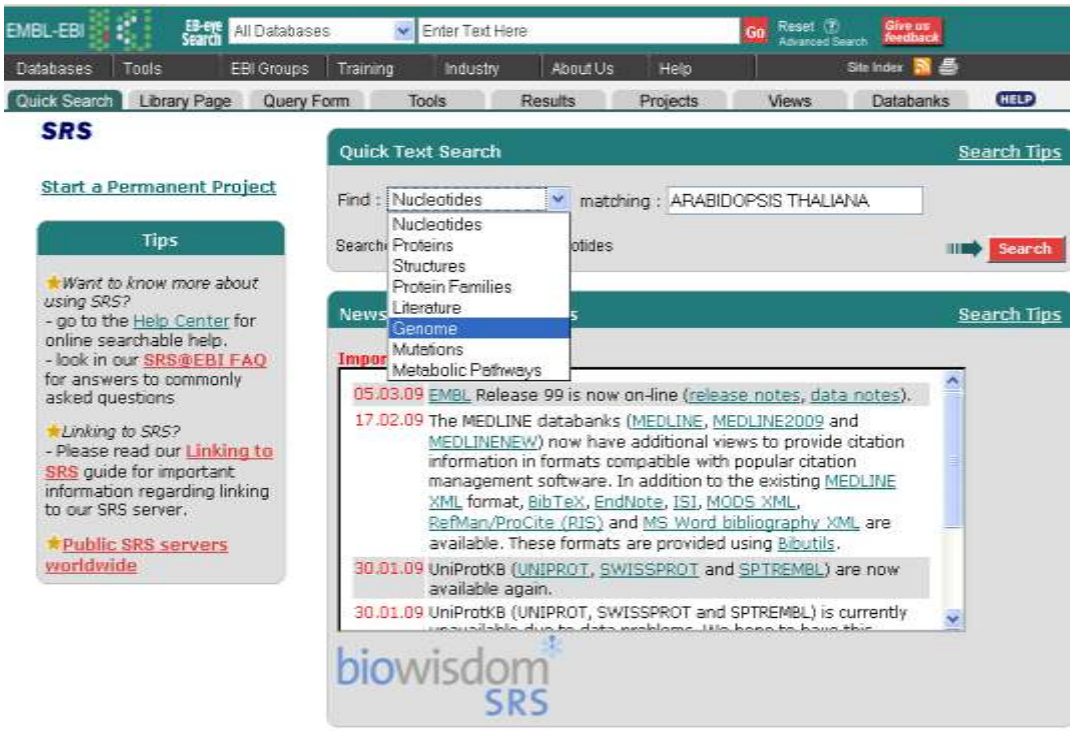
- **Entrez** databáze GenBank v americkém Národním centru pro biotechnologické informace (viz obr. 21)

Jak již bylo řečeno, tato dvě webová rozhraní jsou obsahově stejná, konkrétní případy v jakých ohledech se shodují, budou nyní vyjmenovány:

- vyhledávání záznamů na základě přístupových kódů, klíčových slov v názvech a anotacích, jmen autorů a literárních citací

U tohoto případu bylo zjišťováno vyhledávání sekvencí podle názvu organismu. Byl zvolen dnes již klasický experimentální model prokaryontního organismu Arabidobsis thaliana neboli česky husečníku.

V rozhraní SRS se požadovaný záznam zadal do vyhledávacího políčka a musela se zvolit databáze, z jaké chce uživatel hledat.(viz obr. 20) Výsledek je pak zobrazen v novém okně ve formě přístupových kódů. Po kliknutí na libovolný kód se zobrazila příslušná anotace. (viz Příloha 1)



Obrázek 20 Webové uživatelské rozhraní SRS, zdroj [6]

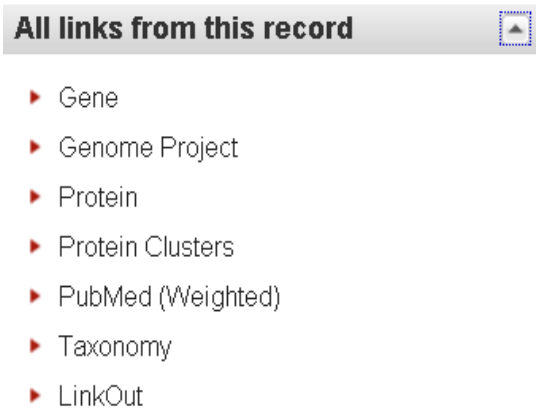
V rozhraní Entez se také zadalo jedno z hesel (přístupový kódů, klíčové slovo v názvech a anotacích, jméno autorů a literární citace) do vyhledávajícího pole a opět byl zvolen název organismu Arabidopsis thaliana. Počty nalezených sekvencí se zobrazily jako čísla v řádcích u jednotlivých databází (viz obr. 21) Po kliknutí na příslušnou databázi se zobrazily též linky kódů anotací.

Obrázek 21 Webové uživatelské rozhraní Entrez - výsledky vyhledávání záznamů v jednotlivých databázích pro Arabidobsis thaliana, zdroj [5]

- putování mezi záznamy, případně mezi databázemi, pomocí křížových odkazů (např. mezi sekvencí DNA a odvozeného proteinu či mezi jednotlivými sekvencemi publikovanými v téže práci) [1]

V rozhraní SRS se tato možnost odkazování na další informace učiní pomocí zakroužkovaného Link (viz obr. 22) a v rozhraní Entrez též pomocí linků, ale již rozepsaných (viz obr. 23)

Obrázek 22 Odkaz na další databáze



Obrázek 23 Odkaz na další databáze v rozhraní Entrez, zdroj [5]

- přístup k pokročilejším nástrojům, např. prohledávání na základě podobnosti sekvencí samotných

Tento přístup zajišťují vyhledávací programy při nastavování parametrů. (viz kap. 11)

Jak již bylo uvedeno v předcházejícím výčtu, vždy, v obou dvou rozhraních, se pomocí nějakého hesla a v nějaké databázi, vyhledaly příslušné sekvence s linkem na anotaci, spolu s linky na další informace k danému organismu, ale také bylo možné zpozorovat, že k informacím se dalo dostat odlišnou cestou, neboli každé rozhraní mělo jiné provedení. V tabulce 3 budou tytou tyto rozdíly rozepsány spolu s označením výhody nebo nevýhody.(viz tab. 4)

Tabulka 3 Výhody a nevýhody rozdílnosti v provedení u uživatelského webového rozhraní SRS a Entrez, zdroj [Vlastní]

Rozhraní	**SRS**	**+/-**	**Entrez**	**+/-**
Účelovost	Skýtá výrazně více funkcí (včetně možnosti zavést si účet na serveru a de facto si postavit vlastní dílčí databázi)	++	Méně funkcí	--
Orientace	Často používané nástroje se obtížně hledají	---	Jednodušší, přívětivější- prakticky všechny běžně používané nástroje jsou rychle a snadno dostupné z jednoho místa	+++
Stálost webových stránek	Stránky procházení neustálým rozvojem	--	Mnoho let stabilně uspořádané	+-
Přístup k vlastním proteinovým databázím	ANO – trEMBL	++	ANO- GenPept	++
Odkaz na 3D strukturu proteinu v anotaci	ANO-databáze PDB	+++	NE	---

Tabulka 4 Ohodnocení výhody/nevýhody v rozdílnosti provedení u SRS a Entrez, zdroj [Vlastní]

+	**výhoda**	**-**	**nevýhoda**
+	malá	-	malá
++	velká	--	velká
+++	největší	---	největší

Obě rozhraní mají zajisté nějaké pro a proti. Ale vzhledem k obrovskému kvantu dat, které rostou exponenciální řadou, je kladen největší důraz na přehlednost a snadnou orientaci při vyhledávání

požadovaných informací, případně i návaznost na další zdroje informací. A tím jednoznačně připadá doporučení na americké webové uživatelské rozhraní Entrez.

11 Prohledávání databází sekvenčním dotazem

Dříve než se začne s vyhledáváním v databázích a tedy i porovnáváním konkrétní sekvence pomocí vyhledávacích programů, musí si uživatel samozřejmě uvědomit, jaký typ sekvence chce porovnávat s jakou databází. Jedno z rozhodnutí, které musí uživatel učinit sám, je právě výběr vyhledávacího programu. A jaký program se hodí na konfrontaci sekvence s databází? S tímto problémem by mu mohla pomoct následující tabulka (viz tab. 5) Jednotlivé programy jsou uvedeny v kapitole 7.2.

Tabulka 5 Přehled program pro vyhledávání homologů v sekvenčních databázích, zdroj [Vlastní]

Dotaz (guery)	**Databáze**	
	DNA	**Protein**
DNA	FASTA BLASTN SSEARCH	BLASTX FASTX FASTY
Protein	TBLASTN TFASTX	BLASTP PSI-BLAST FASTA SSEARCH

Vysvětlení tabulky je následující:

- pokud je sekvence z DNA, tedy nukleových kyselin, a uživatel jí chce porovnat s databází nukleových kyselin, může použít buď program FASTA, BLASTN nebo SSEARCH
- pokud je sekvence z nukleových kyselin, a uživatel jí chce porovnat s databází aminokyselin, může použít program TBLASTN, TFASTX
- pokud je sekvence z proteinu, tedy z aminokyselin, a uživatel jí chce porovnat s databází nukleových kyselin, může použít program BLASTX, FASTX nebo FASTY
- pokud je sekvence z proteinu, a uživatel jí chce porovnat s databází proteinů, může použít program BLASTP, PSI-BLAST, FASTA, SSEARCH

11.1 SSEARCH vs. FASTA vs. BLAST

V další podkapitole 11.2 bude stanovena již konkrétní sekvence (dotaz). Zvolila se sekvence nukleotidových kyselin a bude konfrontován s nukleotidovou databází. Tudíž se může použít podle tabulky 5, buď program BLASTN, FASTA nebo SSEARCH. Dříve než se začne s prohledávání, budou nejdříve v tabulce 6 jednotlivé programy porovnány podle parametrů, podle kterých se bude v závěru hodnotit výsledek vyhledávání.

Tabulka 6 Porovnání algoritmů, zdroj [Vlastní]

PROGRAM	**FASTA**	**BLASTN**	**SSEARCH**
Algoritmus	Heuristický	Heuristický	Přesný
Rychlost vyhledávání	2	1	3
Citlivost vyhledávání	1(zvláště u nukleotidových sekvencí)	3	2
Možnost dopuštění se chyby	3	1	2

Tabulka 7 Ohodnocení kritérií programu FASTA, BLAST, SSEARCH, zdroj [Vlastní]

Známka (čím nižší, tím lepší)	Rychlost	Citlivost	Možnost dopuštění se chyby
1	nejrychlejší	vysoká	nízká
2	normální	střední	relativní
3	pomalá	malá	vysoká

11.2 Formát sekvence

Dotaz, který se bude porovnávat s databázemi, bude opět z organismu Aradopsis thaliana, Sekvence je v „surovém“ formátu, který dokážou přijímat všechny programy, které jsou online na stránkách

- http://blast.ncbi.nlm.nih.gov/Blast.cgi,
- http://www.ebi.ac.uk/Tools/fasta33/index.html,
- http://ssearch.ddbj.nig.ac.jp/top-e.html.

Dotaz vypadá následovně:

```
ATGGACAAAGTTATGAGAATGTCGTCCGAAAAAGGGGTGGTTATATTTACCAAGAGCTCCTGTTGTTTGT
CCTATGCGGTTCAAGTTCTCTTCCAAGATCTTGGTGTTAACCCTAAGATCCACGAGATTGATAAGGACCC
TGAATGCCGAGAGATAGAGAAGGCTCTTATGAGGCTAGGGTGTTCAAAGCCGGTCCCAGCCGTCTTCATT
GGTGGCAAGCTCGTTGGTTCGACCAACGAAGTAATGTCCATGCACCTAAGCAGCTCGCTCGTTCCCCTAG
TGAAGCCATATTTATGTTAA
```

Postup pro nalezení sekvence byl takovýto:

1. na stránkách http://www.ncbi.nlm.nih.gov/ byl zadán název Arabidopsis thaliana

2. v rozhraní Entrez se zobrazilo, kolik je v příslušné databázi záznamů, byla vybrána nukleotidová databáze (viz obr. 21)

3. kliknutím na ID záznamu byl zobrazen formulář záznamu

4. kliknutím na záložku FASTA byl záznam zobrazen v tomto formátu

5. zkopírování formátu bez hlavičky, pouze v surovém formátu,

Nyní, když už je vybrána a zkopírována sekvence, může se začít s prohledáváním databází. Aby si ale mohl čtenář představit, na jakých principech všechny tři programy přibližně fungují, jsou zde v tabulce 8 přehledně zobrazeny kroky, které provádějí při nacházení podobných sekvencí dotazu.

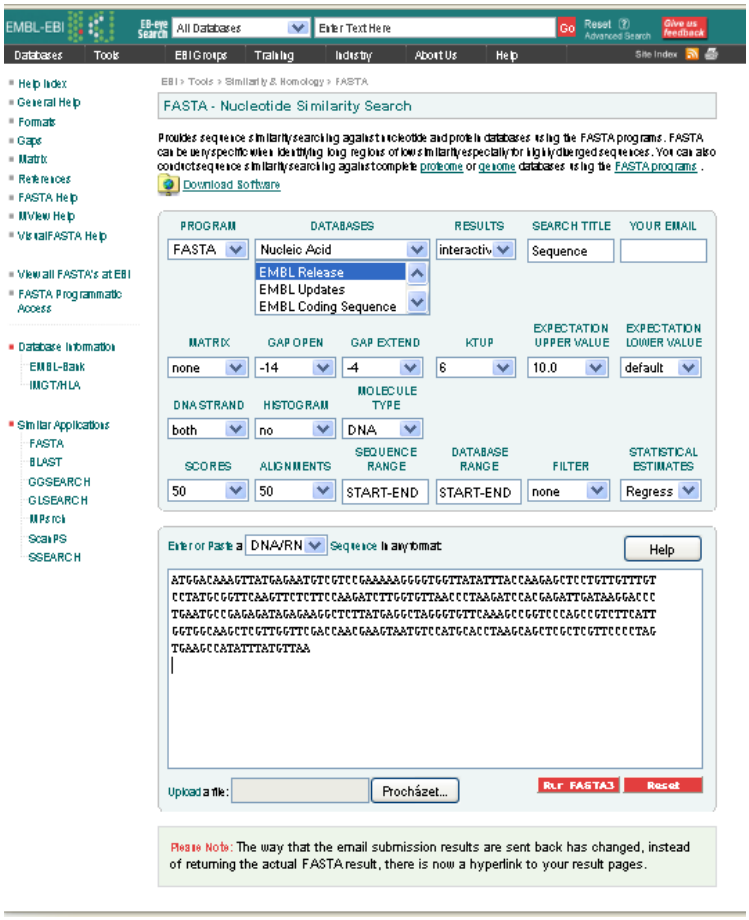
Tabulka 8 Postup výpočtů homologních oblastí u programů SSEARCH, FASTA a BLAST, zdroj [Vlastní]

	Přesný algoritmus (SSEARCH)	**FASTA**	**BLAST**
1.	Pro každou sekvenci z databáze sestrojíme bodový diagram podobnosti s dotazem a z něj odvodíme co nejdelší lokální přiřazení	Pro každou sekvenci z databáze a dotaz je nejprve sestrojen bodový diagram s velikostí „okénka“ (ktuple) o předem zvolené délce. V něm je pak nalezen předem určený počet „nejlepších“ diagonál, tedy diagonál o co největším počtu identit	Pro sekvenci dotazu program nejprve vytvoří „slovník“ (index) výskytu všech přítomných kombinací symbolů („slov“) o předem zvolené délce. Pro každé slovo (word) ze slovníku jsou nejprve zjištěny hodnoty podobnosti za použití zvolené substituční matice
2.	Pro každou sekvenci přiřazení vypočteme celkovou (nenormalizovanou) hodnotu podobnosti s dotazem	Pro tyto diagonály je s využitím předem zvolené substituční matice vypočtena nenormalizovaná hodnota podobnosti (init1). Nedosahuje-li hodnota podobnosti nejlepší diagonály alespoň předem stanovené mezní hodnoty (cutoff), program se danou sekvencí z databáze přestane zabývat	Ze slovníku jsou vypuštěna všechny slova (ať už původní, nebo odvozená), jejichž výskyt v páru přiřazených sekvencí přispívá k výsledné hodnotě podobnosti hodnotou menší, než je jistá předem zvolená mezní hodnota T
3.	Všechny sekvence v databázi seřadíme podle hodnot podobnosti s dotazem; tím jsme již zjistili, která sekvence v databázi je dotazu nejbližší	Pokud hodnota podobnosti init1 pro nejlepší diagonálu překročí mezní hodnotu, program prozkoumá, zda s touto diagonálou nesousedí diagonála jiná. Jestliže ano, program obě diagonály propojí a vypočte novou hodnotu podobnosti initn jako součet hodnot init1 pro výchoýí diagonály, o něhož byla za každou mezeru odečtena konstanta. Není-li s čím propojovat, pak initn =init1	Program prohledává všechny sekvence databáze a eviduje pro každou sekvenci výskyt slov ze slovníku. Pokud program v některé databázové sekvenci nenajde na téže diagonále nejméně dvě slova ze slovníku (hits) oddělená vzdáleností menší či rovnou než je předem stanovená „délka okna“ A, sekvenci opustí.
4.	Zaznamenáme distribuci zjištěných hodnot podobnosti dotazu s jednotlivými sekvencemi	Jestliže hodnota podobnosti initn přesáhne zvolenou mezní hodnotu (threshold), program teprve pro příslušnou diagonálu poctivě sestrojí co nejdelší lokální přiřazení a znovu	Pro každou takto nalezenou dvojici „významných“ slov v těsné vzájemné blízkosti program slova prpojí a použije jako jádro lokálního přiřazení,

	z databáze a zjistíme, jakou křivku rozdělení tyto hodnoty sledují. Hodnoty podobnosti, které leží výrazně mimo křivku odpovídající nahodilé distribuci, odpovídají sekvencím, jejich míra shody s dotazem je zřejmě větší než nahodilá.	vypočte optimalizovanou hodnotu podobnosti opt za použití zvolené substituční matice a ceny mezer	které pak oběma směry rozšiřuje, dokud hodnota podobnosti nezačne klesat. Pokud zjištěná hodnota podobnosti (score) dosáhne alespoň předem zvolené mezní hodnoty (cutoff), program přiřazení zaznamená jako tzv. HSP (high scoring pair), jinak přiřazení opustí.
5.		Hodnoty podobnosti opt jsou průběžně zaznamenávány a je stanoven a jejich distribuce, označovaných jako Z-score.	Pro každou databázovou sekvenci obsahující HSP pak program vypočte úhrnnou normalizovanou hodnotu podobnosti ze všech HSP, která je následně použita ve statistickém hodnocení.

11.3 Konkrétní prohledávání databází

A) Nejdříve proběhlo prohledávání nukleové databáze databanky EMBL programem FASTA - tento program na stránkách http://www.ebi.ac.uk/Tools/fasta33/index.html (viz obr. 24) je opatřen grafickým rozhraním, přístupným prostřednictvím webového prohlížeče. Parametry byly ponechány na výchozích, tedy substituční matice pro výpočet hodnoty podobnosti byla nechána none. Výsledky jsou k zhlédnutí v Příloze 2. Vyhledávání bylo prováděno ve všední den dopoledne, s rychlostí připojení k internetu kolem 2 Mb/s.



Obrázek 24 Zadávání sekvence a parametrů do online programu FASTA dtb. EMBL, zdroj [Vlastní]

V tabulce 9 jsou vyjmenována určitá shrnující kritéria, která hodnotí prohledávání nukleotidové databáze databanky EMBL nukleotidovým dotazem pomocí programu FASTA. A k jednotlivým kritériím (od zadávání parametrů přes prohledávání k možnosti grafické úpravy) jsou uvedeny výsledky.

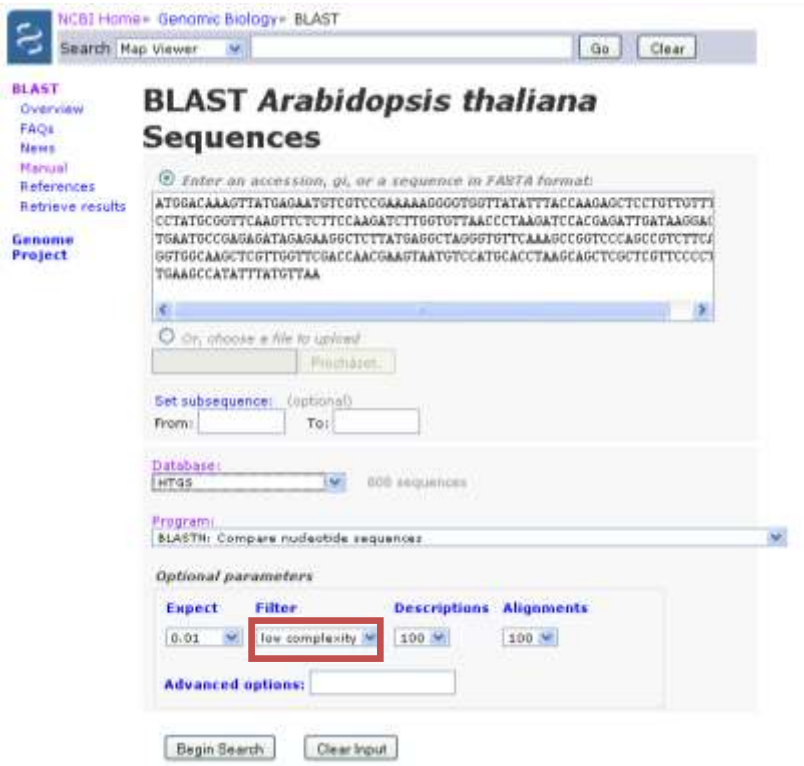
Tabulka 9 Kritéria při prohledávání databáze EMBL programem FASTA, zdroj [Vlastní]

Č.	KRITERIA	Výsledky
1.	Délka prohledávání	35 minut
2.	Počet nalezených identických sekvencí	50
3.	Možnost přednastavení parametrů	ano
4.	Možnost výběru matice	ano
5.	Možnost nastavení grafického formátu výstupu	ne
6.	Zobrazení hodnoty očekávatelnosti E	ano
7.	Filtrace oblastí o nízké komplexitě	ne
8.	Zobrazení histogramu skóre	ano
9.	Odkaz na 3D strukturu	ne
10.	Možnost odeslání výsledků na email	ano

Program FASTA vyhledal během 35 minut požadovaných 50 výsledků. Tyto sekvence pouze zobrazil s příslušným ID, zdrojem z jakého organizmu daná sekvence je (Source), délkou záznamu (Lenght), shodností v procentech (Identity %), podobností v procentech (Similar %), překrývající částí (Overlap). (viz Příloha 2) Pro zjištění dalších informací se muselo kliknout na odkaz ID. Zde je pak celá anotace záznamu. Histogram se zobrazil po kliknutí na tlačítko FASTA Results. Kde jsou i další

statistické hodnoty. Výsledky se však musí brát s rezervou, neboť program se mohl dopustit falešně negativních výsledků tím, že zmešká sekvenci významně podobnou dotazu, pokud podobnost spočívá v přítomnosti mnoho velmi krátkých úseků identity, ale i falešně pozitivních výsledků, tedy nacházení sekvencí, které dotazu příbuzné nejsou.

B) Dále proběhlo testování americké databáze GenBank pomocí programu BLASTN – který je určen pro prohledávání nukleových databází nukleovým dotazem, NCBI BLAST je opět opatřen přehledným grafickým rozhraním, přístupným prostřednictvím webového prohlížeče (viz obr. 28) a obsáhlou dokumentací. Rozhraní umožnilo celkem jednoduchý a intuitivní výběr varianty programu vhodné pro konkrétní typ dotazu i databáze, i omezení prohledávané databáze např. na sekvence odvozené z jediného organismu (opět Arabidopsis thaliana). Filtr byl nastaven na hledání oblastí s nízkou komplexitou,(viz obr 25) ostatní parametry byly ponechány na výchozí. Možnost výběru matice nebyla žádná. Výsledky testu jsou zobrazeny v Příloze 3. Vyhledávání proběhlo ve všední den dopoledne, s rychlosti připojení k internetu kolem 2 Mb/s.



Obrázek 25 Zadávání sekvence a parametrů do online programu BLAST dtb. GenBank, zdroj [Vlastní]

V tabulce 10 jsou vyjmenována stejná kritéria jako u tabulky 9, která shrnuje tentokrát prohledávání nukleotidové databáze GenBank pomocí programu BLASTN. A k jednotlivým kritériím (od zadávání parametrů přes prohledávání k možnosti grafické úpravy) jsou uvedeny výsledky.

Tabulka 10 Kritéria při prohledávání databáze GenBank programem BLAST, zdroj [Vlastní]

Č.	KRITÉRIA	Výsledky
1.	Délka prohledávání	sekundy
2.	Počet nalezených identických sekvencí	50

3.	Možnost výběru matice	ne
4.	Zobrazení histogramu skóre	ne
5.	Hodnoty očekávatelnosti E (jako měřítko příbuznosti)	ano
6.	Filtrace oblastí o nízké komplexitě	ano
7.	Možnost grafického formátu	ano
8.	Možnost přednastavení parametrů	ano
9.	Odkaz na 3D strukturu	ano
10.	Možnost odeslání výsledků na email	ne

U prohledávání databáze GenBank bylo možno vybrat databázi přímo Arabidopsis thaliana. Tudíž prohledávání trvalo jen několik vteřin. Také výstup programu byl obohacen o grafickou prezentaci ilustrující umístění podobných úseků v rámci nalezené databázové sekvence a dotazu i míru podobnosti (pomocí barevného kódování). Nalezené sekvence šly přímo stáhnout z databáze prostřednictvím webových odkazů, avšak nebyla zde funkce odeslání výsledků na email. Přednastavené parametry zahrnují i filtraci oblastí o nízké komplexitě, takže se šlo snadno vyvarovat tohoto běžného zdroje falešně pozitivních výsledků.

C) Dále proběhlo prohledávání třetí z Velké trojky japonské databáze DDBJ. Databáze je i verzi anglické, tudíž se šlo na stránkách zorientovat. I zde proběhlo prohledávání databáze stejným nukleotidovým dotazem tentokrát pomocí programu SSEARCH. Byla zde možnost označit, s jakou databází bude dotaz porovnáván. Byla požadovaná emailová adresa, kam byl výsledek zaslaný formou html odkazu. Ve webovém formuláři (viz obr. 26) byly parametry ponechány na výchozích, tedy i substituční matice pro výpočet hodnoty podobnosti byla nechána jako default. Výsledky jsou zobrazeny v Příloze 4. Vyhledávání bylo prováděno ve všední den dopoledne, s rychlostí připojení k internetu kolem 2 Mb/s.

Obrázek 26 Zadávání sekvence a parametrů do online programu SSEARCH dtb. DDBJ, zdroj [Vlastní]

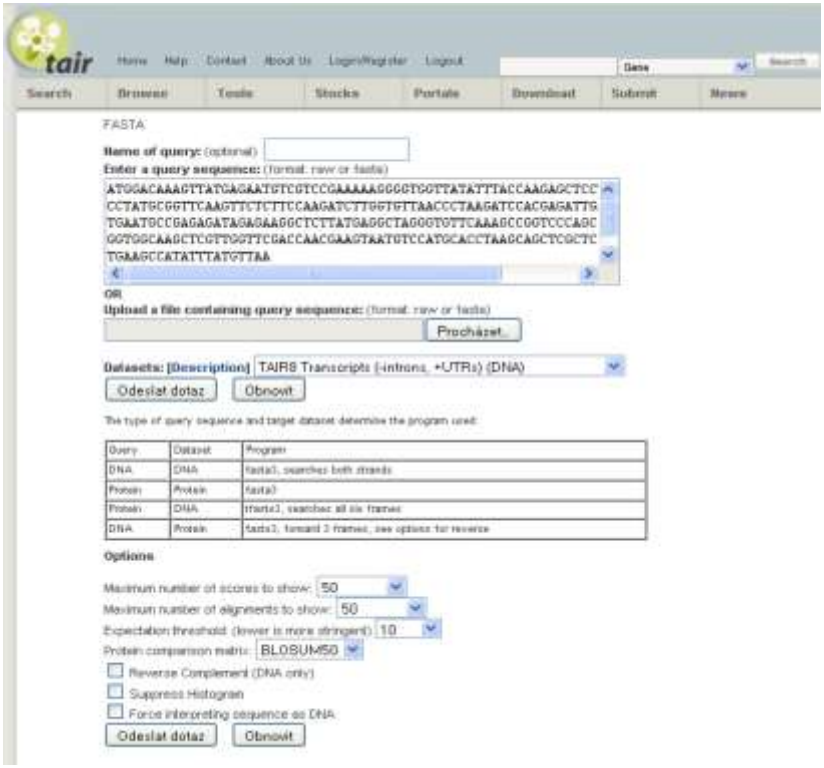
V tabulce 11 jsou opět vyjmenována stejná kritéria shrnující prohledávání, jako u předešlých dvou testů, která hodnotí testování databáze DDBJ nukleotidovým dotazem pomocí programu SSEARCH. K jednotlivým kritériím (od zadávání parametrů přes prohledávání k možnosti grafické úpravy) jsou uvedeny výsledky.

Tabulka 11 Kritéria při prohledávání databáze DDBJ programem SSEARCH, zdroj [Vlastní]

Č.	KRITÉRIA	Výsledky
1.	Délka prohledávání	3 hodiny
2.	Počet nalezených identických sekvencí	50
3.	Možnost výběru matice	ano
4.	Zobrazení histogramu skóre	ano
5.	Hodnoty očekávatelnosti E (jako měřítko příbuznosti)	ano
6.	Filtrace oblastí o nízké komplexitě	ne
7.	Možnost grafického formátu	ne
8.	Možnost přednastavení parametrů	ano
9.	Odkaz na 3D strukturu	ne
10.	Možnost odeslání výsledků na email	ano

Výsledek byl zaslán přibližně po 3 hodinách ne email, jak již bylo řečeno, jako html odkaz. Výsledek byl zobrazen v jednom okně se všemi statistickými hodnotami ve značně nepřehledné úpravě, po kliknutí na zvolenou sekvenci byl zobrazen výpis záznamu bez odkazů na jakékoli další informace. Manipulace s výsledky byla tudíž velmi komplikovaná.

D) Poslední testování proběhlo jako experiment ve specializované databázi TAIR (The Arabdopsis Information Resource) pro jediný organismus a tím je opět Arabidopsis thaliana (na stránkách http://arabidopsis.org , kde jsou mj. k nalezení i údaje o genové expresi, fenotyp mutantů, relevantní literaturu a adresář vědců, kteří se zabývají výzkumem Arabidopsis, což jiné databáze neumožňují) pomocí programu FASTA. Parametry byly ponechány na výchozích, tedy substituční matice pro výpočet hodnoty podobnosti byla nechána matice BLOMSUM50. (viz obr. 27) Výsledky prohledávání jsou zobrazeny v Příloze 5. Test byl prováděn ve všední den, dopoledne, s rychlostí připojení k internetu kolem 2 Mb/s.



Obrázek 27 Zadávání sekvence a parametrů do online program FASTA dtb. TAIR, zdroj [Vlastní]

I v tabulce 12 jsou kritéria při prohledávání tentokrát specializované databáze nukleotidovým dotazem.

Tabulka 12 Kritéria při prohledávání databáze TAIR programem FASTA, zdroj [Vlastní]

Č.	KRITÉRIA	Výsledky
1.	Délka prohledávání	5 sekund
2.	Počet nalezených identických sekvencí	50
3.	Možnost výběru matice	ne
4.	Zobrazení histogramu skóre	ano
5.	Hodnoty očekávatelnosti E (jako měřítko příbuznosti)	ano
6.	Možnost filtrace oblastí o nízké komplexitě	ne
7.	Možnost grafického formátu	ne
8.	Možnost přednastavení parametrů	ano
9.	Odkaz na 3D strukturu	ano
10.	Možnost zaslání výsledků na email	ano

Prohledávání proběhlo velmi rychle navzdory pomalejšímu algoritmu. Bylo to způsobeno menším počtem dat vzhledem k zaměření databáze se konkrétně na jediný organismus. Výsledek byl zobrazen na jedné stránce včetně histogramu - působilo nepřehledným dojmem. Po klinutí na vybranou sekvenci, byla zobrazena podrobná anotace i s odkazy na další informace o této sekvenci, např. zobrazení 3D struktury, což je velká výhoda oproti nespecializovaným databázím.

11.3.1 Závěr prohledávání vybraných databází

V tabulky 6, kde byly jednotlivé programy ohodnoceny podle: rychlosti prohledávání, citlivosti prohledávání a možnosti dopuštění se chyby, se těmito 4 testy hodnocení potvrdilo, neboť:

- za nejrychlejší byl zde označen program BLAST, což se testem potvrdilo (viz tab. 9, 10, 11, 12)
- u možnosti dopuštění se chyby byl zvolen nejlepším opět BLAST, to bylo v praxi zařízeno tím, že lze nastavit filtr oblastí o nízké komplexitě.
- citlivostí byl nejlépe ohodnocen FASTA, čímž v praxi bylo zrealizováno tím, že ve výsledcích se zobrazovaly sekvence o velmi vysoké hodnotě E, což svědčí o tom, že byly prohledávané i vzdálené sekvence.

Při prohledávání jednotlivých databází jednotlivými programy byl ale také kladen důraz na možnosti výběru parametrů, na rychlosti prohledávání, zobrazení statistických hodnot, možnost dalších odkazu a na grafické pojetí výstupu, tudíž i celkové přehlednosti. Pro porovnání nukleotidové sekvence s nukleotidovou databází se zdá být nejvhodnější ve všech směrech program BLAST, resp. jeho modifikace BLASTN. Avšak databáze TAIR vzhledem ke konkrétnímu zaměření by měla být brána na největší zřetel.

Vzhledem k tomu, že různé programy využívají různé algoritmy, není nijak překvapující, že se výsledky liší. Ale i zde platí, že dva jsou víc než jeden – tedy že hodnotu výsledků výrazně zvyšuje shoda několika programů, především pokud jsou založeny na odlišném principu.

12 Závěr

V této práci prvním z cílů bylo utvořit si představu nad tím, co BI zahrnuje. Tento cíl byl docílen objasněním pojmů z molekulární biologie, řečením několika definic a historií, vyjmenováním nejdůležitějších veřejných biologických zdrojů dat a nástrojů na prohledávání databází. Hlavní důraz byl kladen na podrobné vysvětlení principů, na jakých vyhledávací algoritmy pracují a tím snad objasnění „černé skříňky“, za jakou by ji jinak uživatel považoval program, který použije. Tím také i pomoct se zadáváním parametrů, které jsou spíše matematického charakteru než biologického.

Další z cílů bylo porovnání databází, resp. jejich uživatelských rozhraní, co se týče spíše přehlednosti, ale i funkcí, a pomoci tak například začínajícímu biologovi s výběrem jedné z těch nejvíce navštěvovaných databází - tedy databází Velké trojky. Jako nejlepším pro začátečníky byla stanovena databáze GenBank amerického institutu NCBI, resp. její webové rozhraní Entrez.

Posledním z cílů bylo porovnání vyhledávacích programů. Ten byl docílen provedením testů, kdy byla zvolena jedna nukleotidová sekvence a ta byla základem pro 3 zvolené programy ve 4 různých databázích. Všechny 3 programy byly hodnoceny podle stejných kritérií, kde nejdůležitějším kritériem byla stanovena rychlost a také možnost dopuštění se chybného výsledku. Jako nejvhodnějším kandidátem pro prohledávání se ukázal být program BLAST opět amerického institutu NCBI.

Literatura

[1] CVRČKOVÁ, Fatima. *Úvod do praktické bioinformatiky*. [s.l.] : [s.n.], 2006. 148 s.

[2] ZENDULKA, Jaroslav. *ZZN_8_bio.pdf*. [s.l.] : [s.n.], 2006. 21 s.

[3] *Www.pmfhk.cz/Prednasky/Bioinformatika.pdf* [online]. 2007 [cit. 2009-04-20]. Dostupný z WWW: <http://www.pmfhk.cz/Prednasky/Bioinformatika.pdf>.

[4] PANTŮČEK, Roman. Využití informačních zdrojů na internetu v praktických cvičeních z bioinformatiky na Přírodovědecké fakultě MU v Brně (URL: http://orion.sci.muni.cz/kgmb/bioinformat/) . In Řehout, V. Pedagogický software. První vydání. České Budějovice : Scientific Pedagogical Publishing, 2004. ISBN 80-85645-49-1, s. 551-554. Jihočeská univerzita, České Budějovice.

[5] Www.ncbi.nlm.nih.gov/ [online]. Revised: February 4, 2009. [cit. 2009-04-20]. Dostupný z WWW: <http://www.ncbi.nlm.nih.gov/>.

[6] *Www.ebi.ac.uk/* [online]. c2009 [cit. 2009-04-20]. Dostupný z WWW: <http://www.ebi.ac.uk/>.

[7] *Arabidopsis.org* [online]. 2009 [cit. 2009-04-20]. Dostupný z WWW: <http://arabidopsis.org/>.

[8] *Www.ddbj.nig.ac.jp/* [online]. Last modified: Apr. 24, 2009 [cit. 2009-04-25]. Dostupný z WWW: <http://www.ddbj.nig.ac.jp/>.

[9]Programujte.com/index.php?akce=clanek&cl=2006030301-bioinformatika-i [online]. c2004-2009 [cit. 2009-04-20]. Dostupný z WWW: <http://programujte.com/index.php?akce=clanek&cl=2006030301-bioinformatika-i>.

[10]Www.apsnet.org/Education/K-12PlantPathways/TeachersGuide/Activities/DNA_Easy/ [online]. c2009 [cit. 2009-04-20]. Dostupný z WWW: <http://www.apsnet.org/Education/K-12PlantPathways/TeachersGuide/Activities/DNA_Easy/>.

[11] *Www.jcvi.org/* [online]. c2009 [cit. 2009-04-20]. Dostupný z WWW: <http://www.jcvi.org/>.

[12] *Www.rcsb.org/pdb/home/home.do* [online]. c2009 [cit. 2009-04-20]. Dostupný z WWW: <http://www.rcsb.org/pdb/home/home.do>.

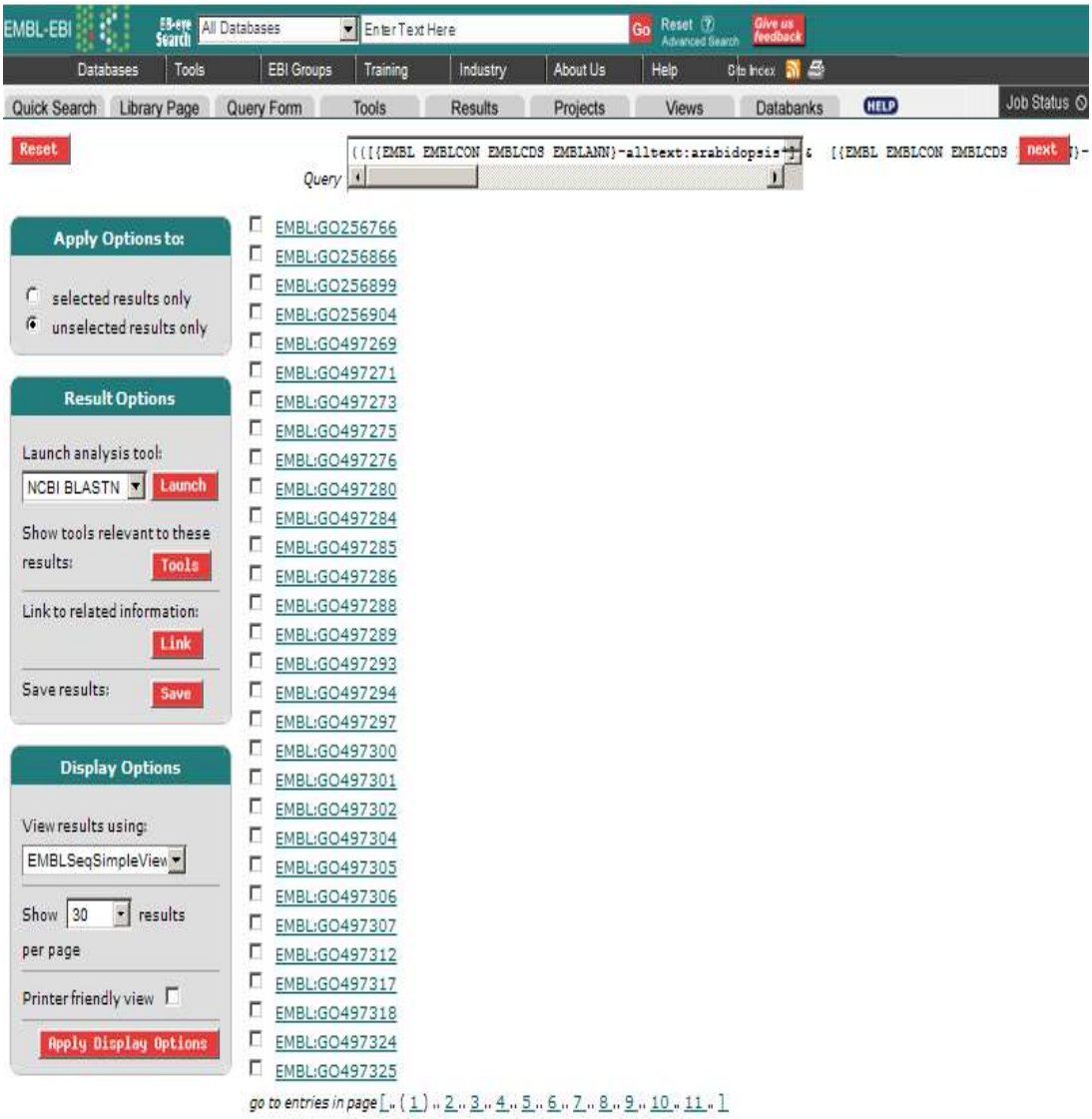
[13]Www.cbi.pku.edu.cn/docs/faq/FastaSpecifics.html [online]. c1996-2008 [cit. 2009-04-20]. Dostupný z WWW: <http://www.cbi.pku.edu.cn/docs/faq/FastaSpecifics.html>.

12.1 Seznam obrázků

12.2 Seznam tabulek

12.3 Seznam grafů

Příloha 1


EMBL-EBI
All Databases
Enter Text Here
Go
Reset
Databases
Tools
EBI Groups
Training
Industry
About Us
Help
Quick Search
Library Page
Query Form
Tools
Results
Projects
Views
Databanks
Job Status
Reset
Query
next
Apply Options to:
selected results only
unselected results only
Result Options
Launch analysis tool:
NCBI BLASTN
Launch
Show tools relevant to these results:
Tools
Link to related information:
Link
Save results:
Save
Display Options
View results using:
EMBLSeqSimpleView
Show 30 results per page
Printer friendly view
Apply Display Options
EMBL:GO256766
EMBL:GO256866
EMBL:GO256899
EMBL:GO256904
EMBL:GO497269
EMBL:GO497271
EMBL:GO497273
EMBL:GO497275
EMBL:GO497276
EMBL:GO497280
EMBL:GO497284
EMBL:GO497285
EMBL:GO497286
EMBL:GO497288
EMBL:GO497289
EMBL:GO497293
EMBL:GO497294
EMBL:GO497297
EMBL:GO497300
EMBL:GO497301
EMBL:GO497302
EMBL:GO497304
EMBL:GO497305
EMBL:GO497306
EMBL:GO497307
EMBL:GO497312
EMBL:GO497317
EMBL:GO497318
EMBL:GO497324
EMBL:GO497325
go to entries in page [« (1) » 2 » 3 » 4 » 5 » 6 » 7 » 8 » 9 » 10 » 11 »]

General Information

Primary Accession #	GO256766
Accession #	GO256766
SRS Entry ID	EMBL:GO256766
Molecule Type	linear mRNA
Sequence Length	761
Entry Division	PLN (*Plants*)
Entry Data Class	EST (*Expressed Sequence Tag*)
Sequence Version	GO256766.1
Creation Date	13-MAR-2009
Modification Date	13-MAR-2009
EMBL-SVA	GO256766

Description

Description	JACCCA1001F05.b Hymenaea courbaril mixed tissues Hymenaea courbaril var. stilbocarpa cDNA clone JACCCA1001F05 similar to AT2G28305.1 | Symbols: | similar to unknown protein [Arabidopsis thaliana] (TAIR:AT2G37210.1); similar to unnamed protein product [Vitis vinifera] (GB:CAO21862.1); contains InterPro domain Conserved hypothetical protein CHP00730 (InterPro:IPR005269), mRNA sequence.
Keywords	EST.;
Organism	Hymenaea courbaril var. stilbocarpa
Organism Classification	Eukaryota; Viridiplantae; Streptophyta; Embryophyta; Tracheophyta; Spermatophyta; Magnoliophyta; eudicotyledons; core eudicotyledons; rosids; eurosids I; Fabales; Fabaceae; Caesalpinioideae; Detarieae; Hymenaea.

References

1. Del Bem,L.E.V.; Brandao,A.D.; Costa,G.G.L.; Da Silva,M.J.; Buckeridge,M.S.; Vincentz,M.;
Transcriptome analysis of the rainforest tree Hymenaea courbaril
Unpublished.

Position 1-761

Features

Key	Location	Qualifier	Value
source	1..761	**organism**	Hymenaea courbaril var. stilbocarpa
		mol_type	mRNA
		dev_stage	45 days plantlets
		clone_lib	Hymenaea courbaril mixed tissues
		clone	JACCCA1001F05
		db_xref	taxon:497389

Sequence

Characteristics: **Length:** 761 BP, **A Count:**294, **C Count:**128, **G Count:**111, **T Count:**228, **Others Count:**0

Sequence:

```
>emb|GO256766|GO256766 JACCCA1001F05.b Hymenaea courbaril mixed tissues
gatcttgtgaatctacctaatttggtaagggatcaagcccagttatataggtttgcagtt
ggtcaaatagctgattaatagaaaagatagtcattatggtatgacgggttacttgagaca
tgaattcttgttccatagcatgatctaaaaaaggcaataaatcatcgtaaaaattattaa
tatttaacaattgcaatggctttttatgaaagtttagattggcccattatgcaatgctag
aaattccttccagtgtttctaaatcatcaaataatgcgataaaagcttcagcattgaaaa
acatgtctcatgcgtttaagcaagattgaaacctttatctcaataccataataaaagtg
agcaatgtttctatcaatcaactcctttacaacaatagccacccttaaacaattgtagaa
```

Příloha 2

FASTA Results

SUBMISSION PARAMETERS			
Title	Sequence	Database	em_rel
Sequence length	300	Sequence type	n
Program	fasta	Version	35.04 Mar. 8, 2009
Expectation upper value	10.0	Sequence range	1-
Number of scores	50	Number of alignments	50
Word size	6	Open gap penalty	-14
Gap extension penalty	-4	Histogram	false

Show Annotation | FASTA Result | MView | VisualFasta | XML | SUBMIT ANOTHER JOB

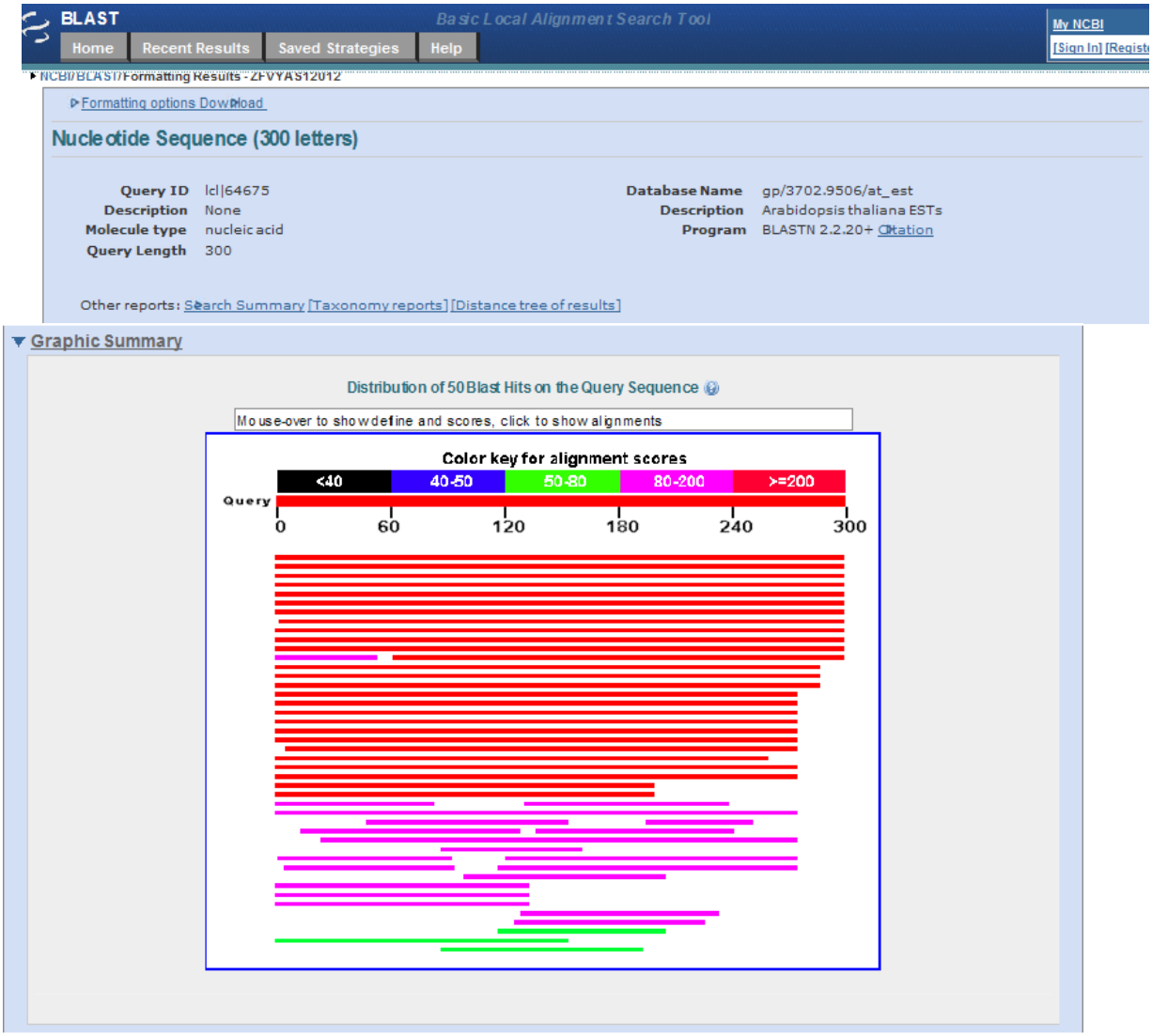
Clear all | Check all | Invert selection | Reset | Show Alignments | Download | fasta

Alignment	DBID	Source	Length	Identity%	Similar%	Overlap	E()
1	EM_PL:AC011001	Arabidopsis thaliana chromoso	91608	100.0	100.0	300	1.1e-82
2	EM_EST:AV825141	Arabidopsis thaliana cDNA cl	506	100.0	100.0	300	4.3e-82
3	EM_PL:AY072366	Arabidopsis thaliana glutared	505	100.0	100.0	300	4.3e-82
4	EM_PL:AY085047	Arabidopsis thaliana clone 12	500	100.0	100.0	300	4.3e-82
5	EM_EST:CK121725	202a18.p1 AtM1 Arabidopsis t	492	100.0	100.0	300	4.3e-82
6	EM_EST:DR243208	4494583 CERES-149 Arabidopsi	469	100.0	100.0	300	4.4e-82
7	EM_EST:DR243205	160159 CERES-149 Arabidopsis	452	100.0	100.0	300	4.4e-82
8	EM_EST:DR243206	4498544 CERES-149 Arabidopsi	452	100.0	100.0	300	4.4e-82
9	EM_EST:DR243204	4515279 CERES-149 Arabidopsi	414	100.0	100.0	300	4.5e-82
10	EM_PL:AY114614	Arabidopsis thaliana putative	373	100.0	100.0	300	4.7e-82
11	EM_EST:DR380580	974389 CERES-148 Arabidopsis	520	99.7	100.0	300	6.6e-82

kráceno

21	EM_EST:CA992195	HC0714 GIBCOBRL CAT. NO. 196	462	90.1	90.1	293	8.3e-64
22	EM_EST:AV790199	Arabidopsis thaliana cDNA cl	405	100.0	100.0	238	6.3e-63
23	EM_EST:EX033956	BR018600 callus cDNA library	478	88.2	88.2	297	4.4e-62
24	EM_PL:U93215	Arabidopsis thaliana chromosome 2	104861	82.8	82.8	291	3.8e-52
25	EM_EST:EX774560	RR4BQ78JQ RR4(PB) Raphanus r	632	83.2	83.2	291	4.1e-52
26	EM_EST:EX770353	RR4BQ78TF RR4(PB) Raphanus r	622	83.2	83.2	291	4.1e-52
27	EM_EST:EH423147	OL5863R Brassica oleracea va	507	83.2	83.2	291	4.4e-52
28	EM_EST:EH423143	OL5863F Brassica oleracea va	472	83.2	83.2	291	4.5e-52
29	EM_EST:EV226469	0108415 Brassica napus Damag	793	82.8	82.8	291	1.4e-51
30	EM_EST:EV226846	0108772 Brassica napus Damag	736	82.8	82.8	291	1.4e-51
31	EM_PL:AY127010	Arabidopsis thaliana At2g3054	685	82.8	82.8	291	1.5e-51
32	EM_EST:FD570686	RS2FH47TF RS2(RS) Raphanus s	678	82.8	82.8	291	1.5e-51
33	EM_PL:AY087910	Arabidopsis thaliana clone 39	675	82.8	82.8	291	1.5e-51
34	EM_EST:AV824247	Arabidopsis thaliana cDNA cl	644	82.8	82.8	291	1.5e-51
35	EM_HTG:BX821108	Arabidopsis thaliana Full-le	622	82.8	82.8	291	1.5e-51
36	EM_GSS:BH677747	BOMLZ63TR BO_2_3_KB Brassica	616	82.8	82.8	291	1.5e-51
37	EM_EST:AV785457	Arabidopsis thaliana cDNA cl	610	82.8	82.8	291	1.5e-51
38	EM_EST:DR376474	107855 CERES-148 Arabidopsis	563	82.8	82.8	291	1.5e-51
38	EM_EST:DR376474	107855 CERES-148 Arabidopsis	563	82.8	82.8	291	1.5e-51
39	EM_EST:FD574648	RS2FH47JQ RS2(RS) Raphanus s	545	82.8	82.8	291	1.5e-51
40	EM_PL:BT001026	Arabidopsis thaliana At2g3054	309	82.8	82.8	291	1.8e-51
41	EM_HTG:AC232518	Brassica rapa subsp. pekinen	140628	82.1	82.1	291	4.6e-51
42	EM_EST:EV552275	RR3BF91TF RR3(NY) Raphanus r	730	81.8	81.8	291	6.6e-50
43	EM_EST:FD553399	RR4EU51JQ RR4(PB) Raphanus r	664	81.8	81.8	291	6.8e-50
44	EM_EST:FD963683	RR1GH93TF RR1(CS) Raphanus r	652	81.8	81.8	291	6.8e-50
45	EM_EST:FD983599	RR4QC29JQ RR4(PB) Raphanus r	647	81.8	81.8	291	6.8e-50
46	EM_EST:FD987568	RR4QC29TF RR4(PB) Raphanus r	645	81.8	81.8	291	6.9e-50
47	EM_EST:FD536425	RR1FU09TF RR1(CS) Raphanus r	642	81.8	81.8	291	6.9e-50
48	EM_EST:FD557257	RR4EU51TF RR4(PB) Raphanus r	610	81.8	81.8	291	7e-50
49	EM_GSS:BH983648	ode20d10.b1 B.oleracea002 Br	666	82.6	82.6	288	9e-50
50	EM_EST:EV524471	RR1A130TF RR1(CS) Raphanus r	527	81.4	81.4	291	2.6e-49

Příloha 3


BLAST
Basic Local Alignment Search Tool
Home
Recent Results
Saved Strategies
Help
My NCBI
Formatting options
Nucleotide Sequence (300 letters)
Query ID lcl|64675
Description None
Molecule type nucleic acid
Query Length 300
Database Name gp/3702.9506/at_est
Description Arabidopsis thaliana ESTs
Program BLASTN 2.2.20+
Other reports: Search Summary [Taxonomy reports] [Distance tree of results]
Graphic Summary
Distribution of 50 Blast Hits on the Query Sequence
Mouse-over to show defline and scores, click to show alignments
Color key for alignment scores
<40
40-50
50-80
80-200
>=200
Query
0
60
120
180
240
300

Descriptions

Sequences producing significant alignments:		Score (Bits)	E Value	
gb|DR243208.1|	4494583 CERES-149 Arabidopsis thaliana cDNA cl...	542	5e-153	U G
gb|DR243206.1|	4498544 CERES-149 Arabidopsis thaliana cDNA cl...	542	5e-153	U G
gb|DR243205.1|	160159 CERES-149 Arabidopsis thaliana cDNA clo...	542	5e-153	U G
gb|DR243204.1|	4515279 CERES-149 Arabidopsis thaliana cDNA cl...	542	5e-153	U G
gb|CK121725.1|	202a18.p1 AtM1 Arabidopsis thaliana cDNA clone...	542	5e-153	U G
dbj|AV825141.1|	AV825141 RAFL6 Arabidopsis thaliana cDNA clon...	542	5e-153	U
gb|DR380580.1|	974389 CERES-148 Arabidopsis thaliana cDNA clo...	538	7e-152	U
gb|BE038202.1|	AA10C08 AA Arabidopsis thaliana cDNA 5' simila...	538	7e-152	U G
gb|DR243210.1|	961841 CERES-148 Arabidopsis thaliana cDNA clo...	533	3e-150	U G
gb|DR243209.1|	135441 CERES-149 Arabidopsis thaliana cDNA clo...	533	3e-150	U G
gb|DR243207.1|	158013 CERES-148 Arabidopsis thaliana cDNA clo...	533	3e-150	U
dbj|AV790199.1|	AV790199 RAFL6 Arabidopsis thaliana cDNA clon...	430	2e-119	U G
gb|DR376474.1|	107855 CERES-148 Arabidopsis thaliana cDNA clo...	302	9e-81	U G
dbj|AV824247.1|	AV824247 RAFL6 Arabidopsis thaliana cDNA clon...	302	9e-81	U G
dbj|AV785457.1|	AV785457 RAFL6 Arabidopsis thaliana cDNA clon...	302	9e-81	U G
gb|DR234773.1|	1105490 CERES-147 Arabidopsis thaliana cDNA cl...	212	1e-53	U
gb|DR234771.1|	55487 CERES-147 Arabidopsis thaliana cDNA clon...	212	1e-53	U G
gb|DR234770.1|	1133313 CERES-147 Arabidopsis thaliana cDNA cl...	212	1e-53	U G
gb|DR234769.1|	64417 CERES-147 Arabidopsis thaliana cDNA clon...	212	1e-53	U G
gb|DR234768.1|	1146901 CERES-147 Arabidopsis thaliana cDNA cl...	212	1e-53	U G
gb|DR234767.1|	1142609 CERES-147 Arabidopsis thaliana cDNA cl...	212	1e-53	U G
dbj|AV531820.1|	AV531820 Arabidopsis thaliana flower buds Col...	208	1e-52	U G
gb|T04053.1|	3 Lambda-PRL1 Arabidopsis thaliana cDNA clone S1...	208	1e-52	U E G
dbj|C99862.1|	C99862 Arabidopsis thaliana library (Motohashi ...	206	5e-52	U G
gb|EH951480.1|	EBENXNS02IKAVH 8-day Arabidopsis seedlings, ae...	96.9	7e-19	G
gb|EL022466.1|	EBENXNS02GR5AK 8-day Arabidopsis seedlings, ae...	86.0	1e-15	U G
gb|EH961586.1|	EBENXNS02JKQN1 8-day Arabidopsis seedlings, ae...	86.0	1e-15	G
gb|EH892040.1|	EBENXNS01BU6R8 8-day Arabidopsis seedlings, ae...	82.4	1e-14	U G
gb|EH832715.1|	EBENXNS01BEVKC 8-day Arabidopsis seedlings, ae...	82.4	1e-14	G
gb|EH921432.1|	EBENXNS01A3KPY 8-day Arabidopsis seedlings, ae...	78.8	2e-13	U G
emb|BX835042.1|	BX835042 Arabidopsis thaliana Silique Col-0 A...	78.8	2e-13	
gb|EL201746.1|	EB3RODY02GCWFD 8-day Arabidopsis seedlings, ae...	77.0	6e-13	U G

kráceno

Alignments

Se All Get selected sequences Distance tree of results Multiple alignment

>DR243208.1| 44U G3 CERES-149 Arabidopsis thaliana cDNA clone 1006222 5', mRNA sequence.
Length=469

Score = 542 bits (600), Expect = 5e-153
Identities = 300/300 (100%), Gaps = 0/300 (0%)
Strand=Plus/Plus

```
Query  1    ATGGACAAAGTTATGAGAATGTCGTCCGAAAAAGGGGTGGTTATATTTACCAAGAGCTCC  60
            ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  34   ATGGACAAAGTTATGAGAATGTCGTCCGAAAAAGGGGTGGTTATATTTACCAAGAGCTCC  93

Query  61   TGTTGTTTGTCCTATGCGGTTCAAGTTCTCTTCCAAGATCTTGGTGTTAACCCTAAGATC  120
            ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  94   TGTTGTTTGTCCTATGCGGTTCAAGTTCTCTTCCAAGATCTTGGTGTTAACCCTAAGATC  153

Query  121  CACGAGATTGATAAGGACCCTGAATGCCGAGAGATAGAGAAGGCTCTTATGAGGCTAGGG  180
            ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  154  CACGAGATTGATAAGGACCCTGAATGCCGAGAGATAGAGAAGGCTCTTATGAGGCTAGGG  213

Query  181  TGTTCAAAGCCGGTCCCAGCCGTCTTCATTGGTGGCAAGCTCGTTGGTTCGACCAACGAA  240
            ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  214  TGTTCAAAGCCGGTCCCAGCCGTCTTCATTGGTGGCAAGCTCGTTGGTTCGACCAACGAA  273

Query  241  GTAATGTCCATGCACCTAAGCAGCTCGCTCGTTCCCCTAGTGAAGCCATATTTATGTTAA  300
            ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  274  GTAATGTCCATGCACCTAAGCAGCTCGCTCGTTCCCCTAGTGAAGCCATATTTATGTTAA  333
```

Příloha 4

```
Your request to DDBJ SSEARCH service has finished.

Request ID: 20090425223500_27202


Your query is
  * * * * * * * *
 > query
 ATGGACAAAGTTATGAGAATGTCGTCCGAAAAAGGGGTGGTTATATTTAC
 CAAGAGCTCCTGTTGTTTGTCCTATGCGGTTCAAGTTCTCTTCCAAGATC
 TTGGTGTTAACCCTAAGATCCACGAGATTGATAAGGACCCTGAATGCCGA
 GAGATAGAGAAGGCTCTTATGAGGCTAGGGTGTTCAAAGCCGGTCCCAGC
 CGTCTTCATTGGTGGCAAGCTCGTTGGTTCGACCAACGAAGTAATGTCCA
 TGCACCTAAGCAGCTCGCTCGTTCCCCTAGTGAAGCCATATTTATGTTAA


  * * * * * * * *

SSEARCH searches a sequence data bank
 version 3.4t26 July 7, 2006
Please cite:
 T. F. Smith and M. S. Waterman, (1981) J. Mol. Biol. 147:195-197;
 W.R. Pearson (1991) Genomics 11:635-650

Query library /result/ssearch/query/20090425223500_27202 vs @/result/ssearch/tmp/20090425223500_27202.nam library
searching .../a/DNA.DATA/est_atha1.seq 0 library

  1>>> query 300 nt - 300 nt
 vs  @/result/ssearch/tmp/20090425223500_27202.nam library

       opt      E()
 < 20   83     0:=
   22   85     0:=          one = represents 2511 library sequences
   24   54     1:*
   26  120    32:*
   28  348   346:*
   30 1351  2104:*
   32 4180  8134:== *
 36 28628 45304:============       *
 38 55418 74870:========================       *
 40 88584 104437:====================================     *
 42 121333 127662:================================================== *
 44 140814 140823:============================================================*
 46 148047 143432:============================================================*=
 48 150660 137319:============================================================*=====
 50 134031 125305:==================================================*====
 52 120608 110164:================================================*======
 54 101062 94099:=================================*===
 56 85416 78602:=================================*===
 58 71995 64530:==========================*===
 60 58507 52273:====================*===
 62 47818 41908:=================*===
 64 36719 33329:==============*=
 66 27287 26342:==========*
 68 20918 20720:=========*
 70 17029 16238:=======*
 72 12615 12688:======*
 74  9727  9893:===*
 76  7536  7700:===*
 78  5426  5985:==*
 80  4094  4647:=*
 82  3125  3555:=*
 84  2659  2816:=*
 86  2627  2179:*=
 88  1619  1686:*          inset = represents 15 library sequences
 90  1413  1304:*
 92   733  1009:*         :=======================================*
 94   616   781:*         :=======================================*
 96   425   604:*         :============================          *
 98   348   468:*         :========================       *
100   282   362:*         :====================     *
102   188   280:*         :=============     *
104   124   217:*         :==========     *
106    87   168:*         :======     *
108   138   130:*         :=========*=
110    87   100:*         :=======*
```

```
 118    24    36:*          :==*
>120   361    28:*          :**************************
399322087 residues in 1527298 sequences
 statistics sampled from 60000 to 1526947 sequences
  Expectation_n fit: rho(ln(x))= 8.4081+/-0.00016; mu= 14.4421+/- 0.009
 mean_var=67.5483+/-17.648, 0's: 2 Z-trim: 19  B-trim: 0 in 0/45
 Lambda= 0.156051
 Kolmogorov-Smirnov  statistic: 0.0478 (N=29) at  44

 Smith-Waterman (PGopt) (5.2 May 2006) function [+5/-4 matrix (5:-4)], open/ext: -12/-4
 Scan time: 1229.680
The best scores are:                                          s-w bits E(1527298)
DR243204|DR243204.1 4515279 CERES-149 Arabidop ( 414) [f] 1500 340.8 3.1e-92
DR243206|DR243206.1 4498544 CERES-149 Arabidop ( 452) [f] 1500 340.8 3.2e-92
DR243205|DR243205.1 160159 CERES-149 Arabidops ( 452) [f] 1500 340.8 3.2e-92
DR243208|DR243208.1 4494583 CERES-149 Arabidop ( 469) [f] 1500 340.7 3.2e-92
CK121725|CK121725.1 202a18.p1 AtM1 Arabidopsis ( 492) [f] 1500 340.7 3.2e-92
AV825141|AV825141.1 Arabidopsis thaliana cDNA  ( 506) [f] 1500 340.7 3.3e-92
DR380580|DR380580.1 974389 CERES-148 Arabidops ( 520) [f] 1497 340.0 5.3e-92
BE038202|BE038202.1 AA10C08 AA Arabidopsis tha ( 443) [f] 1490 338.5 1.5e-91
DR243209|DR243209.1 135441 CERES-149 Arabidops ( 468) [f] 1479 336.0 8.5e-91
DR243210|DR243210.1 961841 CERES-148 Arabidops ( 476) [f] 1479 336.0 8.5e-91
DR243207|DR243207.1 158013 CERES-148 Arabidops ( 500) [f] 1479 336.0 8.6e-91
AV790199|AV790199.1 Arabidopsis thaliana cDNA  ( 405) [r] 1190 271.0 3.1e-71
DR376474|DR376474.1 107855 CERES-148 Arabidops ( 563) [f] 1005 229.2 1.2e-58
AV785457|AV785457.1 Arabidopsis thaliana cDNA  ( 610) [r] 1005 229.2 1.2e-58
AV824247|AV824247.1 Arabidopsis thaliana cDNA  ( 644) [f] 1005 229.2 1.2e-58
DR234769|DR234769.1 64417 CERES-147 Arabidopsi ( 500) [f]  825 188.7 1.8e-46
DR234767|DR234767.1 1142609 CERES-147 Arabidop ( 500) [f]  825 188.7 1.8e-46
DR234768|DR234768.1 1146901 CERES-147 Arabidop ( 501) [f]  825 188.7 1.8e-46
DR234773|DR234773.1 1105490 CERES-147 Arabidop ( 504) [f]  825 188.7 1.8e-46
DR234771|DR234771.1 55487 CERES-147 Arabidopsi ( 516) [f]  825 188.7 1.8e-46
DR234770|DR234770.1 1133313 CERES-147 Arabidop ( 554) [f]  825 188.7 1.9e-46
C99862|C99862.1 Arabidopsis thaliana YAC CIC3B ( 474) [r]  812 185.8 1.4e-45
T04053|T04053.1 3 Lambda-PRL1 Arabidopsis thal ( 460) [f]  809 185.2 2.2e-45
AV531820|AV531820.1 Arabidopsis thaliana cDNA  ( 532) [r]  809 185.1 2.3e-45
DR373309|DR373309.1 102504 CERES-147 Arabidops ( 587) [f]  804 183.9 5.1e-45
DR234772|DR234772.1 6912847 CERES-AS12 Arabido ( 374) [f]  788 180.5 5.3e-44
AV533728|AV533728.1 Arabidopsis thaliana cDNA  ( 402) [f]  733 168.1 2.9e-40
DR344603|DR344603.1 4512360 CERES-148 Arabidop ( 433) [f]  712 163.4 7.9e-39
```

kráceno

Go to top

```
>>DR243204|DR243204.1 4515279 CERES-149 Arabidopsis thal  (414 nt)
 s-w opt: 1500  Z-score: 1795.9  bits: 340.8 E(): 3.1e-92
Smith-Waterman score: 1500;  100.000% identity (100.000% similar) in 300 nt overlap (1-300:34-333)

                                        10        20        30
                                ATGGACAAAGTTATGAGAATGTCGTCCGAA
                                ::::::::::::::::::::::::::::::
DR2432 TACTTCTTCTTCTTCACCTTATGCAAGATAATGGACAAAGTTATGAGAATGTCGTCCGAA
            10        20        30        40        50        60

               40        50        60        70        80        90
       AAAGGGGTGGTTATATTTACCAAGAGCTCCTGTTGTTTGTCCTATGCGGTTCAAGTTCTC
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
DR2432 AAAGGGGTGGTTATATTTACCAAGAGCTCCTGTTGTTTGTCCTATGCGGTTCAAGTTCTC
            70        80        90       100       110       120

              100       110       120       130       140       150
       TTCCAAGATCTTGGTGTTAACCCTAAGATCCACGAGATTGATAAGGACCCTGAATGCCGA
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
DR2432 TTCCAAGATCTTGGTGTTAACCCTAAGATCCACGAGATTGATAAGGACCCTGAATGCCGA
           130       140       150       160       170       180

              160       170       180       190       200       210
       GAGATAGAGAAGGCTCTTATGAGGCTAGGGTGTTCAAAGCCGGTCCCAGCCGTCTTCATT
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
DR2432 GAGATAGAGAAGGCTCTTATGAGGCTAGGGTGTTCAAAGCCGGTCCCAGCCGTCTTCATT
           190       200       210       220       230       240

              220       230       240       250       260       270
       GGTGGCAAGCTCGTTGGTTCGACCAACGAAGTAATGTCCATGCACCTAAGCAGCTCGCTC
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
DR2432 GGTGGCAAGCTCGTTGGTTCGACCAACGAAGTAATGTCCATGCACCTAAGCAGCTCGCTC
           250       260       270       280       290       300

              280       290       300
       GTTCCCCTAGTGAAGCCATATTTATGTTAA
       ::::::::::::::::::::::::::::::
DR2432 GTTCCCCTAGTGAAGCCATATTTATGTTAAACAACAACGAAGGAGTATTTATGATATTAA
           310       320       330       340       350       360
```

Příloha 5

TAIR Fasta Search Results

Reference: Pearson, W.R. and D.J. Lipman (1988) Improved tools for biological sequence comparison. Proc. Natl. Acad. Sci. U.S.A. 85:2444-2448 [Medline]

Query sequence:

```
  (Length: 300)
ATGGACAAAGTTATGAGAATGTCGTCCGAAAAAGGGGTGGTTATATTTACCAAGAGCTCCTGTTGTTTGT
CCTATGCGGTTCAAGTTCTCTTCCAAGATCTTGGTGTTAACCCTAAGATCCACGAGATTGATAAGGACCC
TGAATGCCGAGAGATAGAGAAGGCTCTTATGAGGCTAGGGTGTTCAAAGCCGGTCCCAGCCGTCTTCATT
GGTGGCAAGCTCGTTGGTTCGACCAACGAAGTAATGTCCATGCACCTAAGCAGCTCGCTCGTTCCCCTAG
TGAAGCCATATTTATGTTAA
```

Dataset to search: TAIR8 Transcripts (-introns, +UTRs) (DNA)

>>>Queued
>>>Running****

Results:

```
fasta34_t  version 3.4t26 July 7, 2006
Histogram of scores
       opt      E()

< 20     0     0:
  22     0     0:           one = represents 60 library sequences
  24     0     0:
  26     0     1:*
  28     2     9:*
  30    42    54:*
  32   188   207:===*
  34   695   563:==========*==
  36  1302  1155:====================*==
  38  2212  1910:==============================*======
  40  2875  2664:================================================*===
  42  3258  3256:============================================================*
  44  3533  3592:==============================================================*
  46  3572  3658:==============================================================*
  48  3425  3502:============================================================*
  50  3147  3196:==========================================================*
  52  2616  2810:================================================  *
  54  2304  2400:==========================================*
  56  1977  2005:==================================*
  58  1624  1646:============================*
  60  1283  1333:======================*
  62  1073  1069:==================*
  64   824   850:===============*
  66   653   672:===========*
  68   553   528:=========*=
  70   413   414:=======*
  72   330   324:======*
  74   248   252:====*
  76   177   196:===*
  78   150   153:===*
  80   107   119:=*
  82    89    91:=*
  84    68    72:=*
  86    41    56:*
  88    36    43:*          inset = represents 1 library sequences
  90    26    33:*
  92    23    26:*         :======================  *
  94     7    20:*         :=======             *
  96    15    15:*         :==============*
  98    11    12:*         :============*
 100    15     9:*         :=========*======
 102     2     7:*         :==    *
 104     4     6:*         :==== *
 106     8     4:*         :===*====
 108     4     3:*         :==*=
 110     5     3:*         :==*==
 112     4     2:*         :=*==
 114     1     2:*         :=*
 116     2     1:*         :*=
 118     0     1:*         :*
>120    19     1:*         :*==================
59766823 residues in 38963 sequences
  Expectation_n fit: rho(ln(x))= 5.0856+/-0.00028; mu= 22.9041+/- 0.020
 mean_var=103.3480+/-21.693, 0's: 0 Z-trim: 17  B-trim: 0 in 0/56
 Lambda= 0.126160
 Kolmogorov-Smirnov  statistic: 0.0194 (N=29) at  42
```

```
The best scores are:                                      opt bits E(38963)
AT1G06830.1 | Symbols:  | glutaredoxin family  ( 505) [f] 1500 281.1   4.2e-75
AT2G30540.1 | Symbols:  | glutaredoxin family  ( 680) [f] 1005 191.2   5e-48
AT2G47880.1 | Symbols:  | glutaredoxin family  ( 632) [f]  834 160.0   1.2e-38
AT3G62960.1 | Symbols:  | glutaredoxin family  ( 615) [f]  807 155.1   3.7e-37
AT3G62950.1 | Symbols:  | glutaredoxin family  ( 686) [f]  451 90.4    1.1e-17
AT2G47870.1 | Symbols:  | glutaredoxin family  ( 312) [f]  415 83.4    1.4e-15
AT4G15690.1 | Symbols:  | glutaredoxin family  ( 598) [f]  413 83.4    1.4e-15
AT4G15700.1 | Symbols:  | glutaredoxin family  ( 500) [f]  413 83.3    1.5e-15
AT4G15660.1 | Symbols:  | glutaredoxin family  ( 505) [f]  404 81.6    4.7e-15
AT5G18600.1 | Symbols:  | glutaredoxin family  ( 697) [f]  395 80.2    1.3e-14
AT4G15670.1 | Symbols:  | glutaredoxin family  ( 587) [f]  386 78.4    4.4e-14
AT3G21460.1 | Symbols:  | electron carrier/ pr ( 575) [f]  324 67.1    1.1e-10
AT3G62930.1 | Symbols:  | glutaredoxin family  ( 486) [f]  299 62.5    2.7e-09
AT5G14070.1 | Symbols: ROXY2 | ROXY2; thiol-di ( 679) [f]  216 47.6    8.5e-05
AT3G02000.1 | Symbols: ROXY1 | ROXY1; thiol-di ( 748) [f]  209 46.4    0.0002
AT3G44205.1 | Symbols:  | transposable element (2301) [r]  151 36.4    0.2
AT1G28480.1 | Symbols: GRX480 | GRX480; thiol- ( 600) [f]  148 35.1    0.47
AT1G61480.1 | Symbols:  | S-locus protein kina (2605) [r]  138 34.1    0.99
AT3G23410.1 | Symbols:  | alcohol oxidase-rela (2420) [f]  137 33.9    1.2
AT1G16660.1 | Symbols:  | transposable element (4435) [f]  128 32.5    2.9
AT1G61440.1 | Symbols:  | S-locus protein kina (2379) [r]  129 32.4    3.2
AT1G61360.1 | Symbols:  | S-locus lectin prote (2647) [r]  127 32.1    3.9
AT1G55130.1 | Symbols:  | endomembrane protein (2041) [f]  126 31.8    4.9
AT1G03990.1 | Symbols:  | alcohol oxidase-rela (2277) [f]  125 31.6    5.4
AT3G43960.1 | Symbols:  | cysteine proteinase, (1227) [r]  126 31.5    5.9
AT5G56890.1 | Symbols:  | protein kinase famil (3732) [f]  121 31.2    7.4
AT3G50370.1 | Symbols:  | similar to unnamed p (7162) [f]  119 31.1    7.6
AT4G08710.1 | Symbols:  | transposable element (2148) [f]  122 31.1    8
AT5G06220.1 | Symbols:  | similar to unknown p (2719) [f]  121 31.0    8.3
AT5G40820.1 | Symbols: ATR, ATATR, ATRAD3 | AT (9189) [f]  117 30.9    8.9
AT2G39810.1 | Symbols: HOS1 | HOS1 (High expre (3180) [f]  120 30.9    8.9
AT1G61430.1 | Symbols:  | S-locus protein kina (2909) [r]  120 30.9    9.2
AT1G11280.2 | Symbols:  | S-locus protein kina (2840) [r]  120 30.8    9.3
AT1G11280.3 | Symbols:  | S-locus protein kina (2804) [r]  120 30.8    9.3
AT1G11280.1 | Symbols:  | S-locus protein kina (2699) [r]  120 30.8    9.5
AT1G11280.4 | Symbols:  | S-locus protein kina (2663) [r]  120 30.8    9.5
AT2G39880.1 | Symbols: AtMYB25, MYB25 | MYB25  (1104) [f]  121 30.5    11
AT5G46590.1 | Symbols: ANAC096 | ANAC096 (Arab (1270) [f]  119 30.2    14
AT1G03980.1 | Symbols: ATPCS2 | ATPCS2 (PHYTOC (2337) [f]  117 30.2    15
AT1G03850.1 | Symbols:  | glutaredoxin family  ( 887) [f]  118 29.9    18
AT3G08840.1 | Symbols:  | D-alanine--D-alanine (1785) [f]  116 29.9    18
AT5G23110.1 | Symbols:  | zinc finger (C3HC4-t (14283) [r]  110 29.9    18
AT1G66235.1 | Symbols:  | similar to unknown p ( 798) [r]  118 29.8    19
AT1G03850.2 | Symbols:  | glutaredoxin family  ( 660) [f]  118 29.7    20
AT2G22807.1 | Symbols:  | Encodes a defensin-l ( 282) [f]  119 29.5    24
AT4G09660.1 | Symbols:  | similar to hAT dimer (1995) [r]  113 29.4    26
```

kráceno

```
>>AT1G06830.1 | Symbols:  | glutaredoxin family protein   (505 nt)
 initn: 1500 init1: 1500 opt: 1500  Z-score: 1471.8  bits: 281.1 E(): 4.2e-75
banded Smith-Waterman score: 1500;  100.000% identity (100.000% similar) in 300 nt overlap (1-300:39-338)

                                        10        20        30
                                    ATGGACAAAGTTATGAGAATGTCGTCCGAA
                                    ::::::::::::::::::::::::::::::
AT1G06 TACTTCTTCTTCTTCACCTTATGCAAGATAATGGACAAAGTTATGAGAATGTCGTCCGAA
       10        20        30        40        50        60

               40        50        60        70        80        90
       AAAGGGGTGGTTATATTTACCAAGAGCTCCTGTTGTTTGTCCTATGCGGTTCAAGTTCTC
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
AT1G06 AAAGGGGTGGTTATATTTACCAAGAGCTCCTGTTGTTTGTCCTATGCGGTTCAAGTTCTC
       70        80        90        100       110       120

              100       110       120       130       140       150
       TTCCAAGATCTTGGTGTTAACCCTAAGATCCACGAGATTGATAAGGACCCTGAATGCCGA
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
AT1G06 TTCCAAGATCTTGGTGTTAACCCTAAGATCCACGAGATTGATAAGGACCCTGAATGCCGA
       130       140       150       160       170       180

              160       170       180       190       200       210
       GAGATAGAGAAGGCTCTTATGAGGCTAGGGTGTTCAAAGCCGGTCCCAGCCGTCTTCATT
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
AT1G06 GAGATAGAGAAGGCTCTTATGAGGCTAGGGTGTTCAAAGCCGGTCCCAGCCGTCTTCATT
       190       200       210       220       230       240

              220       230       240       250       260       270
       GGTGGCAAGCTCGTTGGTTCGACCAACGAAGTAATGTCCATGCACCTAAGCAGCTCGCTC
       ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
AT1G06 GGTGGCAAGCTCGTTGGTTCGACCAACGAAGTAATGTCCATGCACCTAAGCAGCTCGCTC
       250       260       270       280       290       300

              280       290       300
       GTTCCCCTAGTGAAGCCATATTTATGTTAA
       ::::::::::::::::::::::::::::::
AT1G06 GTTCCCCTAGTGAAGCCATATTTATGTTAAACAACAACGAAGGAGTATTTATGATATTAA
       310       320       330       340       350       360
```