دانشگاه پیام نور
دانشکده فنی و مهندسی

پایان‌نامه
برای دریافت درجه کارشناسی ارشد
رشته مهندسی کامپیوتر - گرایش نرم‌افزار
گروه فناوری اطلاعات و ارتباطات

# پیشنهاد یک الگوریتم موازی و آگاه از حافظه نهان برای ساخت درخت پسوندی مربوط به دنباله‌های DNA

معصومه السادات علوی

استاد راهنما:
جناب آقای دکتر معظم

استاد مشاور:
جناب آقای دکتر سید علی رضوی

شهریور ۱۳۹۱

دانشگاه پیام نور
دانشکده فنی و مهندسی
دانشگاه پیام نور مرکز تهران

پایان‌نامه
برای دریافت درجه کارشناسی ارشد
رشته مهندسی کامپیوتر -گرایش نرم‌افزار
گروه فناوری اطلاعات و ارتباطات

# پیشنهاد یک الگوریتم موازی و آگاه از حافظه نهان برای ساخت درخت پسوندی مربوط به دنباله‌های DNA

معصومه السادات علوی

استاد راهنما:
جناب آقای دکتر معظم

استاد مشاور:
جناب آقای دکتر سید علی رضوی

شهریور ۱۳۹۱


بسم الله الرحمن الرحيم

جمهوری اسلامی ایران
وزارت علوم تحقیقات و فناوری

مرکز شمیرانات

دانشگاه پیام نور

دانشگاه پیام نور استان تهران

تاریخ .....
شماره .....
پیوست .....

# تصویب نامه

**پایان نامه کارشناسی ارشد رشته مهندسی کامپیوتر(نرم افزار)**

**تحت عنوان:**

**"پیشنهاد یک الگوریتم موازی و آگاه از کش (cash) برای ساخت درخت پسوندی مربوط به دنباله های DNA"**

**تاریخ دفاع :۱۳۹۱/۰۶/۳۰** **ساعت: ۱۲-۱۱**

**نمره:** ۱۹/- **درجه ارزشیابی:** عالی

**هیات داوران:**

| داوران | نام و نام خانوادگی | مرتبه علمی | امضاء |
|---|---|---|---|
| استاد راهنما | دکتر محمدهادی معظم | استادیار | |
| استاد مشاور | دکتر سید علی رضوی ابراهیمی | استادیار | |
| استاد داور | دکتر مهدی جوانمرد | استادیار | |
| نماینده گروه | دکتر مهدی جوانمرد | استادیار | |

تهران-بزرگراه ارتش-انتهای بلوار شهید مژدی(اوشان) خیابان شهید پیروز شفیعی خیابان یاران-خیابان یاران دوم دانشگاه پیام نور مرکز شمیرانات

تلفن:٤-۲۲۱۹۵۳۰۳

دورنگار:۲۲٤۸٤۸۳٤

http://teh-shemiranat.pnu.ac.ir

info@fani.tpnu.ac.ir

**تقدیم به:**

**مادر فداکارم که مهر بیکرانش را نمی‌توانم جبران کنم.**

**پدر عزیزم که همیشه مدیون محبت‌هایش هستم.**

**همسر مهربانم که یاری‌هایش تکیه‌گاهم در مراحل زندگی است.**

## تشکر و قدردانی:

در ابتدا پروردگار را به خاطر لطف و توجهی که در مراحل مختلف زندگی شامل حالم نموده شاکرم. به عنوان وظیفه، لازم است از استادان و دوستانی که در تدوین این تحقیق مرا یاری نموده‌اند تشکر نمایم.

از استاد راهنمای گران‌قدر جناب آقای دکتر معظم که همواره از توصیه‌های ایشان بهره‌مند شدم کمال تشکر را دارم. همچنین جا دارد از زحمات استاد مشاور محترم جناب آقای دکتر رضوی سپاسگزاری نمایم. بعلاوه توفیق آن را داشته‌ام که از نظرات جناب آقای دکتر پارسا، برخوردار شوم که از صمیم قلب از ایشان سپاسگزاری می‌نمایم. نیز از آقای دکتر حمزه‌ای به عنوان نخستین الهام بخش در انتخاب موضوع ساخت درخت‌های پسوندی که زمینه‌ساز این تحقیق شده است، قدردانی نمایم. همچنین از آقای مهندس کهندانی که در پیاده‌سازی این تحقیق کمک بسیار بزرگی بوده‌اند، کمال تشکر را دارم.

از حمایت بی‌دریغ تک‌تک افراد خـــانواده که تدوین این تحقیق را برایم فراهم نموده‌اند نهایت سپاسگزاری را دارم.

## چکیده

درخت پسوندی ساختار داده‌ای است که تمامی پسوندهای یک رشته را نمایش می‌دهد. به کمک تشکیل این ساختار داده‌ای می‌توان اعمالی نظیر یافتن یک زیر رشته را در دنباله‌های طولانی مانند DNA به طور کلی در دنباله‌ها انجام داد. الگوریتم‌های متفاوتی برای ساخت درخت پسوندی وجود دارد. الگوریتم‌های اولیه، تشکیل درخت پسوندی را در زمان خطی ممکن می‌ساخت، اما برای دنباله‌های طولانی‌تر این الگوریتم‌ها دارای کارایی لازم نیستند چرا که دنباله ورودی و درخت حاصل در حافظه اصلی جای نمی‌گیرند و در نتیجه دست‌یابی‌هایی به دیسک وجود خواهد داشت. بنابراین تمرکز الگوریتم‌های بعدی در کاهش این دست‌یابی‌ها از طریق ارائه روش‌هایی برای جای دادن درخت پسوندی حاصل در حافظه بوده است.

در این تحقیق الگوریتم موازی برای ساخت درخت پسوندی ارائه شده است. این الگوریتم درخت پسوندی را به جای حافظه اصلی در کش جای می‌دهد. همچنین موازی بودن آن امکان اجرای سریع‌تر الگوریتم را ممکن می‌سازد.



**واژه‌های کلیدی:** *درخت پسوندی، اندیس‌گذاری دنباله‌ها، الگوریتم موازی.*

# فهرست مطالب

**عنوان** **صفحه**

# فهرست اشکال

## عنوان صفحه

# فهرست جداول

**عنوان** **صفحه**

## فهرست علائم اختصاری

| | | |
|---|---|---|
| DFS | Depth First Search | جستجوی اول-عمق |
| LCA | Least Common Ancestor | نزدیک‌ترین جد مشترک |
| LCP | Longest Common Prefix | طولانی‌ترین پیشوند مشترک |
| p($v$) | parent($v$) | والد گره‌ای به نام $v$ |
| PCST | Parallel and Cache-Aware Suffix Tree Construction | ساخت موازی و آگـاه از حافظـه نهان درخت پسوندی |
| SL | Suffix Link | زنجیره پسوندی |

# فصل۱

# مقدمـــــه

## ۱-۱. مقدمه

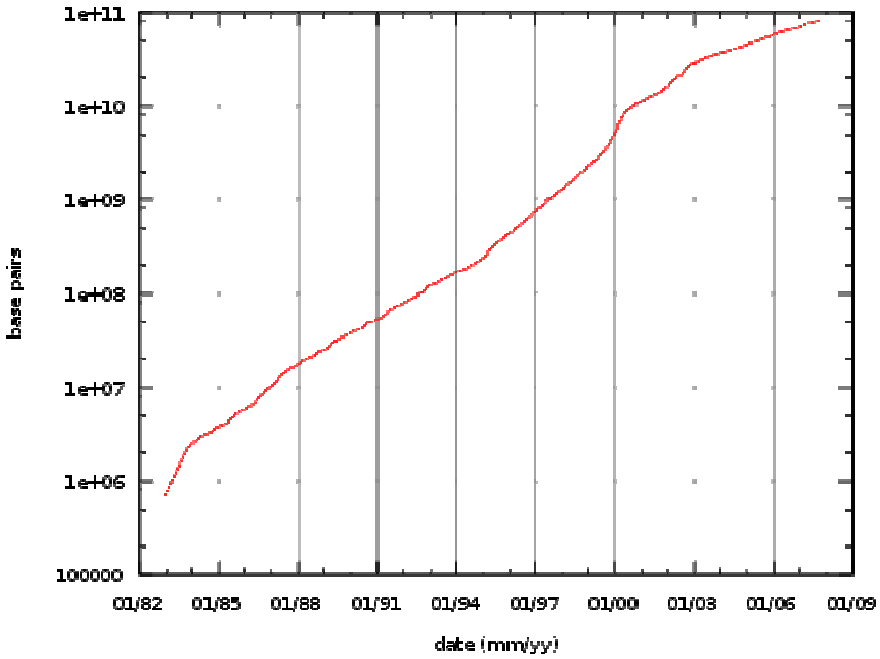
بیوانفورماتیک یا تحقیقات زیستی، دانش استفاده از علوم کامپیوتر در شاخه زیست‌شناسی مولکولی است. استفاده عمده از این واژه حداقل تا اواخر دهه ۱۹۸۰ در ژنتیک و مخصوصاً در حوزه‌هایی از ژنتیک شامل توالی DNA (تشخیص ترتیب نوکلئوتیدها A، G، C و T) بوده است (Bioinformatics, 2011). پروژه‌های مربوط به حوزه ژنتیک بسیار رایج است، بعنوان نمونه، پروژه شناسایی ژنوم انسان (وانگ[1] و گوو [2]، ۲۰۰۱) در سال ۱۹۹۰ آغاز شد و در سال ۲۰۰۳ پایان یافت و اکنون اطلاعات کاملی مربوط به توالی همه کروموزوم‌های انسان موجود می‌باشد (بیوانفورماتیک, 2011).

امروزه بیوانفورماتیک شامل ایجاد و بهبود پایگاه داده‌ها و الگوریتم‌های ناشی از تحلیل داده‌های زیستی می‌شود. توسعه سریع فناوری‌های مربوط به ژنتیک در دهه‌های اخیر، سبب ایجاد میزان زیادی از اطلاعات درباره ژنوم (محتوای ژنتیکی) انسان‌ها، حیوانات و گیاهان و در نتیجه ایجاد پایگاه داده‌های بزرگ در این حوزه شده است. به طوری که اندازه پایگاه داده مربوط به ژن‌ها (GenBank) هر ۱۸ ماه دو برابر می‌شود (GenBank release notes, 2011). رشد نمایی اندازه این پایگاه داده در شکل ۱-۱ نشان داده شده است.

با توجه به آنچه درباره حجم پایگاه داده حاوی ژنوم موجودات زنده بیان شد، به راحتی می‌توان فهمید که اعمالی مانند جستجو در این پایگاه داده‌ها به موضوعی بحرانی تبدیل می‌شود. بنابراین برای استفاده کارا از این پایگاه داده‌ها، بکارگیری الگوریتم‌های کارآمد امری ضروری است. در نتیجه اندیس‌گذاری دنباله‌های موجود در این پایگاه داده‌ها، به طراحی الگوریتم‌های کارآمد در حل مسئله در ۷ کمک خواهد کرد.

نکته‌ای که باید به آن در اینجا اشاره شود این است که توالی DNA یک دنباله است و نه یک رشته؛ چیزی که این دو را از یکدیگر متمایز می‌کند این است که ممکن است در رشته

[1] Wang, J.
[2] Gu, J.




**شکل ۱- ۱.** رشد نمایی اندازه **GenBank**

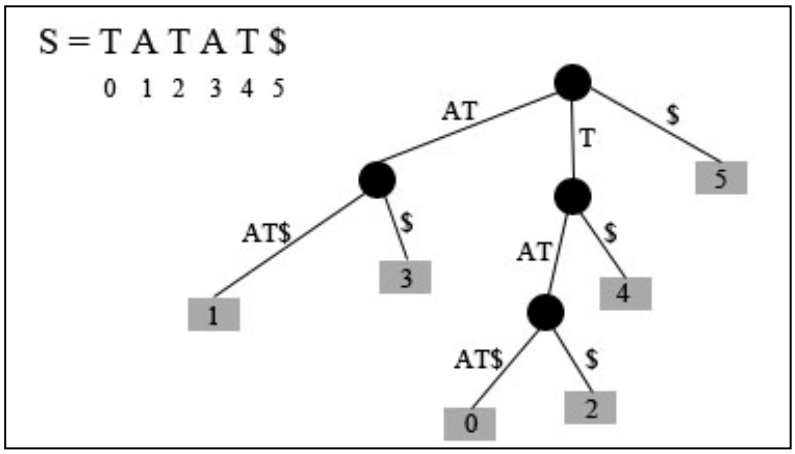
فاصله‌هایی وجود داشته باشد، این در حالی است که در دنباله‌ها یک چنین فاصله‌هایی وجود ندارد و تمامی حروف الفبا پشت سر هم قرار می‌گیرند. بنابراین ساختار DNA به گونه‌ای نیست که بتوان آن‌را بصورت کارا به کلمه‌ها شکست و در نتیجه روش‌های اندیس‌گذاری رشته‌های عادی مانند اندیس وارونه[1] و B-Tree نمی‌تواند بصورت موثر در مورد دنباله‌های DNA بکار گرفته شود (با رس کی[2]، استج[3]، تومو[4] و آپتون[5]، ۲۰۰۸)

درخت پسوندی ساختار داده‌ای است که به کمک آن می‌توان تمامی پسوندهای یک رشته را نمایش داد، در اینصورت اعمالی نظیر یافتن یک الگوی خاص در دنباله‌های DNA به راحتی امکان‌پذیر می‌شود. جهت سهولت نوشتار در این تحقیق از تفاوت دنباله و رشته صرف‌نظر می‌شود، از رشته بجای دنباله استفاده می‌شود.

درخت پسوندی برای یک رشته، درخت ریشه‌داری است که برچسب ساق‌ها نمایانگر زیر

[1] Inverted Index
[2] Barsky, M
[3] Stege
[4] Thomo
[5] Upton

رشته‌ای از رشته اصلی می‌باشند. برای هر پسوند، یک مسیر منحصربه‌فرد در درخت وجود دارد که از ریشه آغاز و در برگ به پایان می‌رسد. شکل ۱-۲ نمونه‌ای از این درخت را برای رشته TATAT نشان می‌دهد.




**شکل ۱-۲.** درخت پسوندی برای رشته TATAT

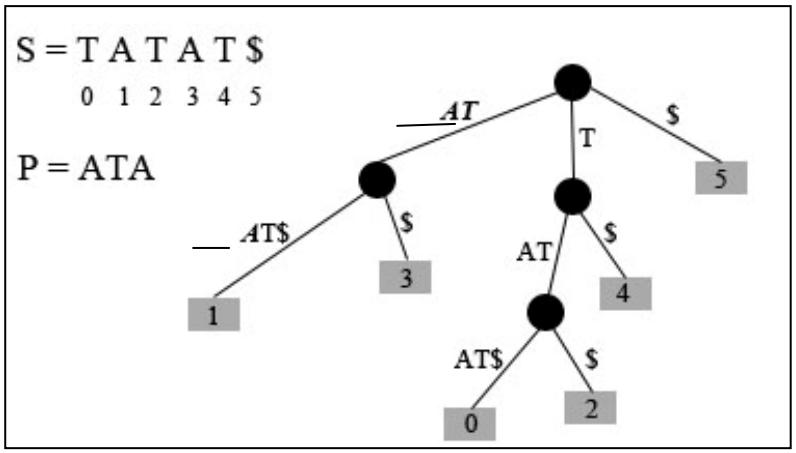
این درخت را می‌توان در زمان خطی نسبت به طول رشته ایجاد کرد. پس از ساخت درخت، درخواست‌هایی نظیر تطبیق دقیق رشته، یافتن تعداد تکرارها، یافتن طولانی‌ترین زیررشته و ... به راحتی قابل اجرا می‌باشد.

تطبیق دقیق رشته[1]، یافتن محل‌هایی است که الگوی P در رشته رخ داده است. اگر این در محل i از رشته S رخ دهد، در اینصورت P پیشوندی از پسوند iام S می‌باشد. در حقیقت P با بخشی از یک مسیر در درخت پسوندی، مطابقت دارد. به کمک درخت پسوندی، می‌توان این مسئله را به راحتی حل کرد. نمونه‌ای از این مسئله برای S = TATAT و P = ATA، در شکل ۱-۳ نشان داده شده است.

یافتن تعداد تکرارها، شامل تکرارهای دقیق[2]، تکرار با K عدم تطابق[3] و تکرار با K تفاوت[4]

[1] Exact String Matching
[2] Exact repeat
[3] K-mismatch repeat
[4] K-differences repeat

می‌باشد. برای دو زیر رشته $S[i_1..j_1]$ و $S[i_2..j_2]$، این مسائل بصورت زیر تعریف می‌شوند:




**شکل ۱-۳.** یافتن الگوی ATA در رشته TATAT

اگر $S[i_1..j_1] = S[i_2..j_2]$ باشد تکرار، دقیق است. اگر بین $S[i_2..j_2]$ و $S[i_1..j_1]$، دقیقاً k عدم تطابق وجود داشته باشد، تکرار با K عدم تطابق است. اگر k تفاوت (عدم تطابق، کمی و زیادی) بین $S[i_2..j_2]$ و $S[i_1..j_1]$ وجود داشته باشد، تکرار با K تفاوت است.

رشته S3 عبارت است از طولانی‌ترین زیررشته[1] $S_1$ (بقول n) و $S_2$ (به طول m)، که هم در $S_1$ و هم در $S_2$ رخ دهد. این مسئله به کمک ساخت درخت پسوندی تعمیم‌یافته[2]، در زمان O(n+m) قابل حل است. با داشتن یک چنین درختی، کافی است طولانی‌ترین مسیر مشترک که از هر دو رشته دارای برگ است را در نظر بگیریم. شکل ۱-۴ نمونه‌ای از این کاربرد را برای دو رشته TATAT و AATAA نشان می‌دهد.

کاربردهای یاد شده، قدرت درخت‌های پسوندی را نشان می‌دهد، آنچنان که در (بارسکی[3]، ۲۰۰۶) و (گاسفیلد[4]، ۱۹۹۷) بیان شده است : «برای درک قدرت درخت پسوندی کافی است سعی کنیم تا این مسایل را بدون استفاده از درخت حل کنیم».

[1] longest common substring
[2] generalized suffix tree
[3] Barsky
[4] Gusfield