**Условие задачи:**

Задание 1.

Дано два биологических вида (см. Приложение А). Для того, чтобы выполнить следующие два задания, необходимо подготовить файл формата GenBank, например, следующим образом:

- На сайте https://www.ncbi.nlm.nih.gov/ для каждого из двух видов сделать запрос название\_вида complete cds. В полученном списке выбрать пять любых записей, скачать их в формате GenBank.

- Из двух полученных файлов средствами Biopython составить единый файл.

Также возможно использование буфера обмена, где временно хранятся выбранные файлы. Совершая несколько запросов и выбирая нужные записи в этот буфер обмена, можно сохранить их все сразу в один файл. Подробности описаны в Приложении Б.

В результате должен получиться единый файл, содержащий в общей сложности по крайней мере 10 различных кодирующих областей (CDS – coding sequence). Пример содержимого такого файла приведён в Приложении Б.

Задание 2.

Входным файлом является файл GenBank, составленный в задании 1. Нужно прочитать все записи и вывести последовательности в порядке возрастания их GC-составов. Например, для файла из приложения Б вывод может выглядеть так:

EU531471.1: Banana bunchy top virus isolate Cumbam replicase gene, complete cds, GC = 0.39605110336817656

EU531473.1: Banana bunchy top virus isolate Andaman replicase gene, complete cds, GC = 0.40185830429732866

EU531472.1: Banana bunchy top virus isolate Solapur replicase gene, complete cds, GC = 0.40185830429732866

GU130587.1: Banana bunchy top virus isolate Delhi replicase gene, complete cds, GC = 0.40185830429732866

GU130586.1: Banana bunchy top virus isolate Bihar replicase gene, complete cds, GC = 0.40185830429732866

BC021958.1: Homo sapiens abhydrolase domain containing 5, mRNA (cDNA clone MGC:8731 IMAGE:3920855), complete cds, GC = 0.4105191256830601

U60808.1: Human CDP-diacylglycerol synthase (CDS) mRNA, complete cds, GC = 0.4188200877620673

BC074833.2: Homo sapiens CDP-diacylglycerol synthase (phosphatidate cytidylyltransferase) 1, mRNA (cDNA clone MGC:104094 IMAGE:30915545), complete cds, GC = 0.43072702331961593

BC074881.2: Homo sapiens CDP-diacylglycerol synthase (phosphatidate cytidylyltransferase) 1, mRNA (cDNA clone MGC:103857 IMAGE:30915239), complete cds, GC = 0.43082191780821916

BK010446.1: TPA\_exp: Homo sapiens PINT87aa (circPINTexon2) gene, complete cds, GC = 0.5461254612546126

Задание 3.

Входным файлом является файл GenBank, составленный в задании 1. Нужно вывести все белковые последовательности, которые получаются в результате трансляции кодирующих областей в заданных последовательностях. При этом должно быть указано, какой участок какой из заданных последовательностей кодируется в полученную белковую последовательность. Для файла из приложения Б вывод может выглядеть так:

BC074881.2: Homo sapiens CDP-diacylglycerol synthase (phosphatidate cytidylyltransferase) 1, mRNA (cDNA clone MGC:103857 IMAGE:30915239), complete cds Coding sequence location = [49:1435](+)

Translation = MLELRHRGSCPGPREAVSPPHREGEAAGGDHETESTSDKETDIDDRYGDLDSRTDSDIPEIPPSSDRTPEILKKALSGLSSRWKNWWIRGILTLTMISLFFLIIYMGSFMLMLLVLGIQVKCFHEIITIGYRVYHSYDLPWFRTLSWYFLLCVNYFFYGETVADYFATFVQREEQLQFLIRYHRFISFALYLAGFCMFVLSLVKKHYRLQFYMFAWTHVTLLITVTQSHLVIQNLFEGMIWFLVPISSVICNDITAYLFGFFFGRTPLIKLSPKKTWEGFIGGFFSTVVFGFIAAYVLSKYQYFVCPVEYRSDVNSFVTECEPSELFQLQTYSLPPFLKAVLRQERVSLYPFQIHSIALSTFASLIGPFGGFFASGFKRAFKIKDFANTIPGHGGIMDRFDCQYLMATFVHVYITSFIRGPNPSKVLQQLLVLQPEQQLNIYKTLKTHLIEKGILQPTLKV

BC074833.2: Homo sapiens CDP-diacylglycerol synthase (phosphatidate cytidylyltransferase) 1, mRNA (cDNA clone MGC:104094 IMAGE:30915545), complete cds Coding sequence location = [49:1435](+)

Translation = MLELRHRGSCPGPREAVSPPHREGEAAGGDHETESTSDKETDIDDRYGDLDSRTDSDIPEIPPSSDRTPEILKKALSGLSSRWKNWWIRGILTLTMISLFFLIIYMGSFMLMLLVLGIQVKCFHEIITIGYRVYHSYDLPWFRTLSWYFLLCVNYFFYGETVADYFATFVQREEQLQFLIRYHRFISFALYLAGFCMFVLSLVKKHYRLQFYMFAWTHVTLLITVTQSHLVIQNLFEGMIWFLVPISSVICNDITAYLFGFFFGRTPLIKLSPKKTWEGFIGGFFSTVVFGFIAAYVLSKYQYFVCPVEYRSDVNSFVTECEPSELFQLQTYSLPPFLKAVLRQERVSLYPFQIHSIALSTFASLIGPFGGFFASGFKRAFKIKDFANTIPGHGGIMDRFDCQYLMATFVHVYITSFIRGPNPSKVLQQLLVLQPEQQLNIYKTLKTHLIEKGILQPTLKV

BC021958.1: Homo sapiens abhydrolase domain containing 5, mRNA (cDNA clone MGC:8731 IMAGE:3920855), complete cds Coding sequence location = [38:1088](+)

Translation = MAAEEEEVDSADTGERSGWLTGWLPTWCPTSISHLKEAEEKMLKCVPCTYKKEPVRISNGNKIWTLKFSHNISNKTPLVLLHGFGGGLGLWALNFGDLCTNRPVYAFDLLGFGRSSRPRFDSDAEEVENQFVESIEEWRCALGLDKMILLGHNLGGFLAAAYSLKYPSRVNHLILVEPWGFPERPDLADQDRPIPVWIRALGAALTPFNPLAGLRIAGPFGLSLVQRLRPDFKRKYSSMFEDDTVTEYIYHCNVQTPSGETAFKNMTIPYGWAKRPMLQRIGKMHPDIPVSVIFGARSCIDGNSGTSIQSLRPHSYVKTIAILGAGHYVY ADQPEEFNQKVKEICDTVD

U60808.1: Human CDP-diacylglycerol synthase (CDS) mRNA, complete cds Coding sequence location = [149:1535](+)

Translation = MLELRHRGSCPGPREAVSPPHREGEAAGGDHETESTSDKETDIDDRYGDLDSRTDSDIPEIPPSSDRTPEILKKALSGLSSRWKNWWIRGILTLTMISLFFLIIYMGSFMLMLLVLGIQVKCFHEIITIGYRVYHSYDLPWFRTLSWYFLLCVNYFFYGETVADYFATFVQREEQLQFLIRYHRFISFALYLAGFCMFVLSLVKKHYRLQFYMFAWTHVTLLITVTQSHLVIQNLFEGMIWFLVPISSVICNDITAYLFGFFFGRTPLIKLSPKKTWEGFIGGFFSTVVFGFIAAYVLSKYQYFVCPVEYRSDVNSFVTECEPSELFQLQTYSLPPFLKAVLRQERVSLYPFQIHSIALSTFASLIGPFGGFFASGFKRAFKIKDFANTIPGHGGIMDRFDCQYLMATFVHVYITSFIRGPNPSKVLQQLLVLQPEQQLNIYKTLKTHLIEKGILQPTLKV

BK010446.1: TPA\_exp: Homo sapiens PINT87aa (circPINTexon2) gene, complete cds Coding sequence location = [478:742](+)

Translation = MLWLPDRGSCSARSPSGMLRGAPGGWRYGRRCGRRRQSCCCCCCCSHVGAPLSFHREASLVSHDGH DIMKQHCGEESIRGAHGYKNK

EU531473.1: Banana bunchy top virus isolate Andaman replicase gene, complete cds Coding sequence location = [0:861](+)

Translation = MARYVVCWMFTINNPTTLPVMRDEIKYMVYQVERGQEGTRHVQGYVEMKRRSSLKQMRGFFPGAHLEKRKGSQEEARSYCMKEDTRIEGPFEFGAFKLSCNDNLFDVIPDMRETHKRPLEYLYDCPNTFDRSKDTLYRVQAEMNKTKAMNSWRTSFSAWTSEVENIMAQPCHRRIIWVYGPNGGEGKTTYAKHLMKTRNAFYSPGGKSLDICRTVYYEDIVIFDIPRCKEDYLNYGLLEEFKNGIIQSGKYEPVLKIVEYVEVI VMANFLPKEGIFSEDRIKLVAC

EU531472.1: Banana bunchy top virus isolate Solapur replicase gene, complete cds Coding sequence location = [0:861](+)

Translation = MARYVVCWMFTINNPTTLPVMRDEIKYMVYQVERGQEGTRHVQGYVEMKRRSSLKQMRGFFPGAHLEKRKGSQEEARSYCMKEDTRIEGPFEFGAFKLSCNDNLFDVIQDMRETHKRPLEYLYDCPNTFDRSKDTLYRVQAEMNKTKAMNSWRTSFSAWTSKGENIMAHPCHRRIIWVYGPKGGEGKTTYAKHLMKTRNAFYSPGGKSLDICRTVYYEDIVIFDIPRCKEDYLNYGLLEEFKNGIIQSGKYEPVLKIVEYVEVI VMANFLPKEGIFSEDRIKLVAC

EU531471.1: Banana bunchy top virus isolate Cumbam replicase gene, complete cds Coding sequence location = [0:861](+)

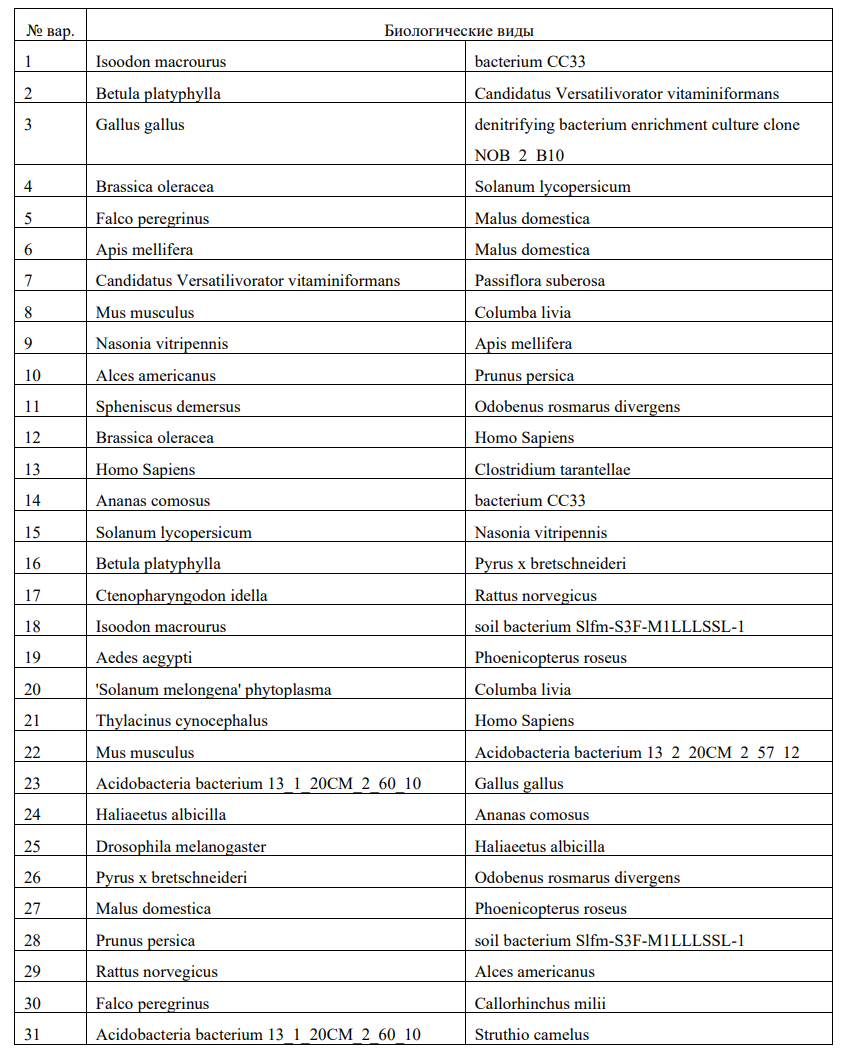
Translation = MARYVVCWMFTINNPTTLPVMRDEIKYMVYQVERGQEGTRHVQGYVEMKRRSSLKQMRGFFPGAHLEKRKGSQEEARSYCMKEDTRIEGPFEFGAFKLSCNDNLFDVIQDMRETHKRPLEYLYDCPNTFDRSKDTLYRVQAEMNKTNAMNSWRTSFSAWTSDVENIMTQPCHRRIIWVYGPNGGEGKTTYAKYLMKTRNAFYSPGGKSLDICRTVYYEDIVIFDIPRCKEEYLNYGLLEEFKNGIIQSGKYEPVLKIVEYVEVI VMANFLPKEGIFSEDRIKLVAC  
  
Coding sequence location = [0:861](+) Translation = MARYVVSWMFTINNPTTLPVMRDEIKYMVYQVERGQEGTRHVQGYVEMKRRSSLKQMRGFFPGAHLEKRKGSQEEARSYCMKEDTRIEGPFEFGAFKLSCNDNLFDVIQDMRETHKRPLEYLYDCPNTFDRSKDTLYRVQAEMNKTKAMNSWRTSFSAWTSEVENIMAQPCHRRIIWVYGPNGGEGKTTYAKHLMKTRNAFYSPGGKSLDICRLYNYEDIVIFDIPRCKEDYLNYGLLEEFKNGIIQSGKYEPVLKIVEYVEVI VMANFLPKEGIFSEDRLKLVAC

GU130586.1: Banana bunchy top virus isolate Bihar replicase gene, complete cds Coding sequence location = [0:861](+)

Translation = MARYVVCWMFTINHPTTLPVMRDEIKYMVYQVERGQEGTRHVQGYVEMKRRSSLKQMRGFFPGAHLEKRKASQEEARSYCMKEDTIIEGPFEFGAFKLSCNDNLFDVIQDMRETHKRPLEYLYDCPNTFDRSKDTLYRVQAEMNKTKAMNSWRTSFSAWTSEVENIMAQPCHRRIIWVYGPNGGEGKTTYAKHLMKTRNAFYSPGGKSLDICRLYNYEDIVIFDIPRCKEDYLNYGLLEEFKNGIIQSGKYERVLKIVEYVEVI VMANFLPKEGIFSEDRIKLVSC

**ПРИЛОЖЕНИЕ А**

**ВАРИАНТЫ ДЛЯ ЗАДАНИЙ**

****

Мой выбор пал на 13ю строку

**ПРИЛОЖЕНИЕ Б**

**ЭКСПОРТ ДАННЫХ ИЗ БАЗЫ ДАННЫХ GENBANK**

Для этого первоначально зайдём на сайт сайт <https://www.ncbi.nlm.nih.gov/> . У строки поиска в выпадающем списке выбираем Nucleotide и вводим нужный запрос, например Homo Sapiens complete CDS.При нажатии на кнопку Search или клавишу Enter появляется список, как на рисунке ниже. Выберем, например, пять первых записей (1). Для их сохранения затем нажимаем кнопку Send to (2), после чего выплывает вот такое меню. Можно, например, далее выбрать File (1), потом в новом всплывшем списке опций – формат GenBank (2) и скачать файл формата GenBank с пятью выбранными записями (3).

**Анализ задачи:**

**Задание 1:** Требуется подготовить GenBank файл, содержащий данные о двух биологических видах.

**Задание 2:** Необходимо прочитать GenBank файл, созданный в Задании 1, и вывести последовательности в порядке возрастания их GC-составов. GC-состав – это доля гуанина и цитозина в последовательности ДНК.

**Задание 3:** Требуется прочитать тот же GenBank файл и вывести белковые последовательности, полученные в результате трансляции кодирующих областей. При этом необходимо указать, какой участок какой последовательности кодирует данную белковую последовательность.

**Написание кода:**

Из задания приложения Б мы уже скачали данные о Homo Sapiens c сайта сайт <https://www.ncbi.nlm.nih.gov/>. Скачаем (если ещё не скачали) библиотеку biopython через терминал (pip install biopython) и импортируем из него SeqIO для работы с GenBank.

Далее импортируем функцию gc\_fraction из модуля Bio.SeqUtils библиотеки Biopython. Эта функция используется для вычисления GC-состава (процентное содержание гуанина и цитозина) в нуклеотидной последовательности.

1. Создадим переменную **genbank\_file** , которая будет содержать путь к вашему файлу GenBank.
2. Создадим пустой список **cds\_entries,** который будет использоваться для хранения информации о каждой кодирующей последовательности (CDS), найденной в файле GenBank.

Дальше нам нужно сделать цикл, который перебирает каждую запись (record) в файле GenBank.Используем SeqIO.parse() - функция из Biopython, которая читает файл и возвращает объекты SeqRecord, представляющие каждую запись. "genbank" указывает, что файл должен быть проанализирован как файл GenBank.

Вложим в этот цикл ещё 1 цикл, который перебирает каждую "feature" (характеристику) в текущей записи record. Features - это аннотации в записи GenBank, которые описывают различные биологические элементы (например, гены, CDS, и т.д.).Уже в этот цикл мы вложим условный оператор if, который проверяет, является ли текущая feature кодирующей последовательностью (CDS) и содержит ли она информацию о трансляции.

Сделаем проверку, является тип feature равен "CDS". feat.type == "CDS"

Проверим, что feature имеет квалификатор с ключом "translation". Квалификаторы предоставляют дополнительную информацию о feature.

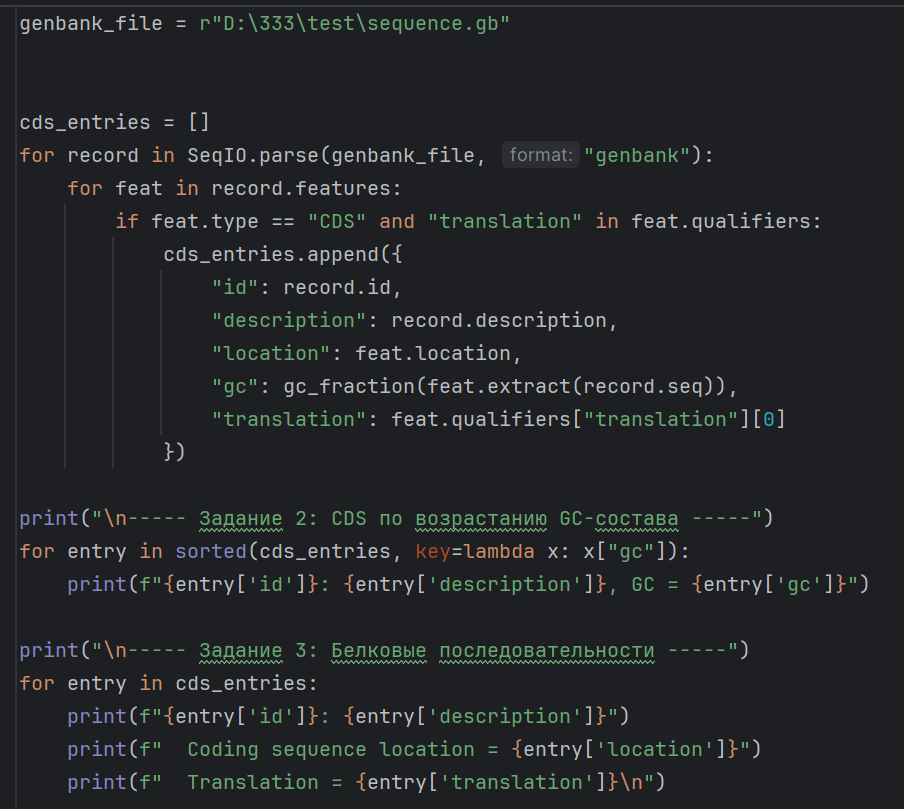
Сделаем так, что если условие выполняется (то есть это CDS с информацией о трансляции), создается словарь, содержащий информацию об этой CDS, и добавляется в список **cds\_entries**.

Добавим идентификатор записи в словарь, описание записи в словарь, информацию о местоположении CDS на последовательности, вычислим GC-состав CDS, извлечём последовательность, соответствующую CDS, из общей последовательности записи, вычислим GC-состав извлеченной последовательности и извлечём транслированную аминокислотную последовательность из квалификатора "translation". [0] используется, чтобы получить первую трансляцию, так как квалификатор "translation" может содержать список.

Самая сложная часть позади. Осталось только вывести заголовок для результатов Задания 2, отсортировать список **cds\_entries** на основе GC-состава и затем перебрать отсортированный список.

Для 2го задания нам необходимо сделать почти тоже самое: вывести заголовок для результатов Задания 3, перебрать все CDS entries в списке, вывести идентификатор и описание записи, местоположение кодирующей последовательности и транслированную аминокислотную последовательность.

**Написание кода**

****

from Bio import SeqIO  
from Bio.SeqUtils import gc\_fraction  
  
genbank\_file = r"D:\333\test\sequence.gb"  
  
  
cds\_entries = []  
for record in SeqIO.parse(genbank\_file, "genbank"):  
 for feat in record.features:  
 if feat.type == "CDS" and "translation" in feat.qualifiers:  
 cds\_entries.append({  
 "id": record.id,  
 "description": record.description,  
 "location": feat.location,  
 "gc": gc\_fraction(feat.extract(record.seq)),  
 "translation": feat.qualifiers["translation"][0]  
 })  
  
print("\n----- Задание 2: CDS по возрастанию GC-состава -----")  
for entry in sorted(cds\_entries, key=lambda x: x["gc"]):  
 print(f"{entry['id']}: {entry['description']}, GC = {entry['gc']}")  
  
print("\n----- Задание 3: Белковые последовательности -----")  
for entry in cds\_entries:  
 print(f"{entry['id']}: {entry['description']}")  
 print(f" Coding sequence location = {entry['location']}")  
 print(f" Translation = {entry['translation']}\n")

**Проверка кода**

Вывод очень большой и не влезет целиком, но вы сами можете убедиться в работоспособности кода.

**Всё готово!**