1. **Prokka:**
   1. Installation:
      1. conda create -n prokka\_env python=2.7.18
      2. source activate prokka\_env
      3. conda install -c biobuilds perl=5.22
      4. conda install -c bioconda prokka
   2. Use:
      1. prokka [inputFileName] --outdir [outDirectoryPath] --prefix metagG --metagenome --kingdom Bacteria
2. **DeepARG**:
   1. Installation:
      1. conda create -n deeparg\_env python=2.7.18
      2. source activate deeparg\_env
      3. conda install -c bioconda diamond==0.9.24
      4. pip install deeparg==1.0.2
      5. deeparg download\_data -o /path/to/local/directory/
   2. Use:
      1. deeparg predict --model [SS or LS] --type [nucl or prot] --input /path/file.fasta --out /path/to/out/file.out
      2. SS for short sequence, LS for long sequence
      3. nucl for nucleotide FASTA files, prot for protein FASTA files
   3. Example:
      1. deeparg predict --model SS --type nucl --input /home/team2/03.functional.annotation/rb\_clustering/clust\_fna/all.cfinal.fasta --out /home/sbelhareth3/deepargOutputs -d /home/sbelhareth3/deeparg/None
3. **Phobius:**
   1. Installation:
      1. Fill the webform on <https://software.sbc.su.se/cgi-bin/request.cgi?project=phobius> and the file will be sent via email almost instantly
      2. gunzip -c phobius101\_linux.tar.gz | tar xf -
   2. Correction needed:
      1. Edit phobius.pl L25 to
         1. $DECODEANHMM = "$PHOBIUS\_DIR/decodeanhmm.64bit"
   3. Usage:
      1. perl phobius.pl -short input.faa > file.out
   4. Note:
      1. Phobius doesn’t run if the sequences are followed by ‘\*’ that indicates a stop codon.
      2. Execute the following command to modify the .faa files
         1. sed -i 's/\*//g' \*.faa
4. **CRISPR Recognition Tool:**
   1. File extraction
      1. Go to website <http://www.room220.com/crt/>
      2. Download CRT1.2-CLI.jar.zip
      3. Unzip the file and there should be a .jar file.
   2. Command to run
      1. java -cp CRT1.2-CLI.jar crt <inputFile> <outputFile>
      2. Options are not necessary (all options are related to length control or screen).
5. **Interproscan:**
   1. Requirements
      1. 64-bit Linux (only works on Linux)
      2. Perl 5 (default on most Linux distributions)
      3. Python 3 (InterProScan 5.30-69.0 onwards)
      4. Java JDK/JRE version 11 (InterProScan 5.37-76.0 onwards)
         1. I used conda installation - <https://anaconda.org/conda-forge/openjdk>
      5. Environment variables set (I have not done this part but still worked)
         1. $JAVA\_HOME should point to the location of the JVM
         2. $JAVA\_HOME/bin should be added to the $PATH
   2. Installation
      1. mkdir my\_interproscan
      2. cd my\_interproscan
      3. wget <ftp://ftp.ebi.ac.uk/pub/software/unix/iprscan/5/5.50-84.0/interproscan-5.50-84.0-64-bit.tar.gz>
      4. wget <ftp://ftp.ebi.ac.uk/pub/software/unix/iprscan/5/5.50-84.0/interproscan-5.50-84.0-64-bit.tar.gz.md5>
         1. # Recommended checksum to confirm the download was successful:
      5. md5sum -c interproscan-5.50-84.0-64-bit.tar.gz.md5
         1. Must return \*interproscan-5.50-84.0-64-bit.tar.gz: OK\*
         2. If not - try downloading the file again as it may be a corrupted copy.
      6. tar -pxvzf interproscan-5.50-84.0-\*-bit.tar.gz where:
         1. p = preserve the file permissions
         2. x = extract files from an archive
         3. v = verbosely list the files processed
         4. z = filter the archive through gzip
         5. f = use archive file
      7. python3 initial\_setup.py
   3. Command
      1. • ./interproscan.sh -appl CDD,COILS,Gene3D,HAMAP,MobiDBLite,PANTHER,Pfam,PIRSF,PRINTS,PROSITEPATTERNS,PROSITEPROFILES,SFLD,SMART,SUPERFAMILY,TIGRFAM -i <input\_file> -f <format> -o <output\_file>
      2. For format, you can just type gff3.
6. **SignalP**
   1. Installation:
      1. tar -vxf signalp-5.0b.Linux.tar.gz
7. **VFDB and Microbesonline:** 
   1. <https://compgenomics2020.biosci.gatech.edu/Team_II_Functional_Annotation_Group#MicrobesOnline>
   2. Install miniconda first to install dependencies:
      1. wget <https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh>
      2. ./Miniconda3-latest-Linux-x86\_64.sh
      3. Chmod 755 ./Miniconda3-latest-Linux-x86\_64.sh
   3. Step 2:
      1. Need blast plus to run the bash scripts: This works for both the bash scripts: from here <https://anaconda.org/biocore/blast-plus>
      2. Need python to run the blast2gff.py script (found a script that changes the blast outfmt 6 to gff)
8. **USEARCH**
   1. Installation:
      1. wget <https://www.drive5.com/downloads/usearch11.0.667_i86linux32.gz>
      2. gunzip –k usearch11.0.667\_i86linux32.gz
   2. Usage:
      1. path\_to\_usearch -cluster\_fast temp/\*.fna -id 0.97 -centroids temp/centroids.fa -uc temp/label\_seq.fa
9. **PlasmidSeeker**
   1. Installation:
      1. git clone https://github.com/bioinfo-ut/PlasmidSeeker
      2. cd PlasmidSeeker
      3. conda install r r-base
   2. Running:
      1. cd PlasmidSeeker
      2. python ./PlasmidSeeker.py
   3. Inside PlasmidSeeker.py:
      1. import os
      2. final\_outputs\_directory = "/home/team2/02.gene.prediction/Final\_Outputs/"
      3. for filename in os.listdir(final\_outputs\_directory):
      4. if filename.endswith(".fna"):
      5. period\_index = filename.find(".")
      6. file\_prefix = filename[:period\_index]
      7. os.system("./seqtk/seqtk seq -F '40' " + filename + " > " + file\_prefix + "../Fastq\_Outputs/fastq")
      8. final\_outputs\_directory = "../Fastq\_Outputs/"
      9. concatenated\_prefix = "concatenated"
      10. concatenated = final\_outputs\_directory + concatenated\_prefix + ".fastq"
      11. os.system("cat \"\" > " + concatenated)
      12. for filename in os.listdir(final\_outputs\_directory):
      13. if filename.endswith(".fastq"):
   4. os.system("cat " + final\_outputs\_directory + filename + " >> " + concatenated)
   5. os.system("perl plasmidseeker.pl --ponly -d db\_w20 -i " + concatenated + " -b e\_coli\_sakai\_ref.fna -o ./Outputs/" + concatenated\_prefix + ".txt")