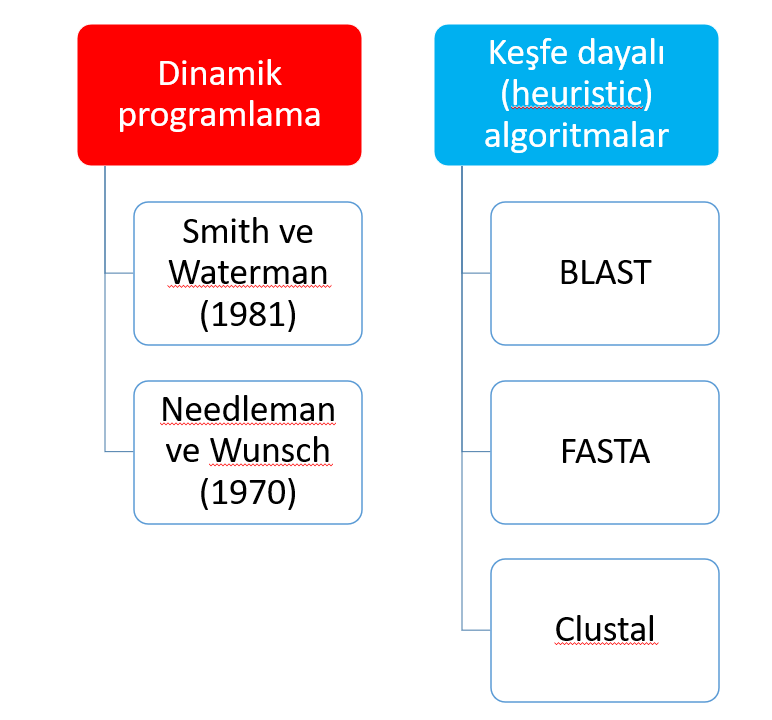
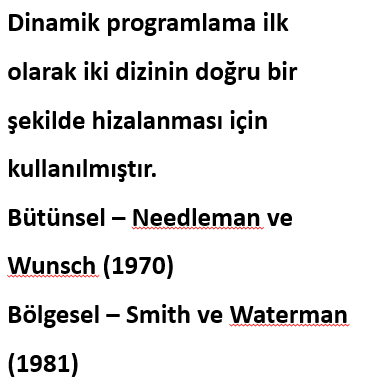
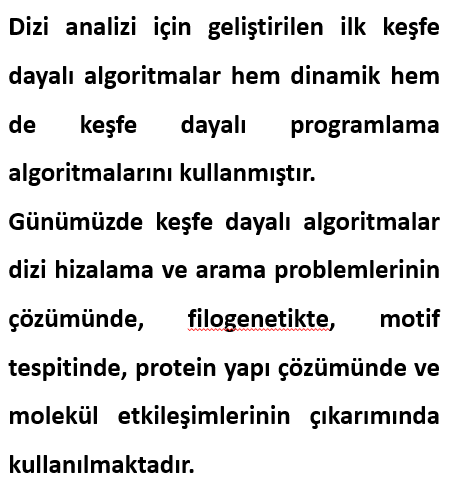
BİYOİNFORMATİK PROJE ÖDEVİ  FASTA ve BLAST Hizalama Algoritmaları

**HASAN KORKMAZ 180508011**

1985 yılında bilgisayarlı moleküler biyolojide filtreleme fikri FASTA algoritmasında David Lipman ve Bill Pearson tarafından kullanılmıştır. BLAST’da daha da geliştirilmiştir ve artık moleküler biyolojide yaygın bir veri tabanı arama motoru haline gelmiştir.





Sözcük yöntemleri

Sözcük yöntemleri, *k*-li yöntemler olarak da bilinir, optimal hizalama çözümünü bulması garantili olmayan, ama dinamik programlamadan önemli derecede daha verimli olan [höristik](https://tr.wikipedia.org/w/index.php?title=H%C3%B6ristik&action=edit&redlink=1) yöntemlerdir. Bu yöntemler büyük ölçekli veritabanlarında özellikle yararlıdır, çünkü bu veritabanlarındaki aday dizilerin büyük bir kısmının sorgu dizisiyle anlamlı bir uyuşma göstermeyeceği peşinen bilinmektedir.

Sözcük yöntemleri, veritabanı arama araçları olan BLAST ailesi üyelerindeki ve FASTA programındaki uygulamaları ile bilinirler. Sözcük yöntemleri sorgu dizisinde bulunan kısa, örtüşmeyen altdizileri ("sözcükleri") tespit edip bunları veritabanındaki aday dizilerle eşleştirir. Bir sözcüğün iki dizideki pozisyonları arasındaki fark hesaplanarak bir kayma değeri elde edilir. Eğer farklı sözcükler aynı kayma değerine sahipse bu kayma değeri bir hizalanma bölgesini belirler. Ancak eğer böyle bir bölge bulunursa, o zaman daha hassas hizalama kıstasları uygulanır. Böylece kayda değer benzerlik göstermeyen diziler arasında pek çok gereksiz karşılaştırmanın önüne geçilmiş olur.

FASTA yönteminde kullanıcı, veritabanını aramakta kullanılacak sözcüğün uzunluğuna karşılık gelen bir *k* değeri seçer. Küçük *k* değerlerinde yöntem daha yavaş ama daha duyarlıdır, bunlar kısa sorgu dizileri için tercih edilir.

BLAST ailesindeki araştırma yöntemleri, belli sorgu tipleri (örneğin uzaktan ilişkili dizilerin bulunması) için optimize edilmiş çeşitli algoritmalara sahiptir. BLAST, FASTA'dan daha hızlı olup doğruluktan çok az bir fedakarlık yapma özelliği için geliştirilmiştir. FASTA gibi BLAST da *k* uzunluklu bir sözcük araması yapar ama sadece en önemli sözcük uyuşmalarını değerlendirir, FASTA gibi her sözcük uyuşmasına bakmaz. Çoğu BLAST uygulamasında varsayılan sözcük uzunluğu değişmez, sorgu ve arama tipine göre optimize edilir, ancak istisnai durumlarda, tekrarlı dizileri ararken veya kısa sorgu dizileri için değiştirilir. BLAST çeşitli Web portallarında bulunabilir, örneğin [EMBL](http://www.ebi.ac.uk/fasta33/).

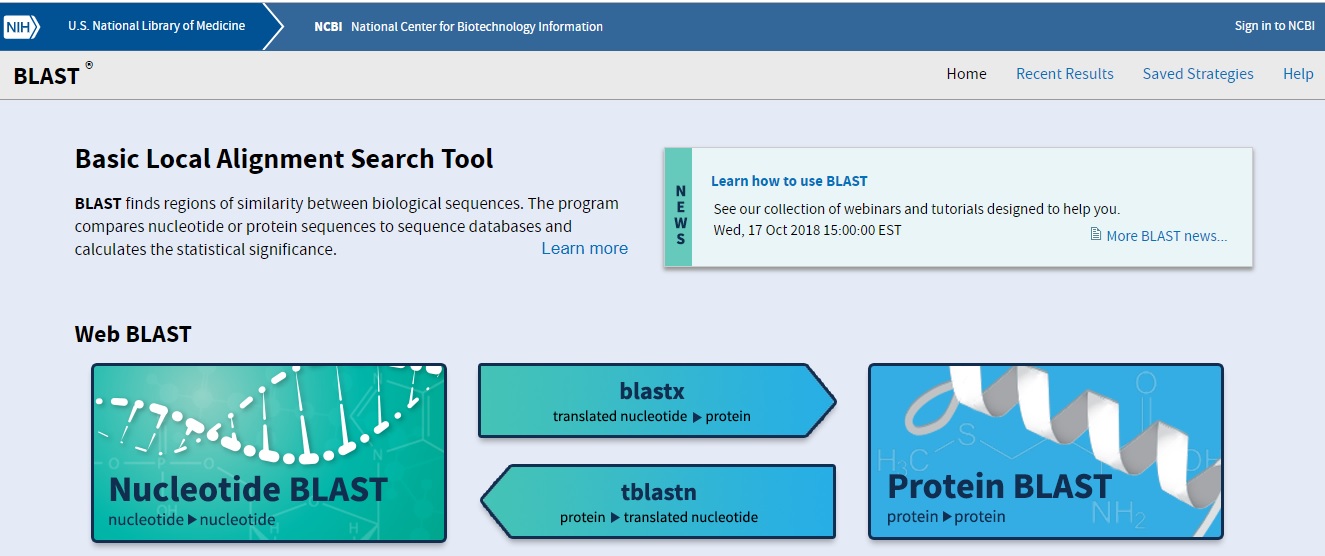
## **BLAST Nedir?**

BLAST, **Temel Yerel Hizalama Arama Aracı** anlamına gelir. Bu, bir sorgu dizisi ile Ulusal Biyoteknoloji Bilgi Merkezi (NCBI) web sitesinde saklanan diziler arasındaki benzerliği arar. Sorgu dizisindeki varsayılan genler, biriken dizilerin dizi homolojisine dayanarak tespit edilebilir. BLAST, iki dizi arasındaki yerel benzerlik bölgelerini hızlı bir şekilde belirleyebildiği için biyoinformatik bir araç olarak popülerdir. BLAST, iki dizi arasındaki eşleşme sayısını tahmin eden bir beklenti değeri hesaplar. Dizilerin yerel hizalamasını kullanır. NCBI BLAST web arayüzü burada bulunabilir.

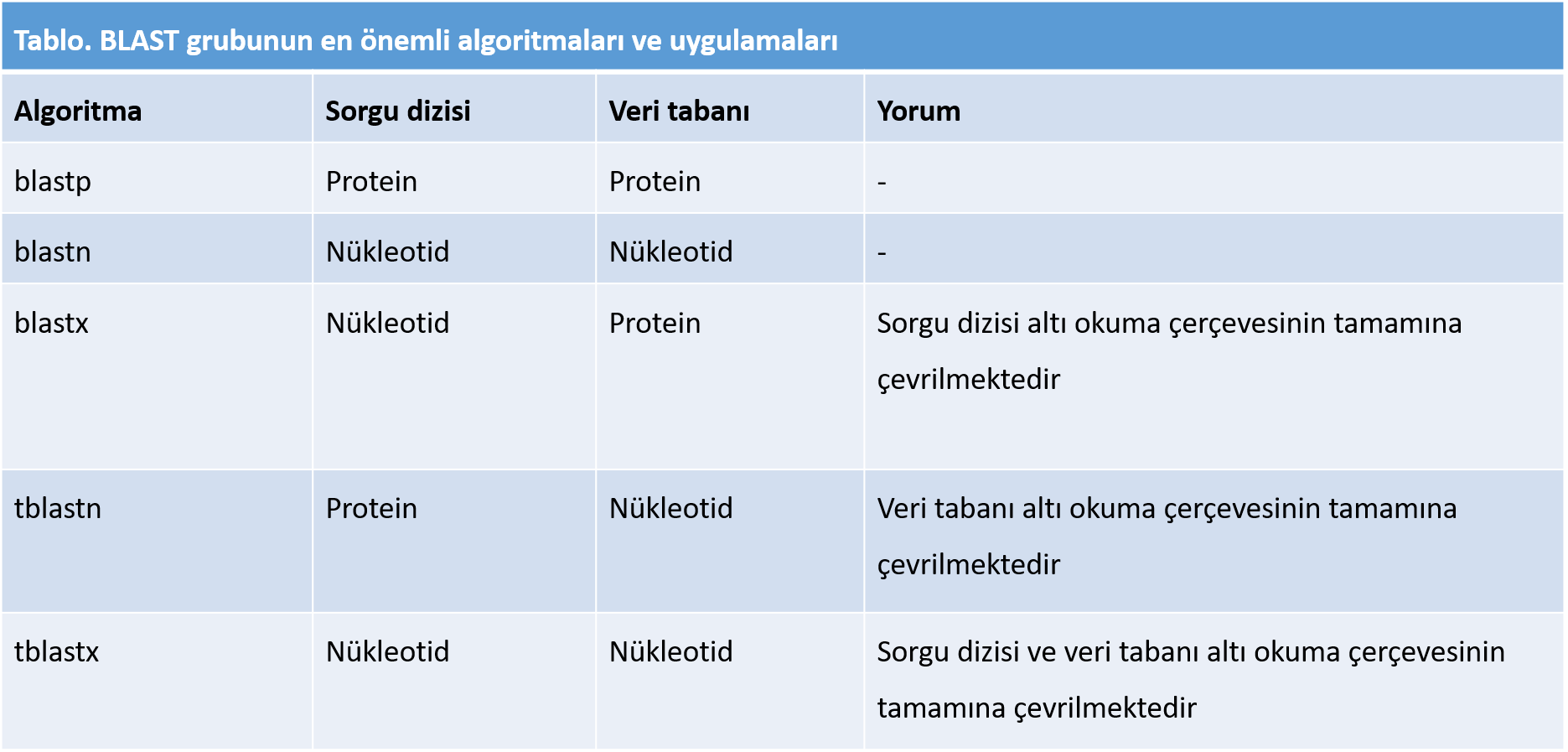
**BLAST (Basic Local Alignment Search Tool = Temel Bölgesel Hizalama Arama Aracı)**

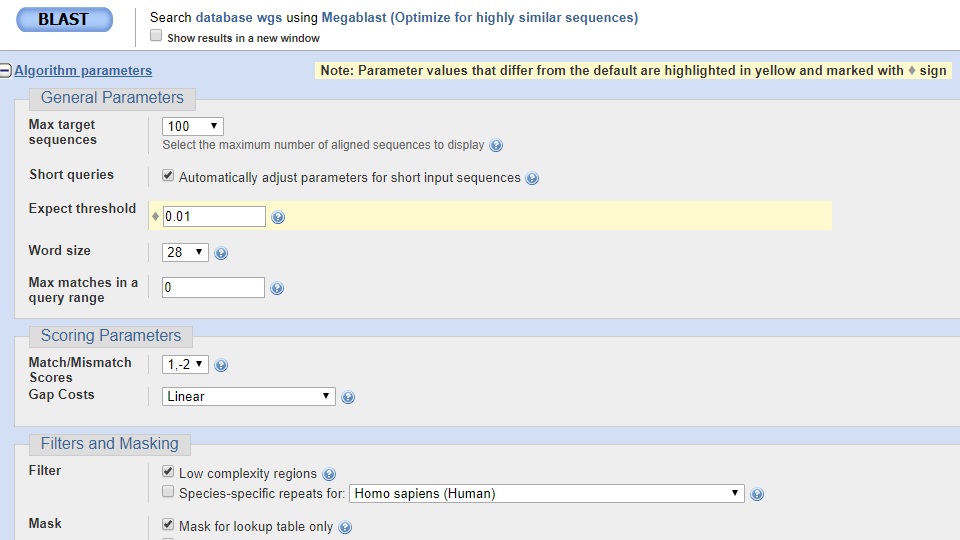
Standart bir Smith-Waterman algoritması kullanarak yerel dizi hizalama araması oldukça yavaş bir süreçtir. Yerel dizi hizalamasını hızlandıran BLAST algoritmasının üç temel adımı vardır. İlk olarak, sorgu dizisiyle hizalandığında bir eşik değerinin üzerinde puan alan tüm kısa dizilerin (WORDS adı verilen) bir listesini oluşturur. Ardından, bu sözcüklerin oluşumları için dizi veritabanı aranır. Kelime uzunluğu çok kısa olduğu için (proteinler için 3 kalıntı, nükleik asitler için 11 tortu), tüm kelimelerin önceden hesaplanmış bir tablosunu ve sıralama hızlarını iyileştirilmiş hız için aramak mümkündür. Bu eşleşen sözcükler daha sonra sorgu dizisi ve veritabanı dizisi arasındaki uyumsuz yerel hizalamalara genişletilir. Uzantılar, hizalama puanı bir eşiğin altına düşene kadar devam eder. Bir sıradaki veya en yüksek skorlu segment çiftlerinde (MSP’ler) en yüksek skorlama hizalamaları, mümkün olan yerlerde lokal hizalamalarla birleştirilir. BLAST yazılım paketine yapılan yeni eklemeler de boşluklu hizalamaları arar.

* BLAST; NCBI [ncbi-blast], EMBL-EBI [embl-blast] ve DDBJ [ddbj-blast] veri tabanlarında Web servisi olarak hizmete sunulmaktadır.
* BLAST ilk olarak genellikle farklı veri tabanlarındaki girdilerin toplandığı tekrarsız bir veri tabanına karşı uygulanmaktadır.
* Tekrarsız bir veri tabanında çoklu girdiler ayıklandığı için her girdinin sadece bir kaydı bulunmaktadır.
* Bu veri tabanları, hem nükleotidler hem de proteinler için mevcuttur.



BLAST algoritmasının iki uygulaması vardır: NCBI BLAST ve WU-BLAST. Her ikisi de web servisleri ve indirilebilir yazılım paketleri olarak kullanılabilir. NCBI BLAST, Ulusal Biyoteknoloji Bilgi Merkezi’nden (NCBI) alınırken, WU-BLAST Washington Üniversitesi’nde geliştirilmekte ve muhafaza edilmektedir. NCBI BLAST, daha sık kullanılan iki tanesidir. Bu programın en yeni sürümleri çoklu dizi profillerini karşılaştırmak için yöntemler geliştirmeye odaklanmıştır. Öte yandan WU-BLAST, boşlukların ele alınması için farklı bir sistem ve genom dizilerini aramak için yararlı olan bir dizi özellik geliştirmiştir.





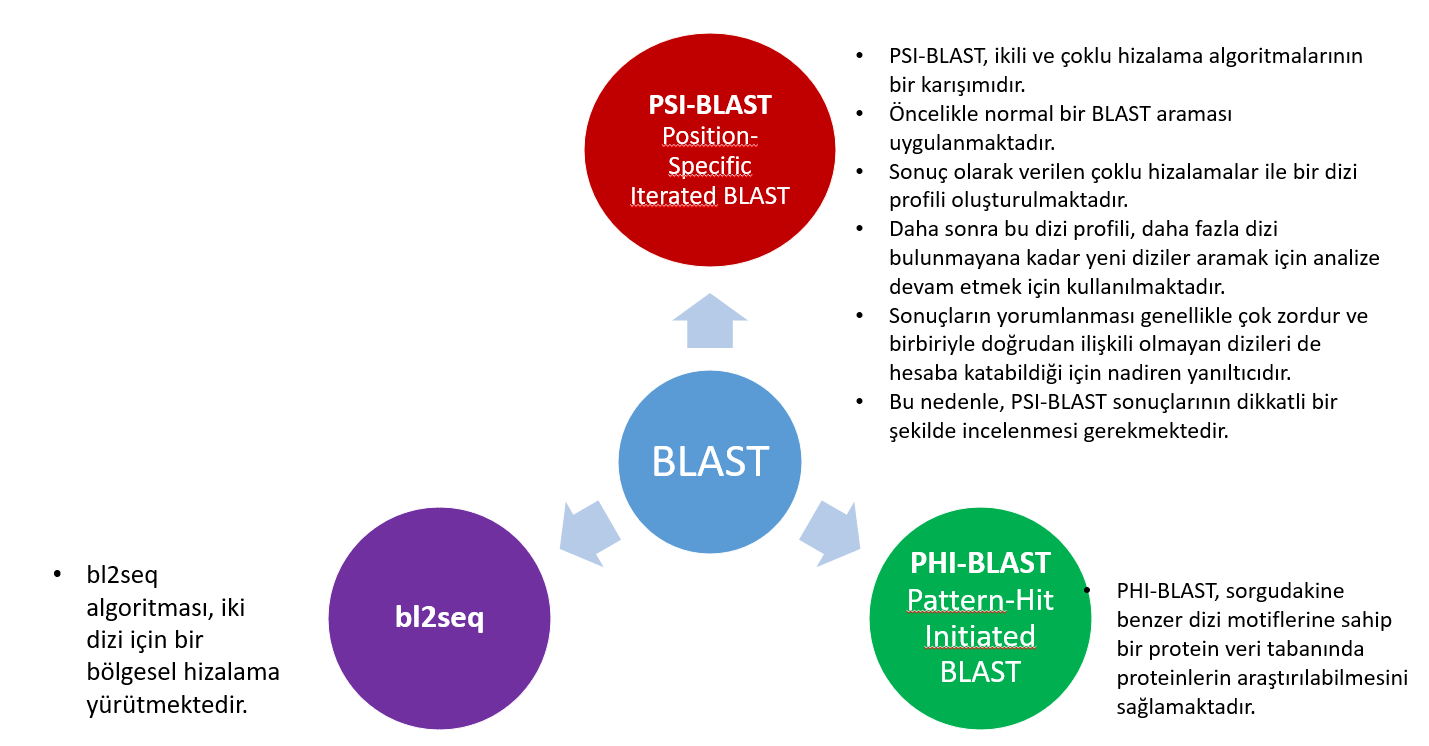
Çok düşük dizi benzerliğinin olduğu durumlarda, ortak bir atasal diziye dayanmayan, görünüşte anlamlı ancak rastgele bir hizalama bulunabilmektedir.

Bu nedenle bir E-değeri her BLAST sonucunun bir parçasıdır ve bir hizalamanın önemliliğini değerlendirmek için kullanılabilmektedir.

E-değeri 0.02’den küçük olan ikili hizalamalar, homolog dizilere dayanan anlamlı hizalamalar olarak kabul edilmektedir.

Rastgele bir hizalama E-değeri 1’den büyük olan sonuçları göstermektedir.

İkisi arasındaki değerler, homoloji hakkında bir fikre varabilmek için daha fazla bilgiye (örn. benzer işlev) gereksinim duymaktadır.



**BLAST sonuçlarını değerlendirme**

BLAST araması, sonuçlarını yorumlamanıza izin veren üç ilgili bilgi parçasını sağlar: ham puanlar, bit puanları ve E değerleri.

Bir yerel dizi hizalaması için ham puan, hizalamayı oluşturan maksimum puanlama segmenti çiftlerinin (MSP) skorlarının toplamıdır. Bit puanları, log tabanı 2’ye hizalamayı oluşturan puanlama matrisinin log tabanından dönüştürülen ham puanlardır. E-değerleri, verilen bir dizi hizalamasının anlamlı olma olasılığı hakkında bilgi sağlar. Bir hizalamanın E değeri, rastgele bir veri tabanına karşı yapılan bir aramada gözlemlenen hizalamanın puanından büyük veya ona eşit olan bir skor ile bulunması beklenen hizalamaların sayısını gösterir. Böylece, büyük bir E değeri (5 veya 10), hizalamanın muhtemelen tesadüfen oluştuğunu ve hedef dizinin, veritabanındaki ilişkisiz bir diziyle hizalandığını gösterir. 0.1 veya 0.05 e-değerleri tipik olarak sıralı veri tabanı araştırmalarında kesinti olarak kullanılır. Bir veritabanı aramasında daha büyük bir E-değeri kesme kullanılması, daha uzak eşleşmelerin bulunmasına izin verir, ancak aynı zamanda daha yüksek oranda sahte hizalamalarla sonuçlanır. Üçten E değerleri, literatürde en sık bildirilen değerlerdir.

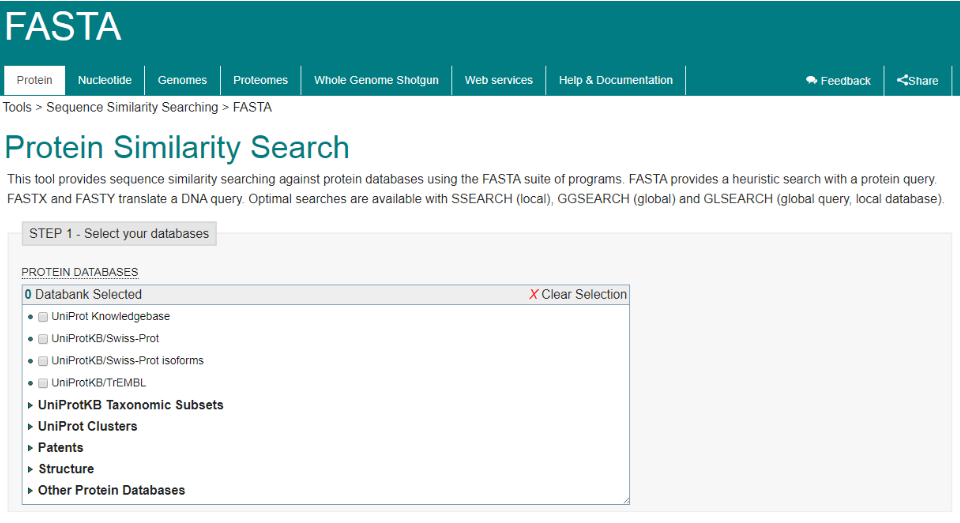
Dizilimin benzerliklerinin, karşılaştırılan dizilerin ilgili olup olmadığı konusunda bilinçsiz hale gelmesinin ötesinde bir sınır vardır. Bu sınır protein dizileri için yaklaşık% 25’lik dizi benzerliğinin altında görülür. Hala ilgili olduğuna inanılan düşük dizi benzerliği olan protein dizileri durumunda, yapısal analiz teknikleri böyle bir ilişki için kanıt sağlayabilir. Yapının bilinmediği yerlerde, düşük benzerliğe sahip sekanslar, birbiriyle ilgisiz olarak kategorize edilir, ancak bu, diziler arasındaki evrimsel mesafenin o kadar büyük olduğu ve bir ilişkinin tespit edilemeyeceği anlamına gelebilir.

## **FASTA Nedir?**

FASTA, DNA ve proteinlerin dizileri arasındaki benzerlikleri aramak için kullanılan başka bir dizi hizalama aracıdır. Sorgu sekansı, sekans paternlerine veya k-tuples olarak bilinen kelimelere ayrılır ve hedef sekanslar, ikisi arasındaki benzerlikleri bulmak için bu k-tupler için aranır. FASTA benzerlik aramaları için iyi bir araçtır. Sıralama benzerlikleri bulurken, aramanızı yürütmenin en iyi yolu önce bir BLAST araması yapmak ve ardından FASTA'ya gitmek. FASTA dosya formatı, BLAST gibi diğer dizi hizalama araçlarında giriş yöntemi olarak yaygın şekilde kullanılır. Avrupa Biyoinformatik Enstitüsü'nde (EBI) bulunan FASTA'nın web arayüzü burada bulunabilir.

### FASTA Kullanarak Yerel Hizalama

Lokal dizi hizalaması için başka bir yöntem FASTA algoritmasıdır. FASTA, BLAST’tan önce ve BLAST gibi, hem Web üzerinde bir hizmet olarak hem de indirilebilir bir program seti olarak kullanılabilir.



### FASTA algoritması

FASTA ilk olarak, hem sorgu dizisinde hem de dizi veritabanında oluşan kısa dizileri (ktup olarak adlandırılır) arar. Daha sonra, BLOSUM50 matrisini kullanarak, algoritma, en özdeş ktup’ları içeren 10 hatalı hizalamayı puanlar. Bu uyumsuz hizalamalar, bir eşiğin altındaki skoru azaltmadan, boşluklu bir hizalamada birleştirilebilmeleri için test edilir. Eşiği aşan birleştirilen hizalamalar için, o bölgenin en uygun yerel hizalaması hesaplanır ve bu hizalamanın (optimize edilmiş skor olarak adlandırılır) skoru bildirilir.

FASTA ktup’leri, BLAST kelimelerinden daha kısa, tipik olarak proteinler için 1 veya 2 ve nükleik asitler için 4 veya 6’dır. Düşük ktup değerleri daha yavaş fakat daha hassas aramalara neden olurken, daha yüksek ktup değerleri daha az yanlış pozitifle daha hızlı arama sağlar.

## **BLAST ve FASTA Arasındaki Fark**

## **Tanım olarak farkı :**

* **BLAST:** BLAST, nükleotit veya amino asit sekansları gibi birincil biyolojik sekans bilgilerinin karşılaştırılması için bir algoritmadır.
* **FASTA:** FASTA, bir DNA ve protein sekansı hizalama yazılım paketidir.

## **Global / Yerel Hizalama**

* **BLAST:** BLAST, yerel dizi hizalamasını kullanır.
* **FASTA:** FASTA ilk önce yerel dizi dizilimini kullanır ve daha sonra benzerlik arayışını global dizilime genişletir.

## **Yerel Sıra Hizalaması**

* **BLAST:** BLAST, iki sekanstaki bireysel kalıntıları karşılaştırarak yerel hizadaki benzerlikleri arar.
* **FASTA:** FASTA, dizi düzenlerini veya sözcükleri karşılaştırarak yerel hizalardaki benzerlikleri arar.

## **Aramanın Türü**

* **BLAST:** BLAST, yakından eşleşen veya yerel olarak en uygun dizilerde benzerlik arayışı için daha iyidir.
* **FASTA:** FASTA, daha az benzer sekanslarda benzerlik aramak için daha iyidir.

## **Bir tür iş**

* **BLAST:** BLAST, protein aramaları için en iyi sonucu verir.
* **FASTA:** FASTA, nükleotid aramaları için en iyi sonucu verir.

## **Sorgu Sırasındaki Boşluklar**

* **BLAST:** BLAST'ta, sorgu ile hedef diziler arasındaki boşluklara izin verilmez.
* **FASTA:** FASTA'da boşluklara izin verilir.

## **Duyarlılık**

* **BLAST:** BLAST hassas bir biyoenformatik araçtır.
* **FASTA:** FASTA, BLAST'tan daha hassastır.

## **Hız**

* **BLAST:** BLAST, FASTA'dan daha hızlı.
* **FASTA:** FASTA, BLAST ile karşılaştırıldığında daha az hızlıdır.

## **Geliştiriciler**

* **BLAST:** BLAST, 1990 yılında Ulusal Sağlık Enstitüsünde Stephen Altschul, Webb Miller, Warren Gish, Eugene Myers ve David J. Lipman tarafından tasarlanmıştır.
* **FASTA:** FASTA, 1985 yılında David J. Lipman ve William R. Pearson tarafından geliştirilmiştir.

## **Önem**

* **BLAST:** Şu anda, BLAST, benzerlik aramaları için en yaygın kullanılan biyoinformatik araçlarıdır.
* **FASTA: FASTA'nın** mirası, şimdi biyoinformatikte her yerde bulunan FASTA formatıdır.

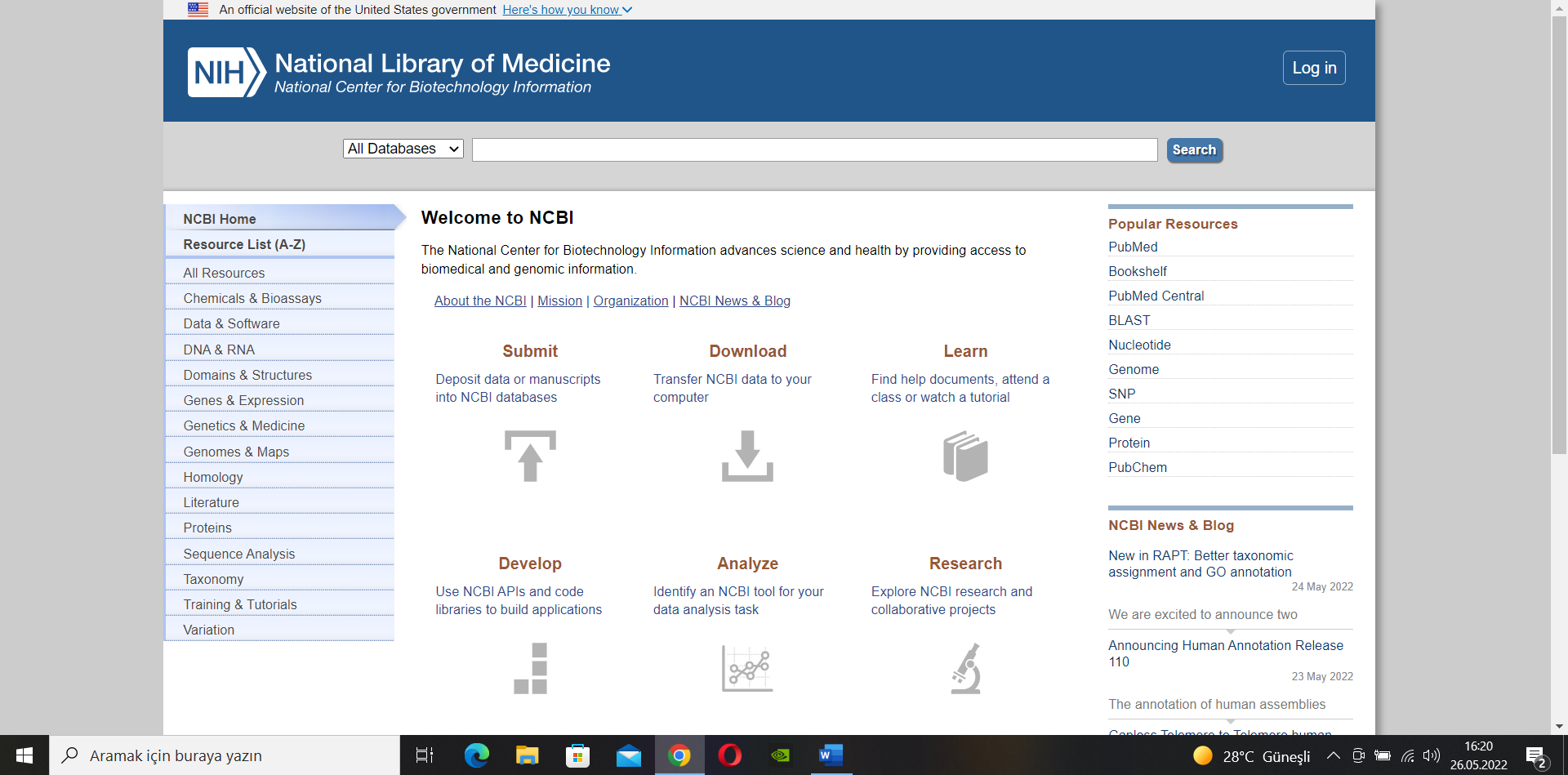
## **Sonuç**

BLAST ve FASTA, biyoinformatikte DNA veya protein dizileri arasındaki benzerlikleri aramak için kullanılan iki çift dizili dizilim aracıdır. BLAST, nükleotid ve amino asit dizilerinin lokal hizalanması için en yaygın kullanılan araçtır. FASTA, dizi desenleri veya kelimeler kullanan ince bir benzerlik arama aracıdır. Daha az benzer diziler arasındaki benzerlik aramaları için en uygun yöntemdir. BLAST ve FASTA arasındaki temel fark, her araçta kullanılan benzerlik arama stratejileridir.

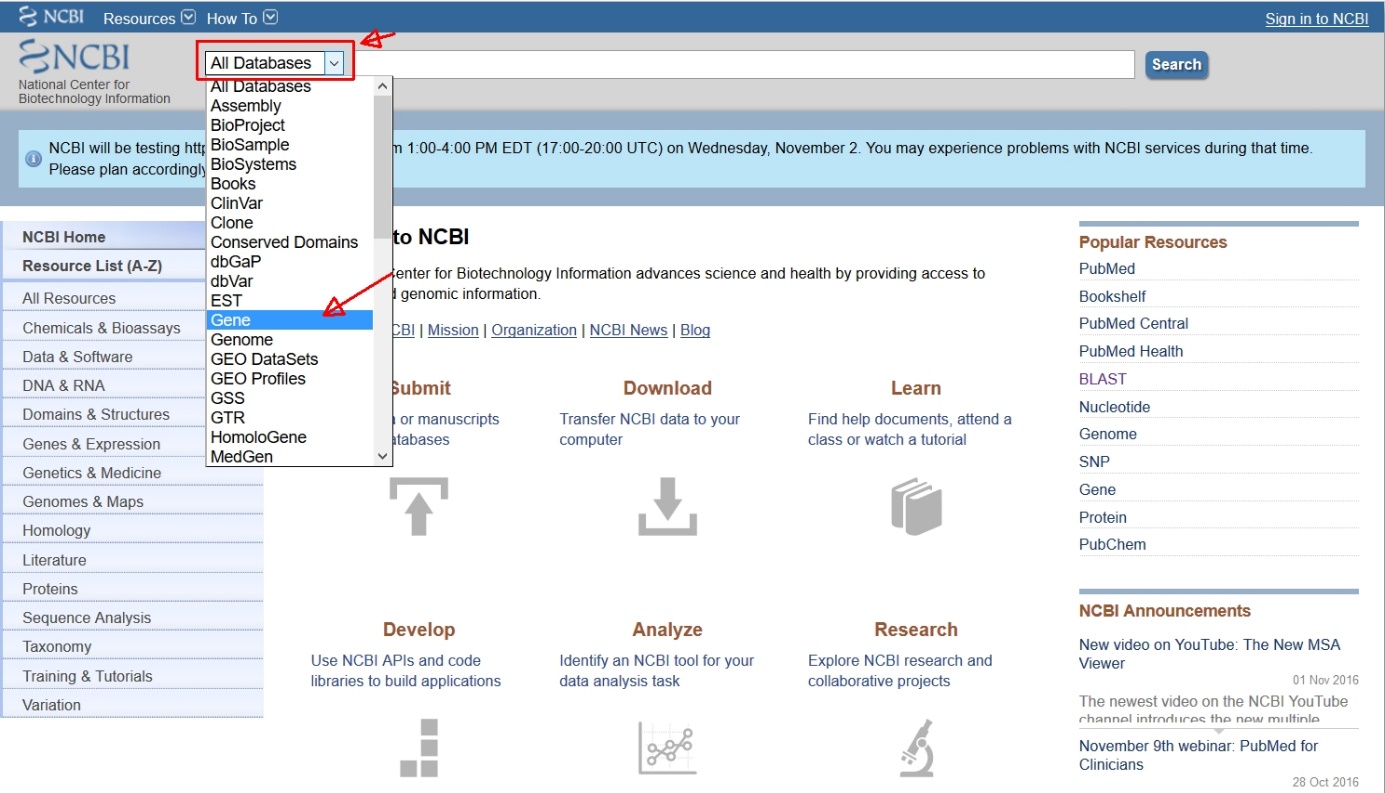
**Örnek uygulama ve açıklamaları:**

**Aşağıda mevcuttur.**

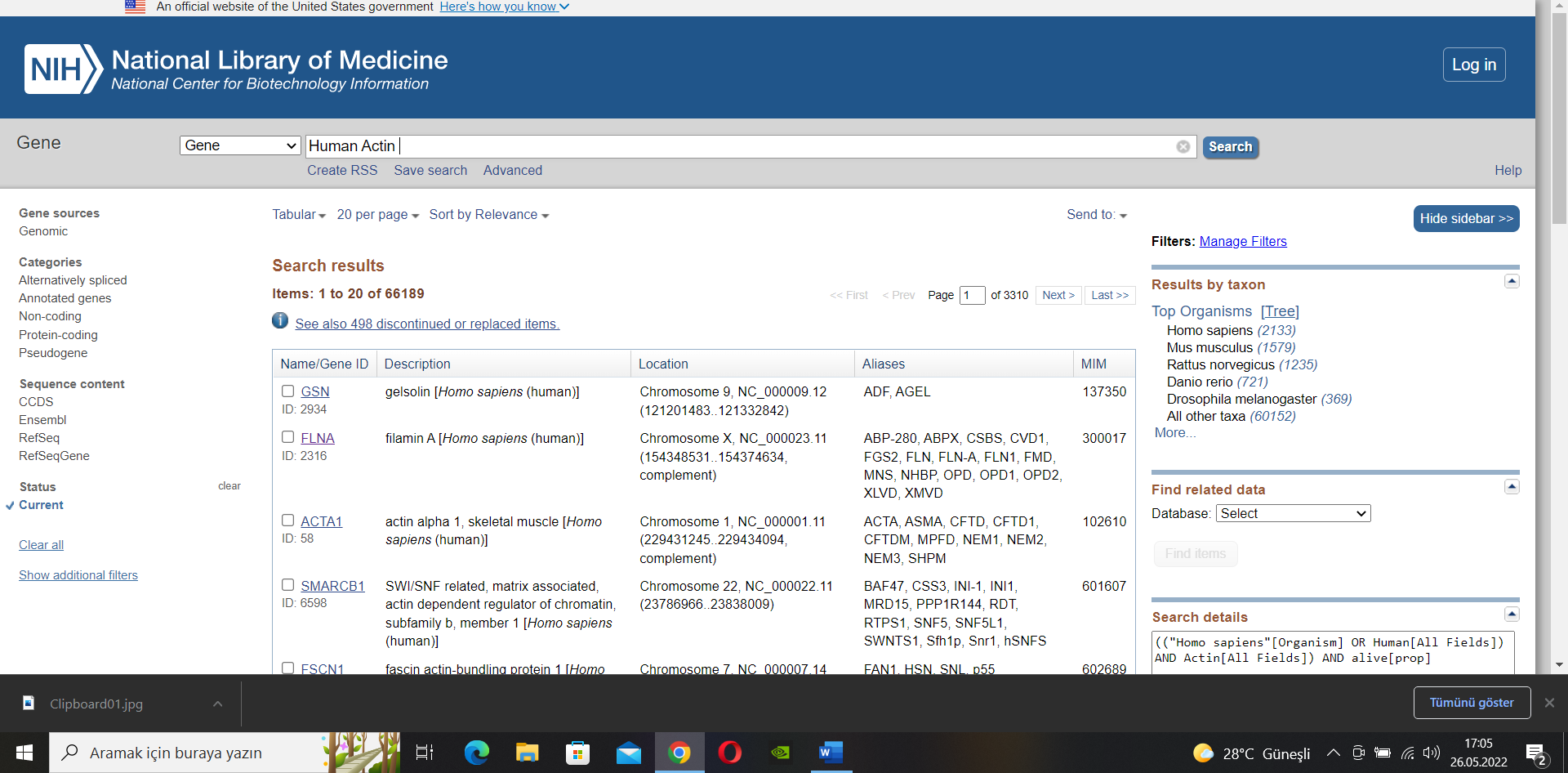
**Öncelikle hizalamak istediğimiz geni bulmak için** [**https://www.ncbi.nlm.nih.gov/**](https://www.ncbi.nlm.nih.gov/) **sitesine giriş yapıyoruz.**



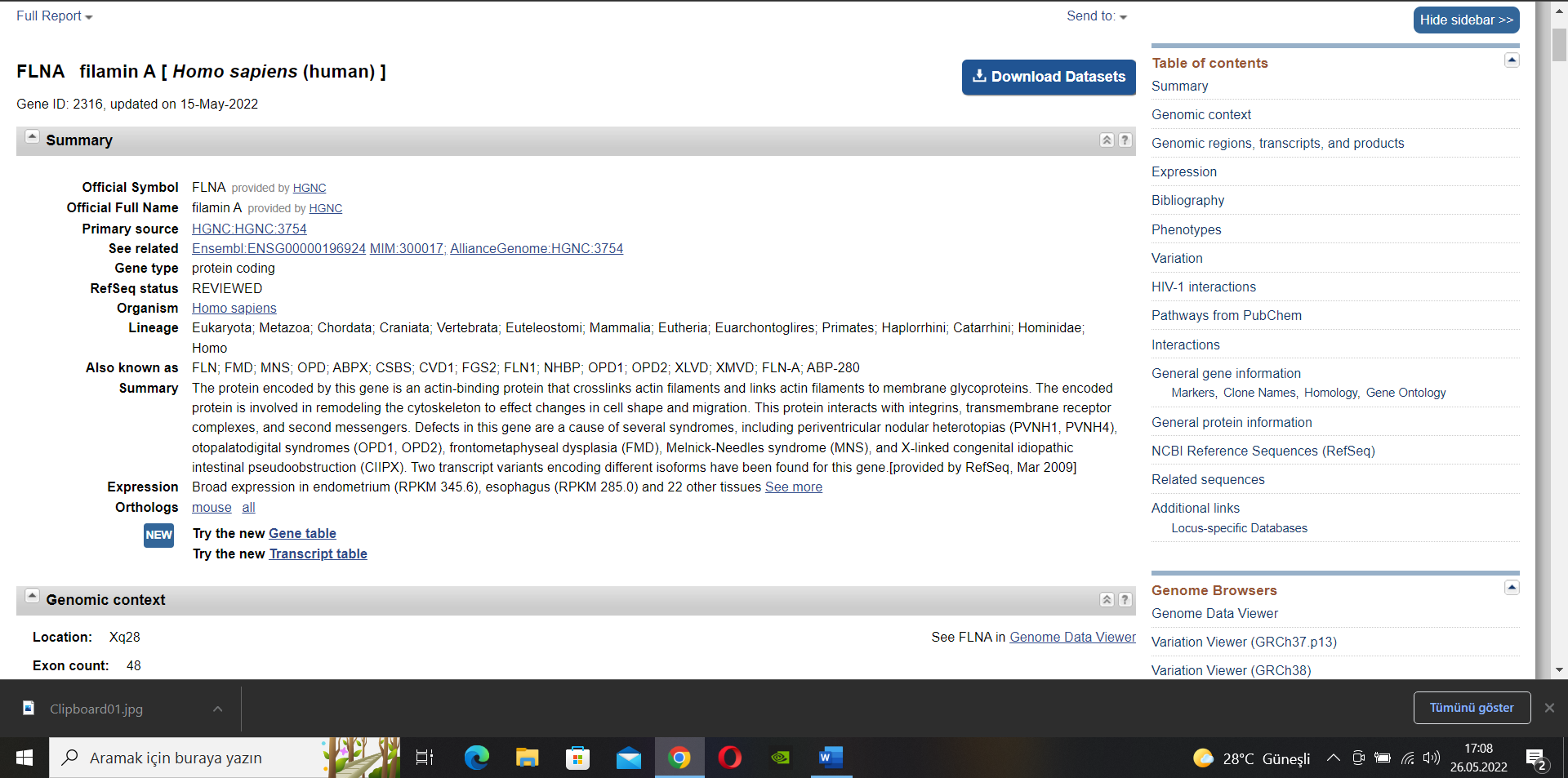
**Ardından arama yapmak için menüden gene seçiyoruz.**



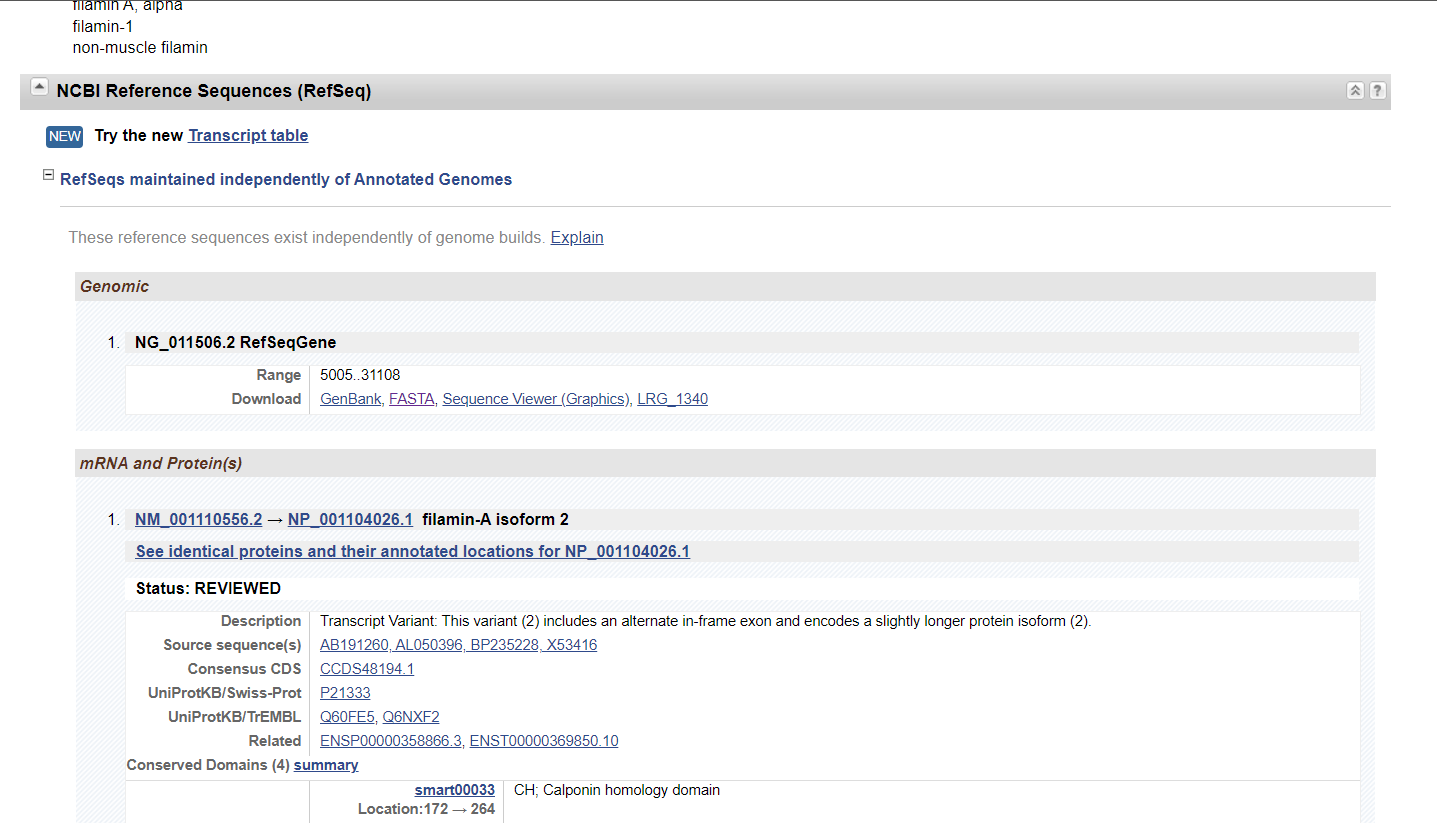
Şimdi aradığınız Genomun ismini belirtiniz, ben Human Actin(insan aktin) yazarak arama yaptım.



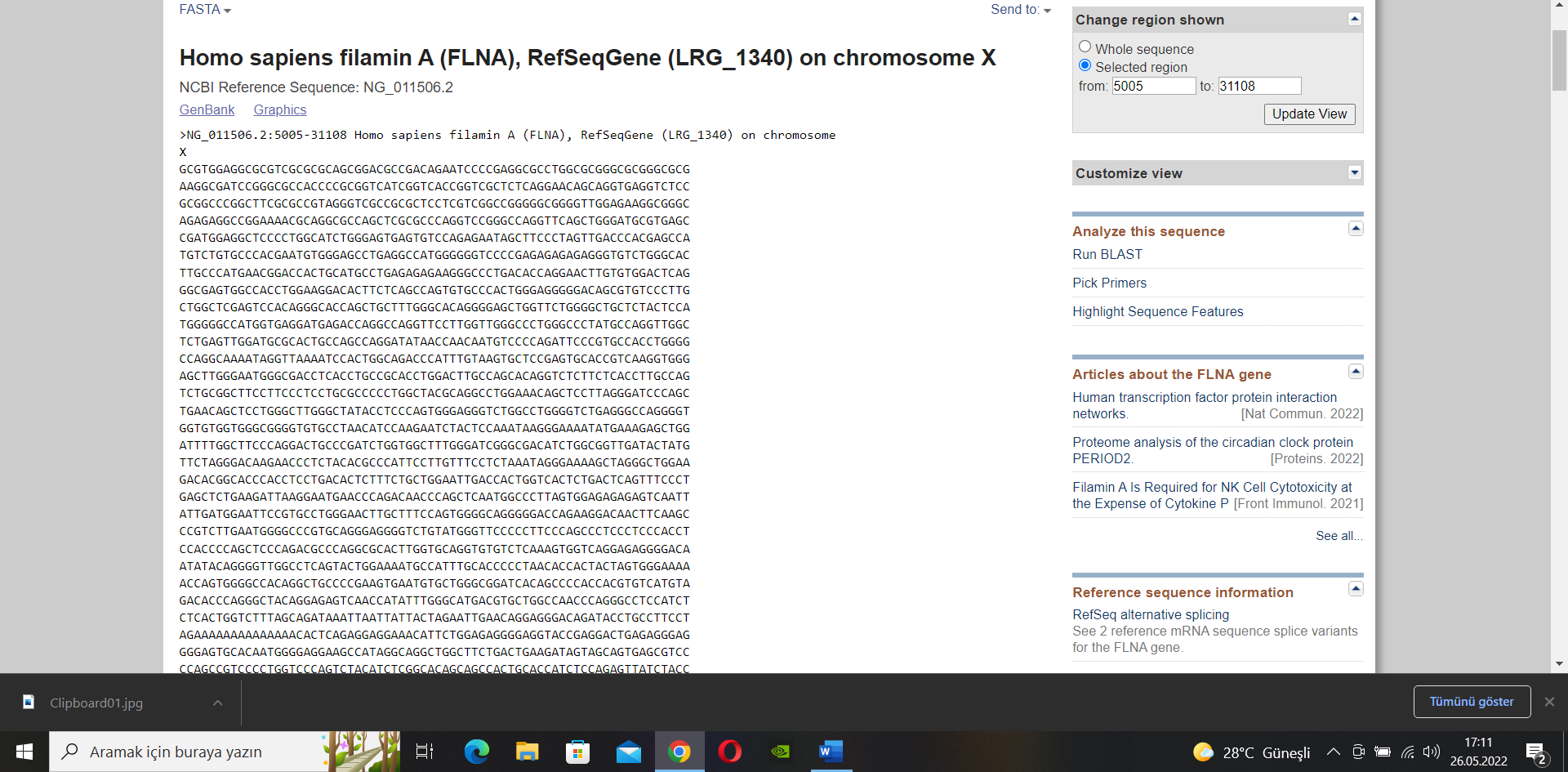
Genlerimiz çıktıktan sonra burda örnek olarak 2. Olan FLNA seçiyorum.FLNA’a tıkladığımız zaman Genomla ilgili temel bilgileri buradan görebiliyoruz.



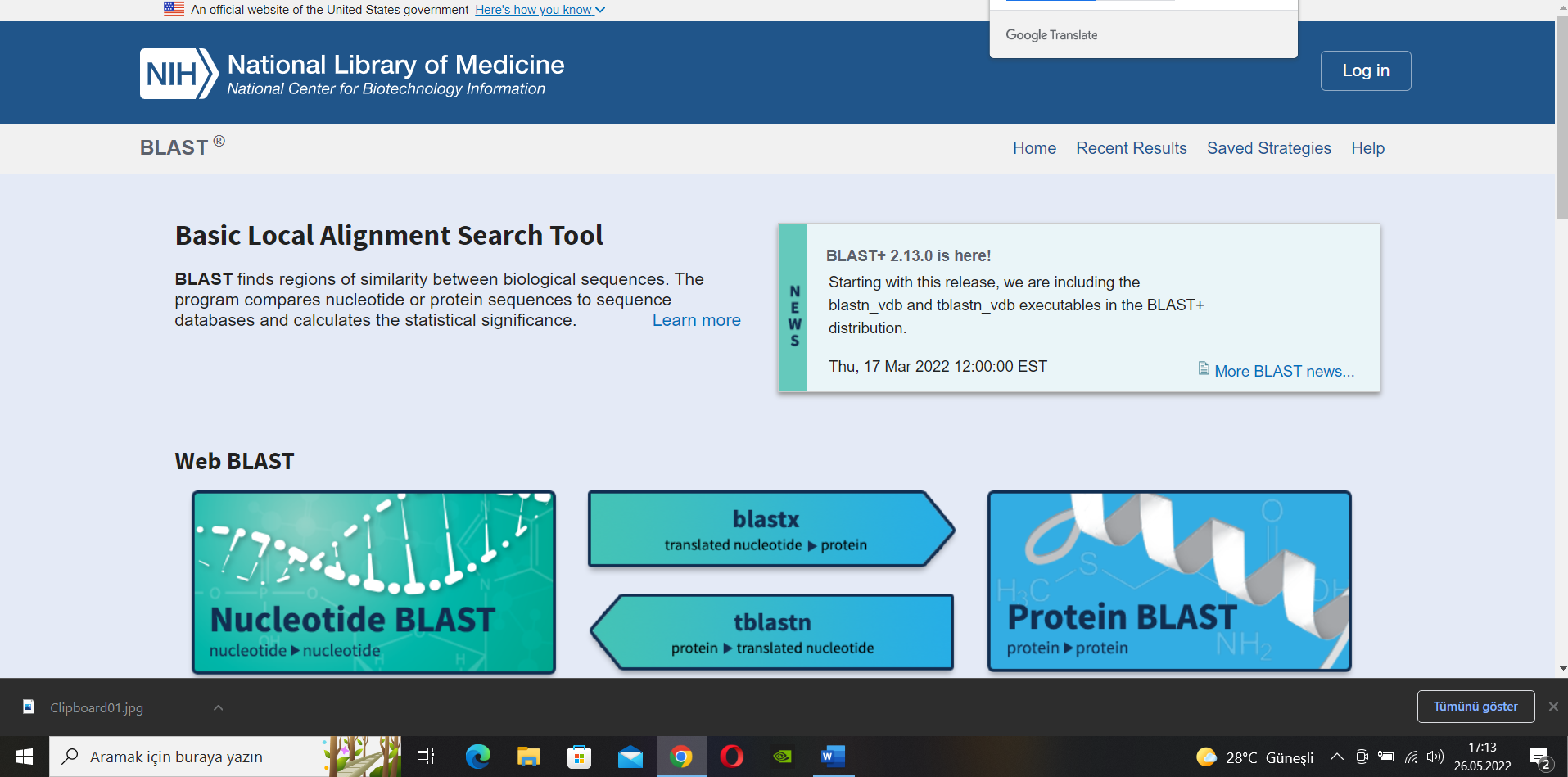
DNA dizisini bulmak için inceliyoruz.Bizim aradığımız DNA dizisi sayfanın oldukça aşağısında "NCBI Reference Sequences (RefSeq)" adlı başlığın altında bulabiliriz. "FASTA" formatında almak için o seçeneği seçiyoruz.



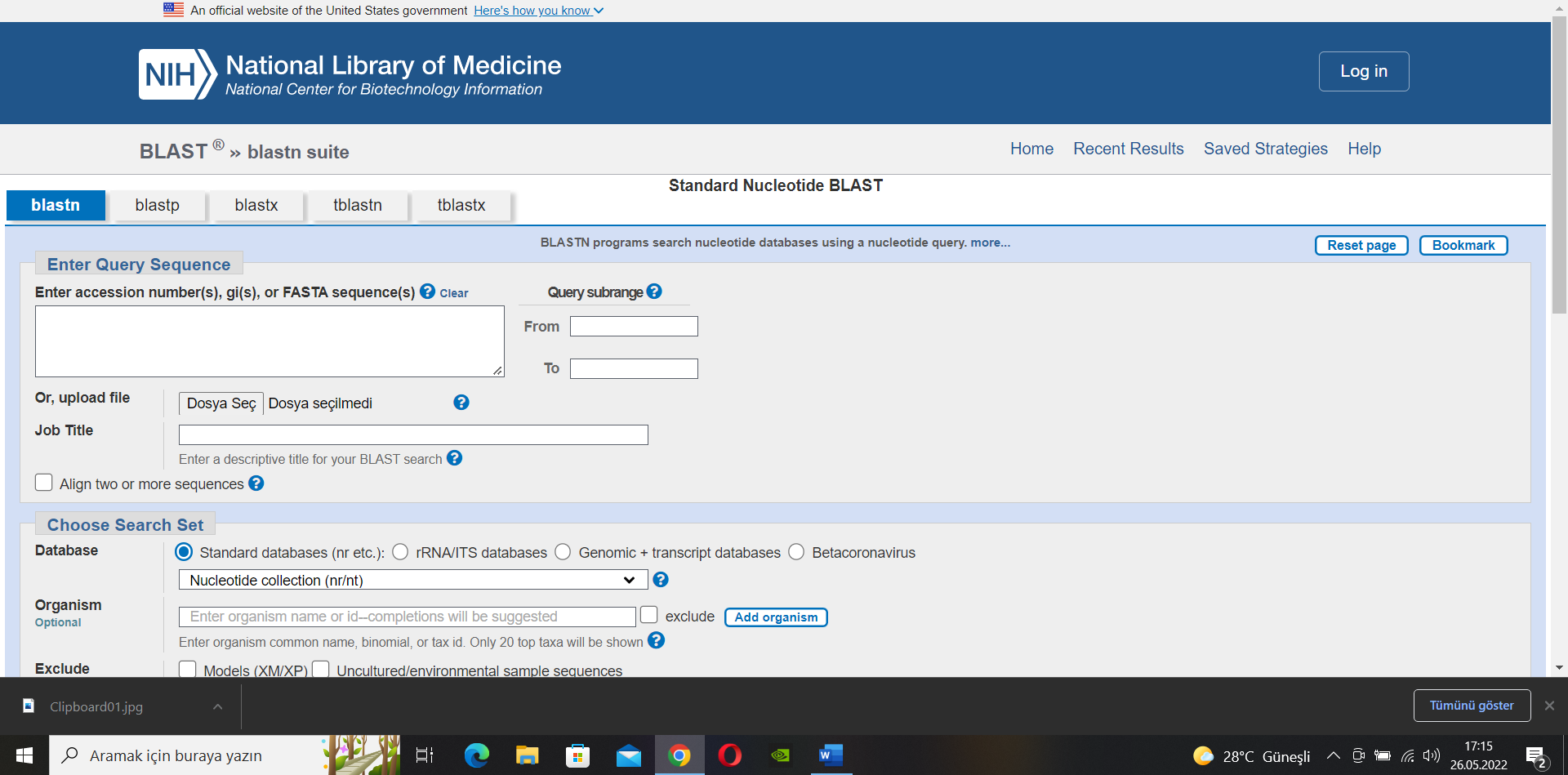
İlgili gene ait faşta formatı :



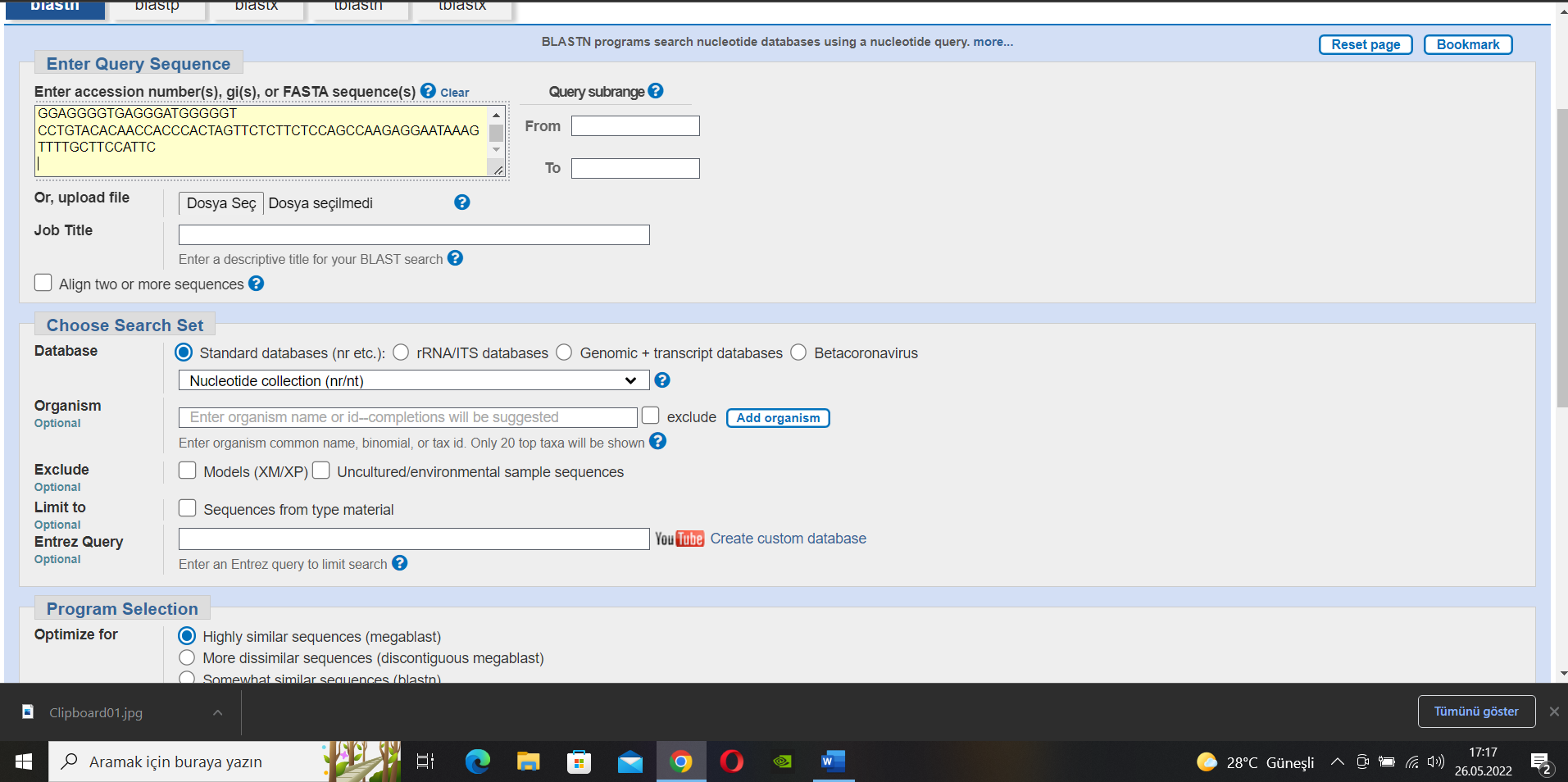
Fasta formatını elde ettikten sonra Açıklama satırı hariç, tüm DNA kodlarını seçerek kopyalayalım. Bu aşamadan sonra Blast programına geçip oraya yapıştıracağız.Ardından blasta giriş yapıyoruz.



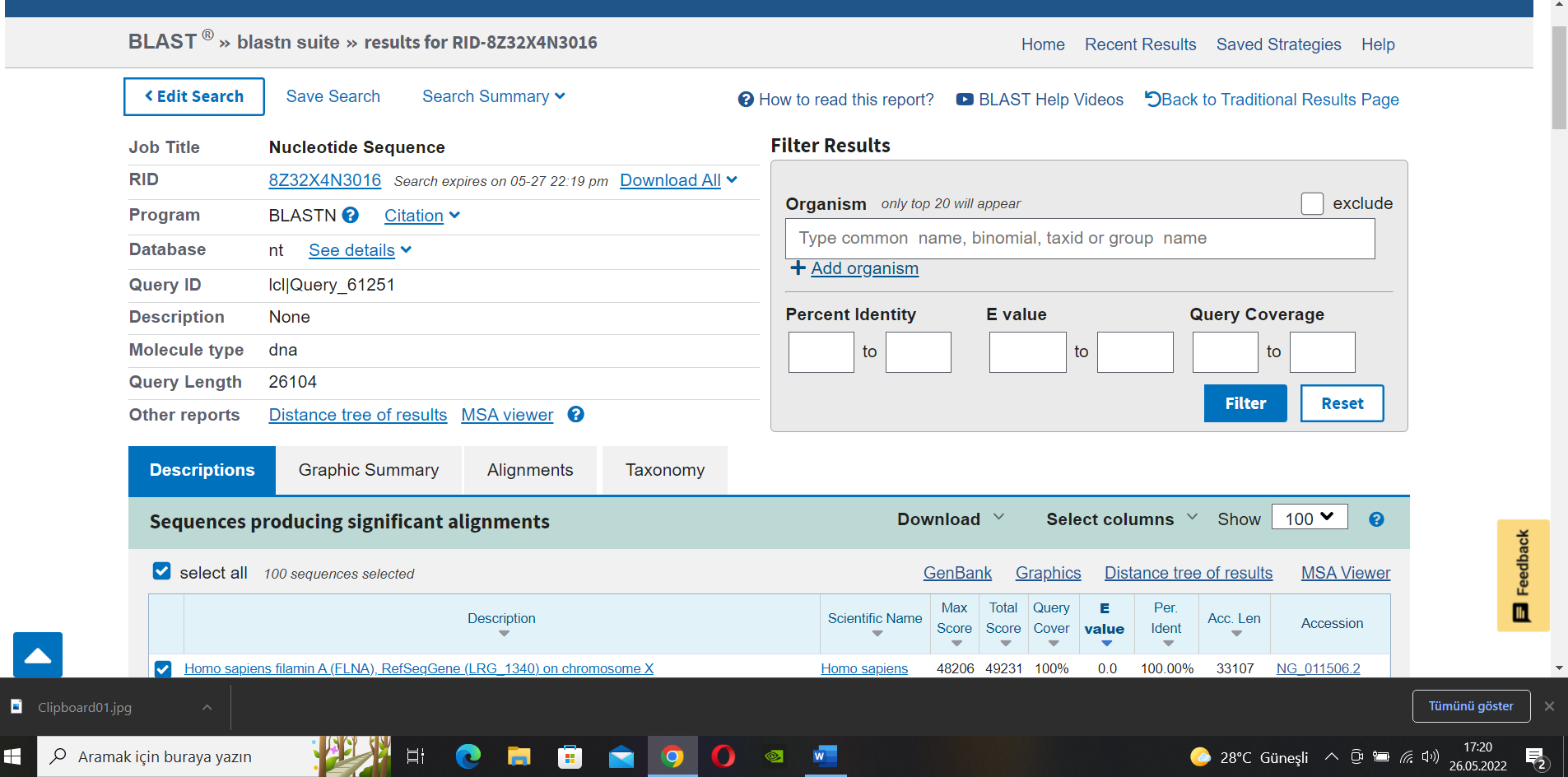
Bu ekranda nükleotit seçeneğini seçiyoruz.

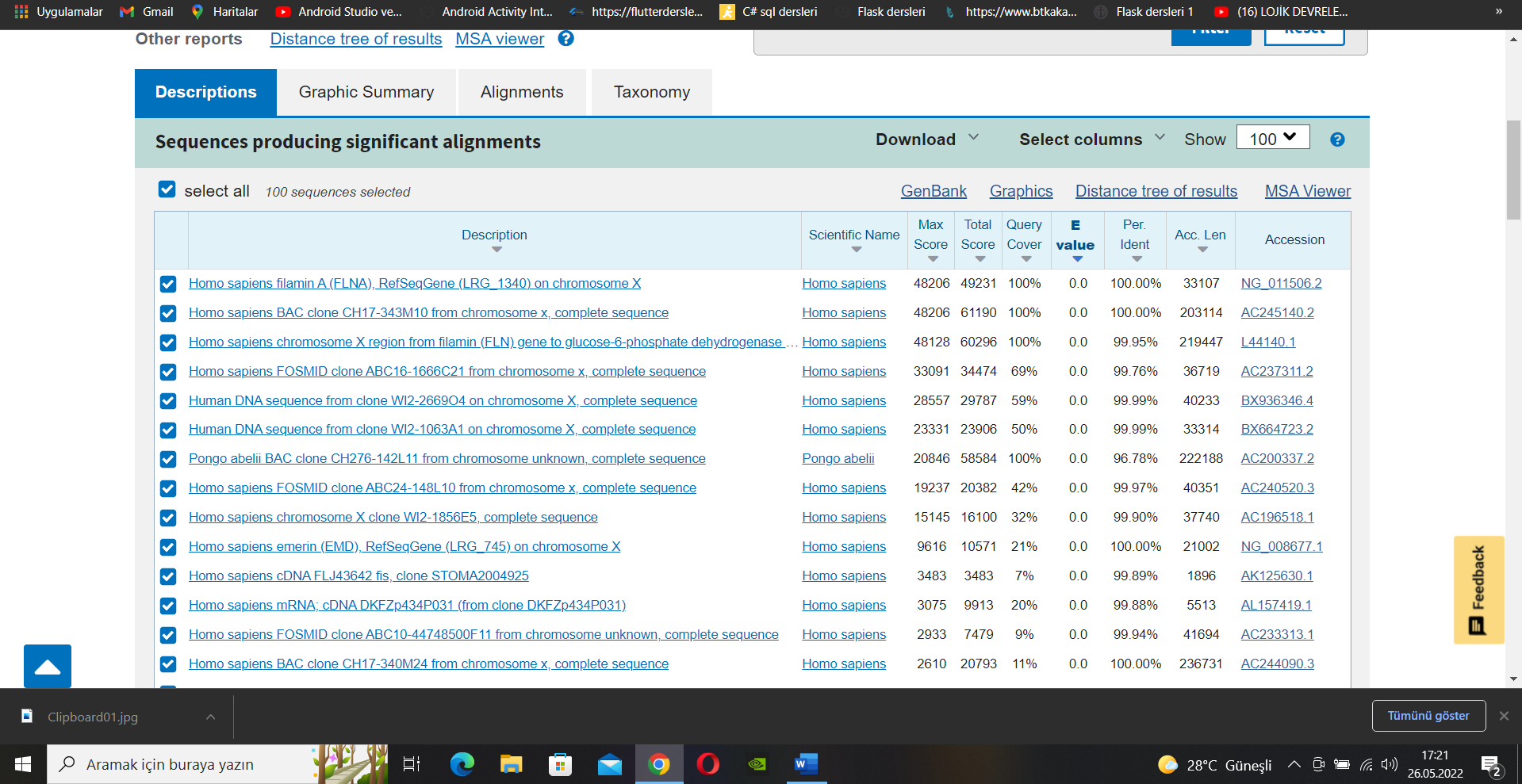


Seçimi yaptıktan sonra böyle bir ekran bizi karşılıyor. Açılan pencerede ana text alanına az önce kopyaladığımız DNA kodlarını yapıştırıyoruz. Veri tabanı olarak standart veri tabanını kullanalım. En aşağıda ise yüksek uyumluluktan - düşük uyumluluğa doğru seçim yapabilirsiniz. Düşük uyumluluk seçtiğinizde binlerce benzer dizi bulunabilir. Biz orada "Highly Similar" seçeneğini seçelim. Bu işlemler bittikten sonra En alttaki BLAST butonuna basıyoruz.

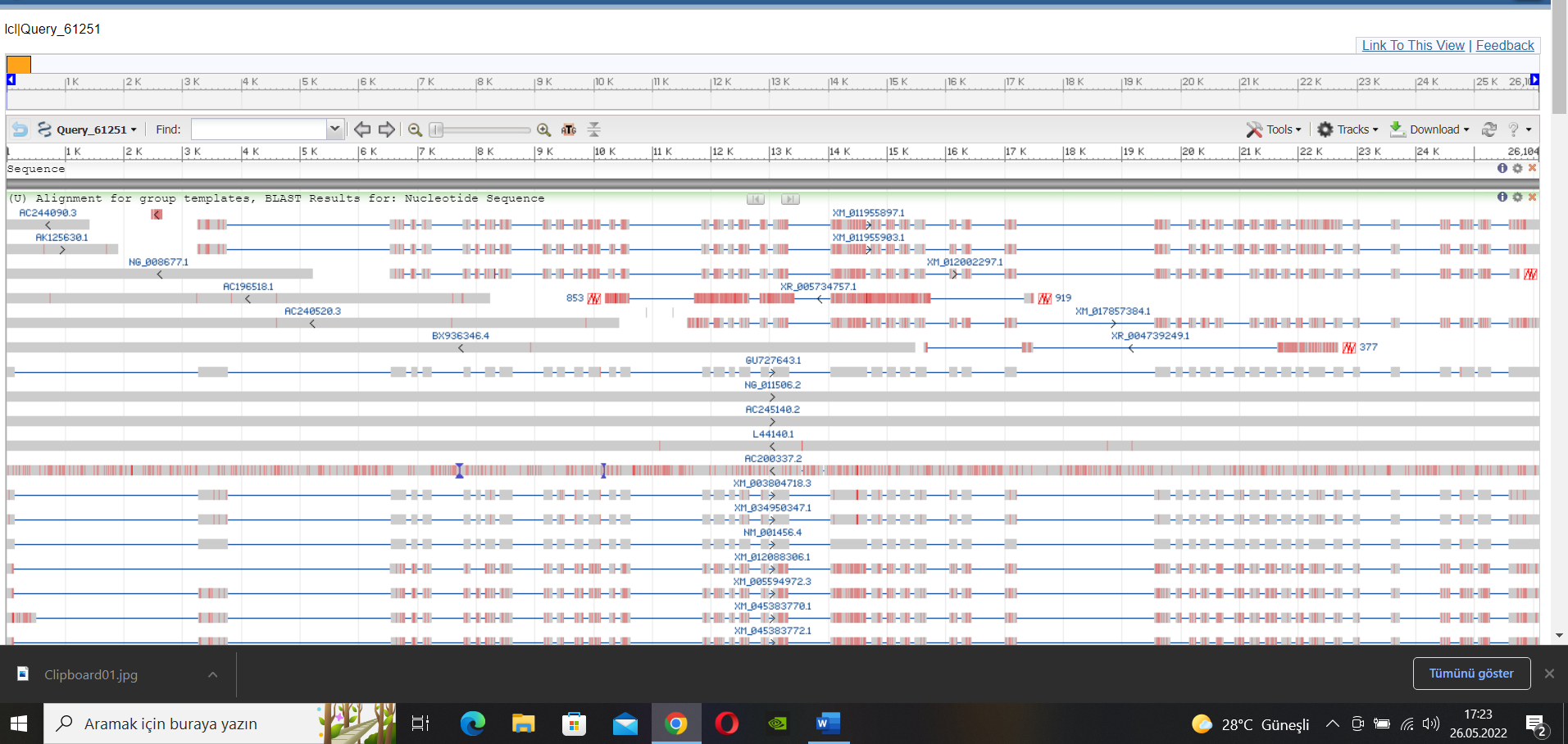


Bu işlemleri yaptıktan sonra aşağıda bulanan blast adlı butona tıklıyoruz.Ardından 1 dakikadan daha az bir süre içerisinde gereken işlemi yaptı ve uyumluluk yüzdeleri ile beraber gerekli bilgileri bize sağladı.

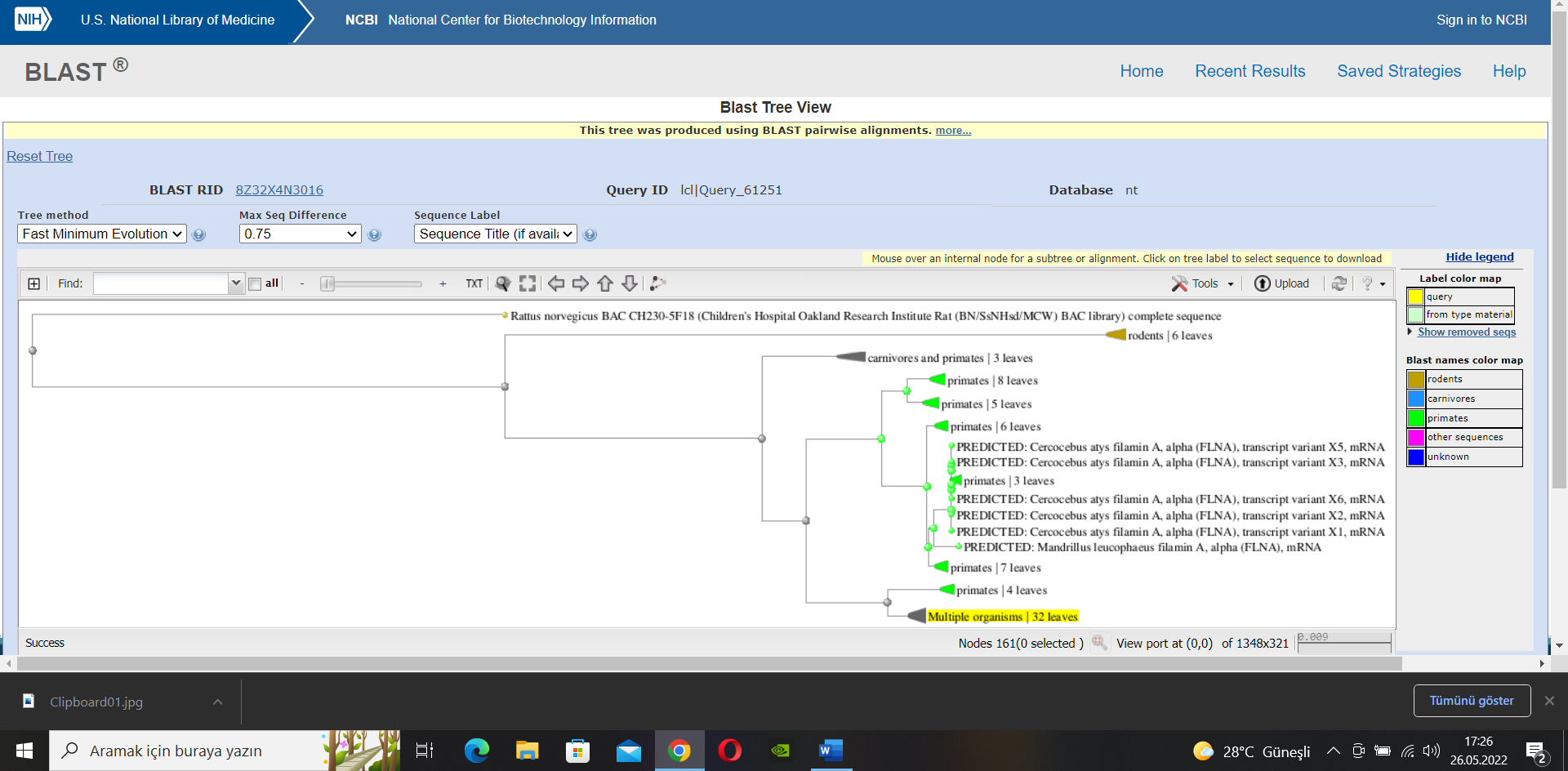




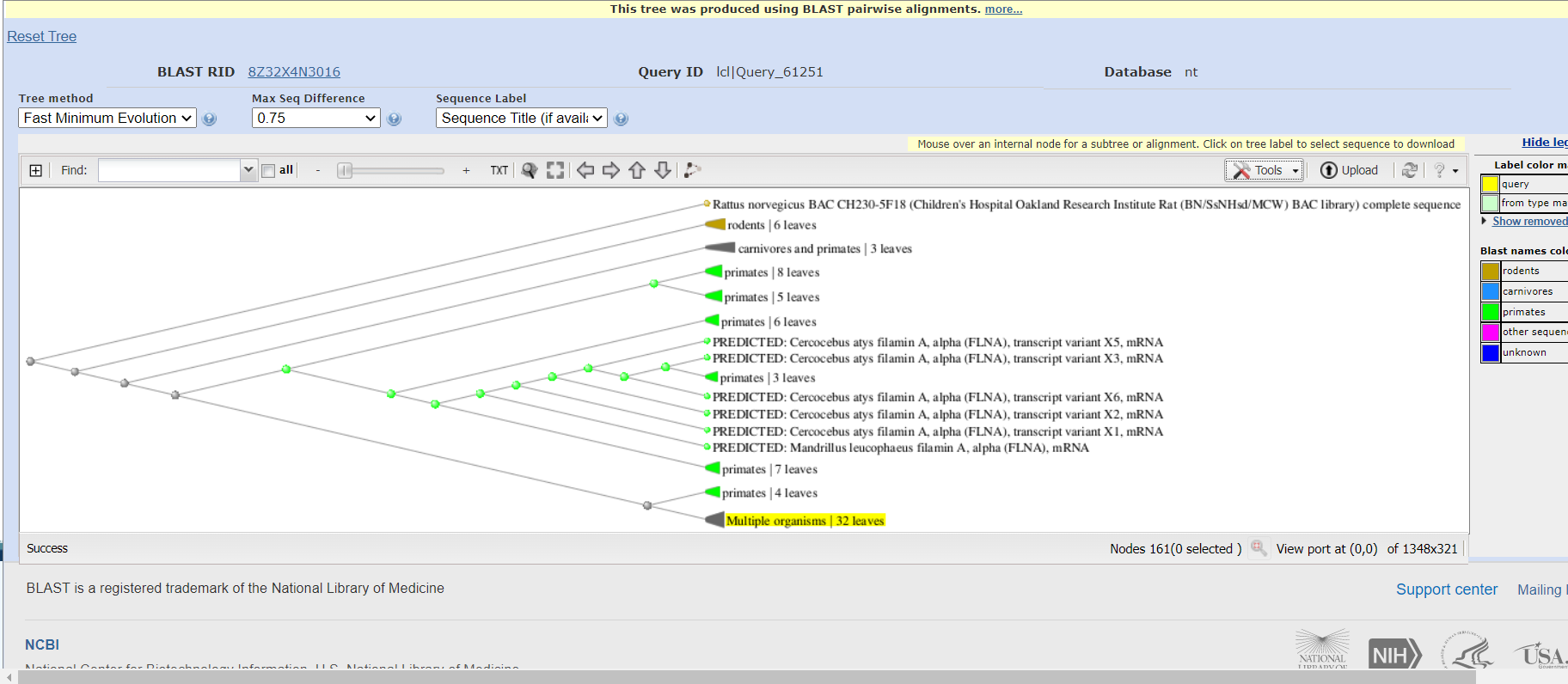
Yüzde 100 ve 99 olan genler yine inşalara ait olduğu görülmektedir. Grafiksel olarak gösterim ise aşağıda mevcuttur.



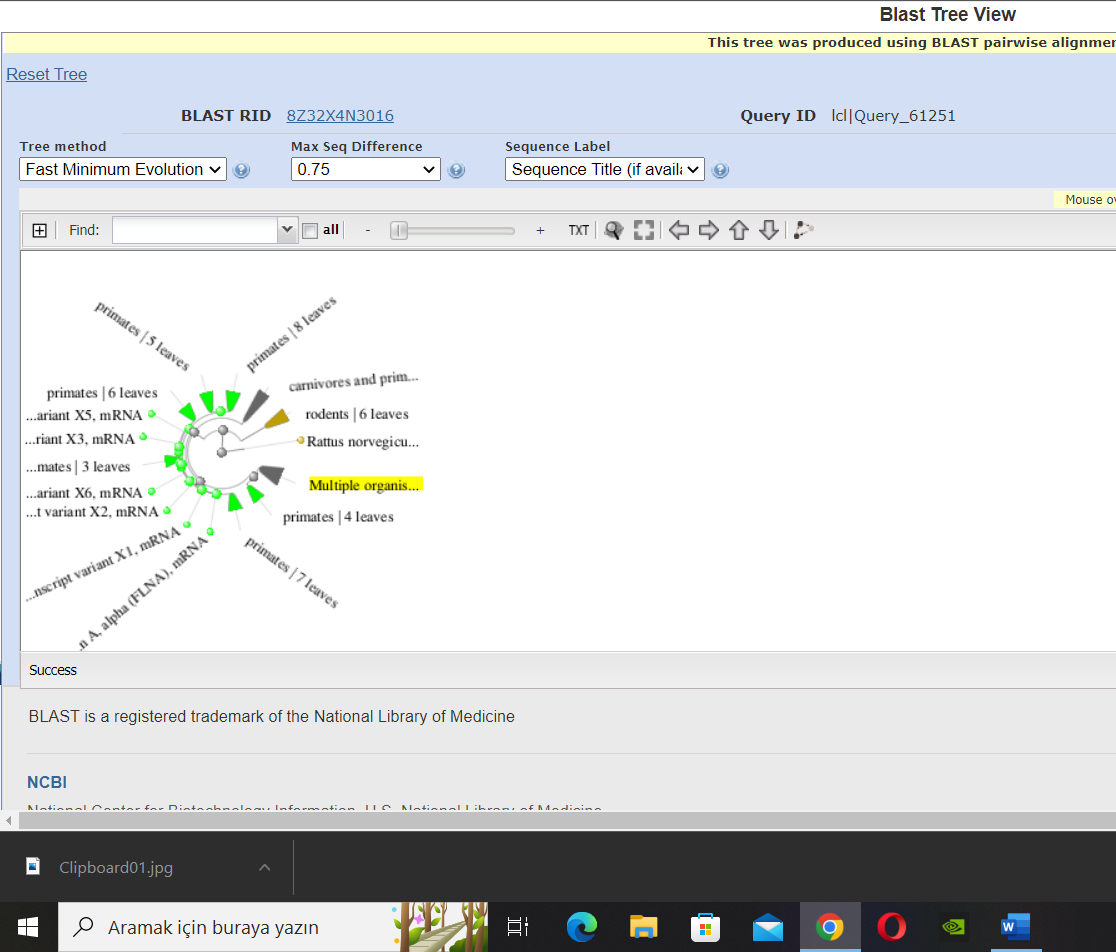
Ardından ağaç yapısını oluşturmak için distance tree of result seçeneğini seçiyoruz.



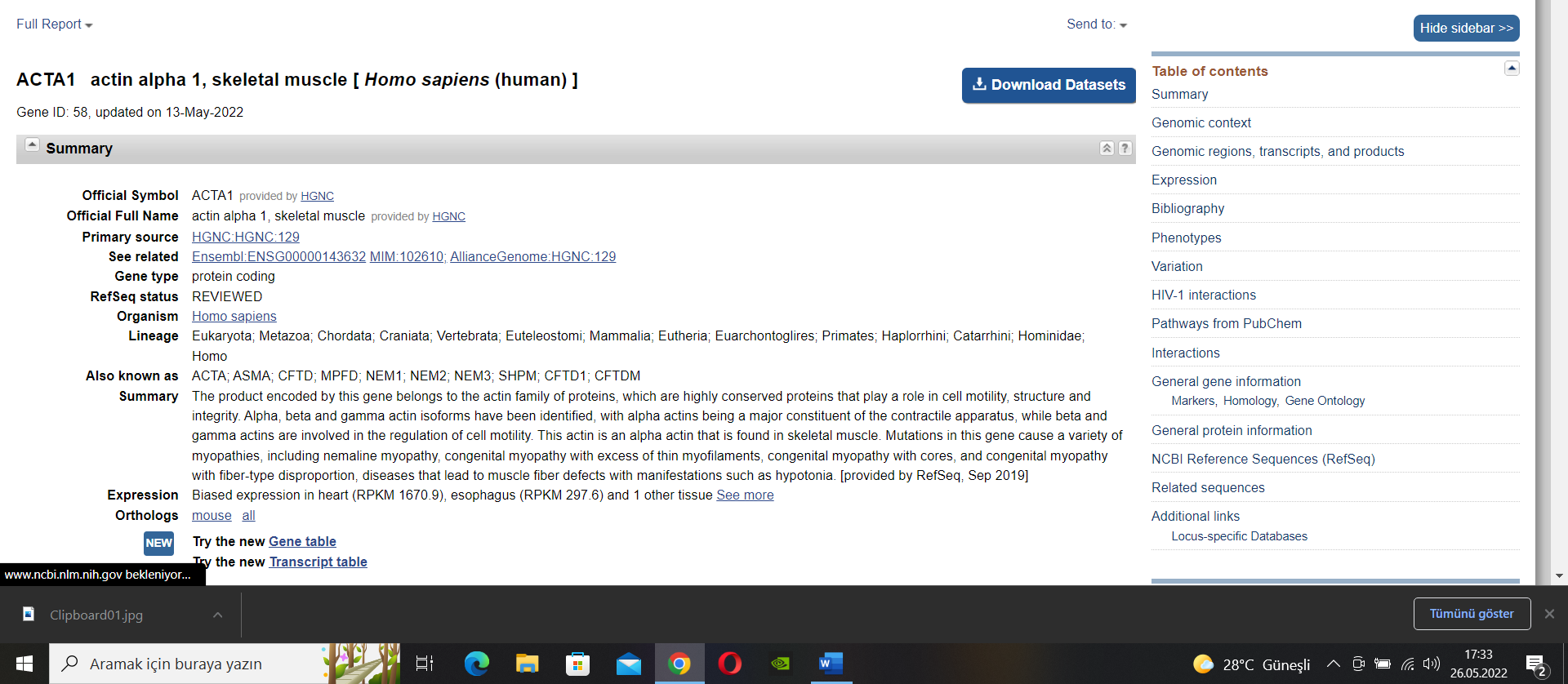
Tools seçeneği altında bulunan Layout türleri ile isterseniz ağaç yapısını farklı şekillerde inceleyebilirsiniz.Slanted cladogram adlı seçeneği seçtikten sonra ağacımızın aldığı şekil.



Farklı bir gösterim daha :



Örnek 2 :Farklı bir veri seti ile 2. bir örnek

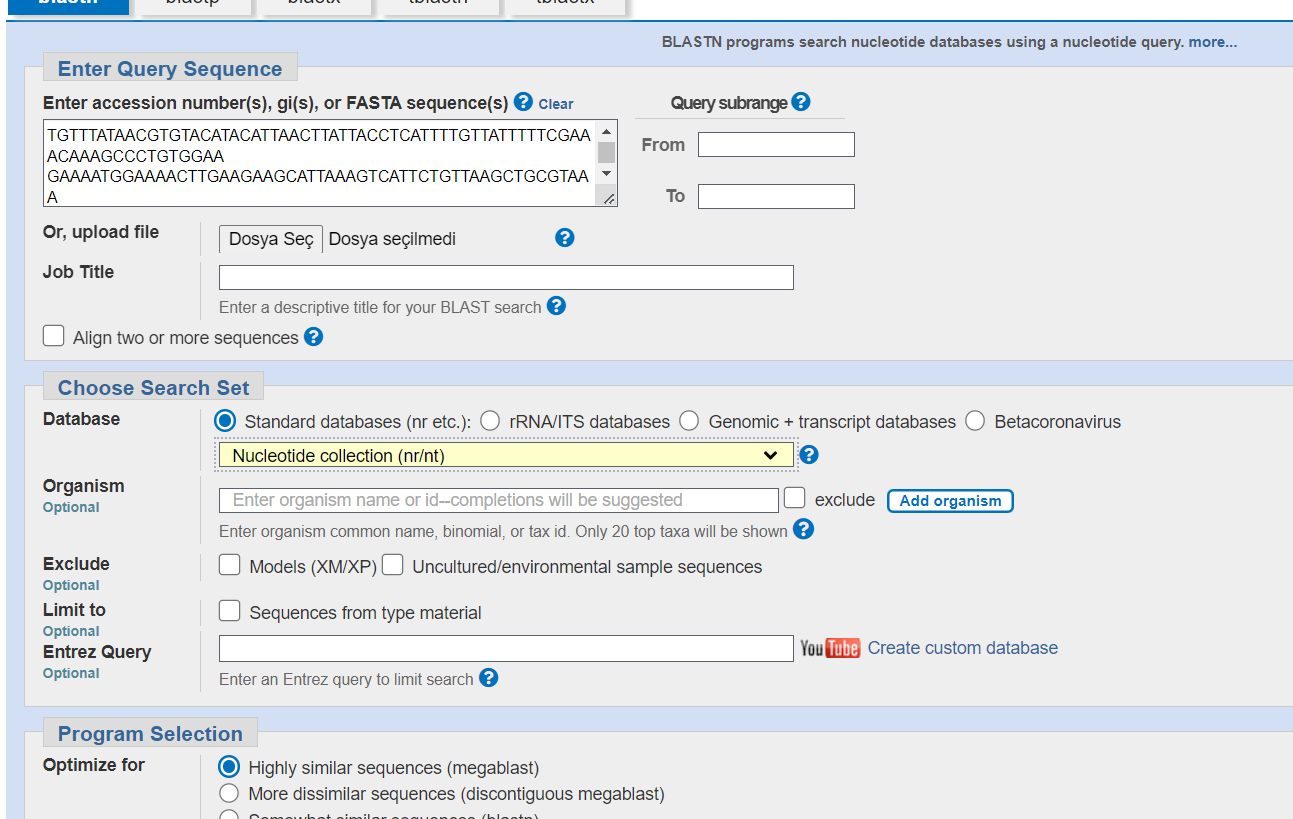


Veri setimiz acta1 adlı veri seti.

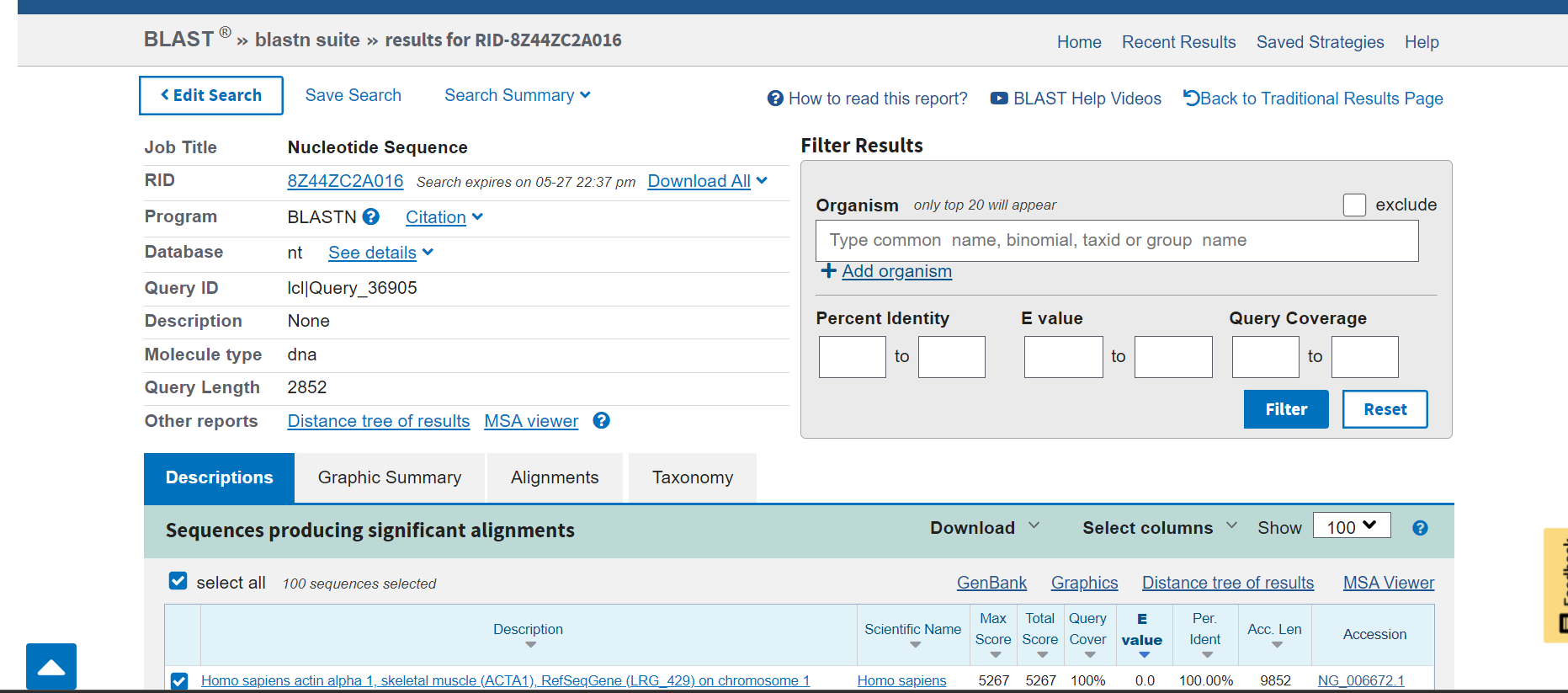
Fasta formatı :



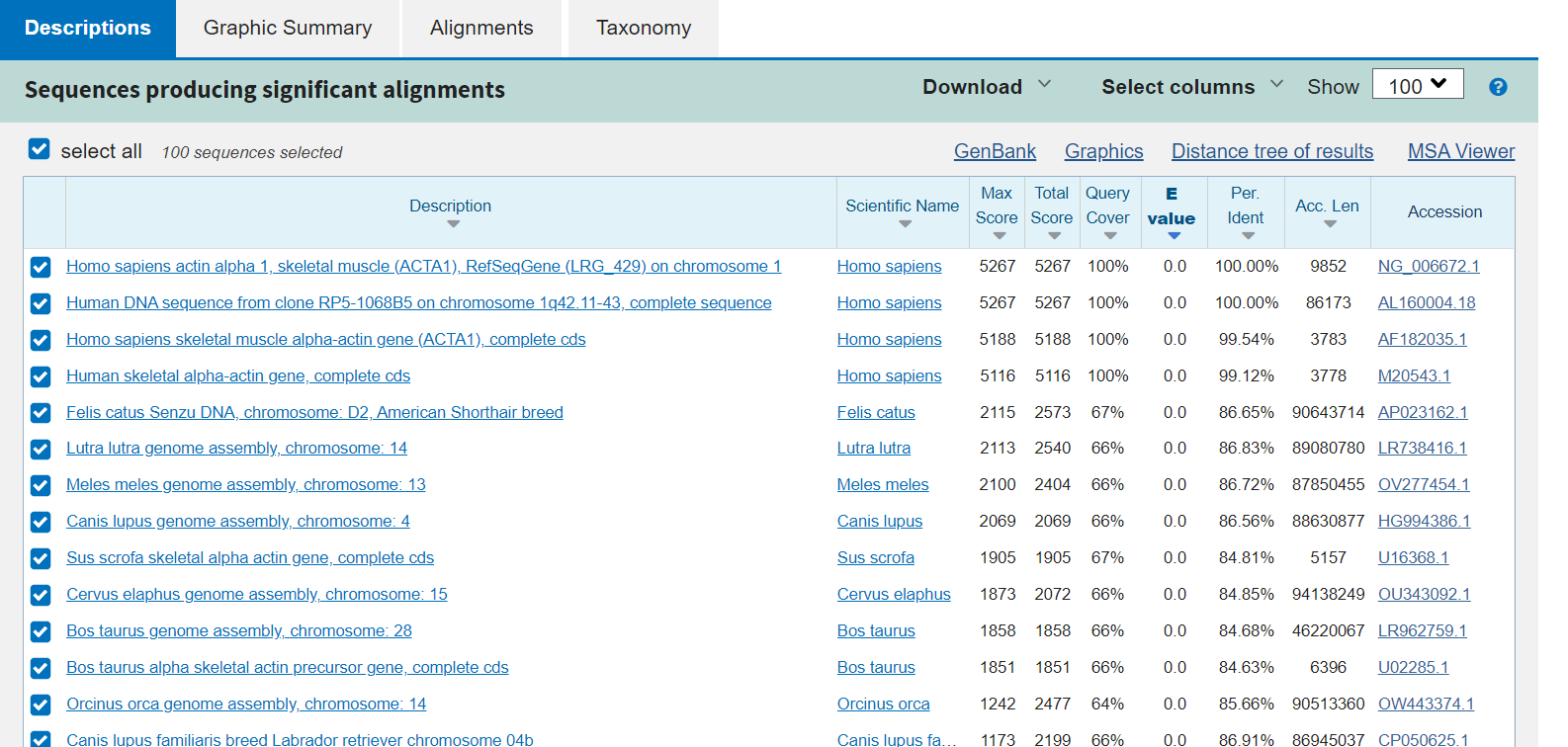
Ardından blasta ortamına geçiyoruz.Ardından ilgili yere genlerimizi yapıştırıyoruz.Ardından veri tabanı ve diğer ayarları yaptıktan sonra blasta butonumuza basıyoruz.



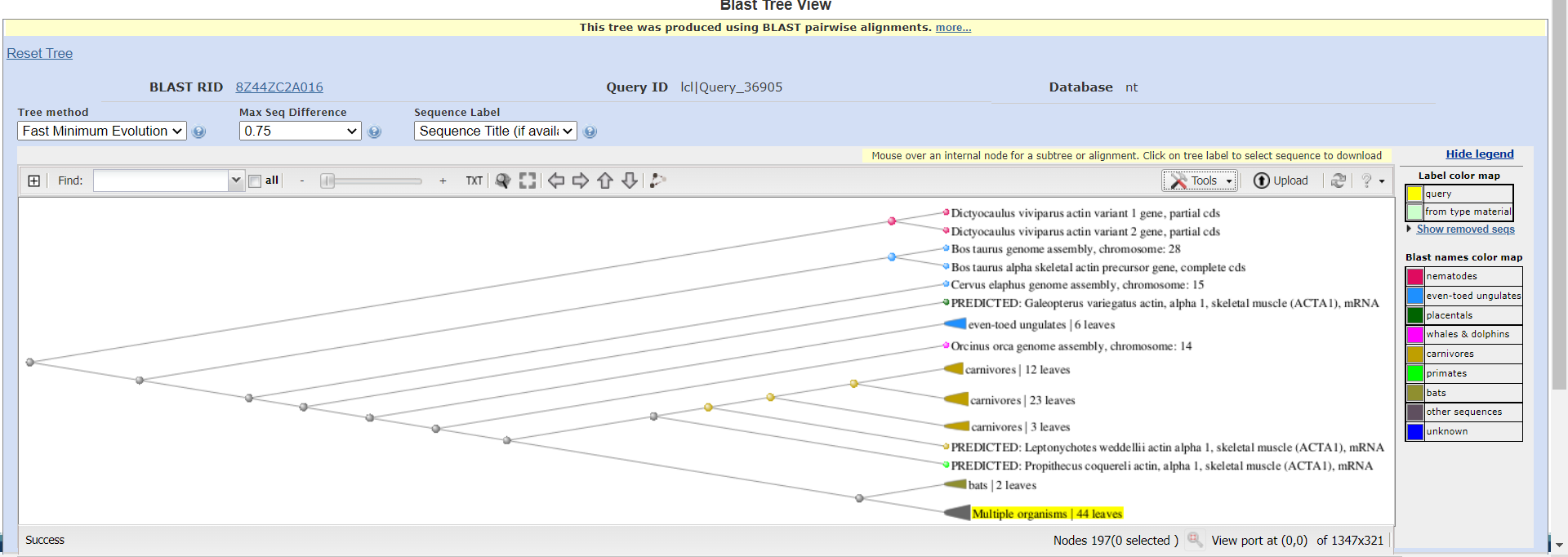
Butona bastıktan sonra kısa bir süre bekledikten sonra sonuçlar karşımıza çıkıyor.



İlgili yüzdeler aşağıda mevcuttur.



Son olarak örnek bir ağaç yapımızı oluşturalım.



Hasan Korkmaz 180508011