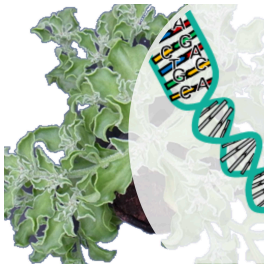
Mar 01, 2023 Version 1

Methods in "The first released available genome of the common ice plant (Mesembryanthemum crystallinum L.) extended the research region on salt tolerance, C3-CAM photosynthetic conversion, and halophism" V.1



DOI
dx.doi.org/10.17504/protocols.io.6qpvr4qdogmk/v1

Ryoma Sato[1], Yuri Kondo[1], Sakae Agarie[2]

[1]Graduate school of Bioresource and Bioenvironmental Sciences, Kyushu University;
[2]Faculty of Agriculture, Kyushu university

Sakae Agarie: Corresponding author;

Ryoma Sato

OPEN ACCESS

DOI: https://dx.doi.org/10.17504/protocols.io.6qpvr4qdogmk/v1

Protocol Citation: Ryoma Sato, Yuri Kondo, Sakae Agarie 2023. Methods in "The first released available genome of the common ice plant (Mesembryanthemum crystallinum L.) extended the research region on salt tolerance, C3-CAM photosynthetic conversion, and halophism". **protocols.io https://dx.doi.org/10.17504/protocols.io.6qpvr4qdogmk/v1**

Protocol status: Working
We use this protocol and it's working

Created: February 28, 2023

Last Modified: March 01, 2023

Protocol Integer ID: 77834

Keywords: De novo shotgun-genome assembly, the common ice plant, halophytes, available genome of the common ice plant, type seeds of the common ice plant, common ice plant, available genome, search for genomic sequence, genome, phylogenetic tree, genomic sequence, genomic dna, completeness of the genome, frozen leaf, gene prediction, gene, dna, type seed, seedling

Funders Acknowledgements:
JSPS KAKENHI
Grant ID: 21K19120

Abstract

The wild-type seeds of the common ice plant were sowed on a germination medium. The seedlings were grown in particular soil and treated with a solution that included salt and nutrients for two weeks in a greenhouse. Genomic DNA was extracted from frozen leaves and made into a library using special kits. Obtained NGS data were trimmed and assembled by *Musket*, *ALGA*, and *Redundans*. The completeness of the genome was checked using BUSCO and BLASTN.

In this protocol, five types of analysis methods were introduced, including the establishment of a phylogenetic tree based on 18S rDNA via *NGPhylogeny.fr,* detection of repetitive regions with *RepeatModeler2*, *TEclass*, and *RepeatMasker*, search for genomic sequences coding tRNA and miRNA by *tRNAscan-SE2.0* and *Infernal*, gene prediction using *BRAKER2* and *DIAMOND*, and protein domain searches based on Pfam database using HMMER.

Materials

Seeds of the common ice plant (Mesembryanthemum crystallinum) were personally provided by Dr. John C. Cushman from the University of Nevada and stored under coolness and darkness until use. Originally, wild-type seeds were collected from the plants identified by Dr. Klaus Winter, an expert on the common ice plant, on a coastal cliff at the Mediterranean Sea shore close to Caesarea in Israel (around 32° 29′ 43.4″N, 34° 53′ 22.8″E) in 1978 (**Winter et al. 1978**). Three voucher specimens of *M. crystallinum* have been deposited in the Herbarium at the Royal Botanic Gardens Kew (55793.000, K000296094, and K000267571). In this study, our biological materials were recognized as the same plants as those specimens.

Experiments, including collecting samples for this study, were conducted in compliance with relevant institutional, national, and international guidelines and laws.

The seeds were aseptically sown on a medium for germination containing 4.6 g MS salt (mixed salts for Murashige-Skoog medium) 30 g sucrose , 1 mL B5 vitamin (**Gamborg et al., 1968**; 1 g nicotinic acid , 1 g pyridoxine hydrochloride , 10 g thiamine hydrochloride , and 100 g myo-inositol per 1 L of B5 vitamin), 0.8 % (w/w) agarose , and pH 5.7 per 1 L . The raising of seedlings was performed according to the methods published by **Agarie et al. (2009)**.

The two-week-old seedlings grown in a growth chamber under 12:00:00 of light and 12:00:00 of darkness at 25 °C were transferred to plastic pots filled with the growth medium soils composed of 50 % peat moss , 30 % cocopeat , and 20 % perlite , specified for the ice plants (Japan Agricultural Cooperatives Ito-Shima, Fukuoka, Japan) and irrigated with a nutrient solution of 1.5 g per one litter OAT House No. 1 and 1.0 g per one litetr OAT House No. 2 (OAT Agrio Co., Ltd., Tokyo, Japan) in a greenhouse at Kyushu University 33°35′35.1″N 130°12′53.2″E for five weeks.

The plants were treated with the liquid solution including 0.3 % (w/w) NaCl for two weeks. Approximately 0.6 g of tissue from each leaf was collected, quickly frozen in liquid nitrogen, and stored at -80 °C .

Troubleshooting

Table of contents

① DNA extraction, library construction, and sequencing

2 Total genomic DNA was extracted from the leaf tissue and purified using MagExtractor™-Plant Genome Nucleic Acid Purification Kits (Toyobo Co., Ltd., Shiga, Japan), according to the manufacturer's instructions. The DNA samples were fragmented by sonication and used to construct short insert paired-end libraries construction using NEBNext® Ultra™DNA Library Prep Kits for Illumina (New England Biolabs Ltd., Ipswich, MA, USA). Briefly, in the end-repair step, fragmented DNA was phosphorylated at the 5' end and adenylated at the 3' end. During the ligation step, full-length circulated adaptor sequences were ligated to the fragments. After adaptor cleavage, purification, and size selection were performed. The indexed PCR products were taken to obtain the final sequencing libraries. The mean insert size for paired-end libraries was 300 bp. The paired-end (2×150 bp) sequencing was conducted on an Illumina NovaSeq 6000 platform (Illumina Inc., San Diego, CA, USA).

② Clean read preparation and genome size estimation

3 The mean insert size was calculated using REAPR (v1.0.18)(**Hunt *et al.* 2013**), and raw paired-end sequences were filtered based on the frequency of 21-mer sequences using the program Musket (v1.1)(**Liu *et al.* 2013**).
The key parameter values were as follows: musket -omulti output -inorder pair1.fastq pair2.fastq.

```
#download of musket
#download from https://sourceforge.net/projects/musket/ and moved it to
the DDBJ NIG sUPER COMPUTER server using SFTP command
cd $HOME
tar xvzf musket-1.1.tar.gz
cd musket-1.1/
make
./musket #check
cd $
~/musket-1.1/musket -omulti output -inorder pair1.fastq
pair2.fastq -p 12
```

Sequence reads that appeared rarely or abnormally frequently were removed to obtain clean read data. In the corrected reads, unique and duplicate read numbers in the corrected reads were measured using fastqc (v0.11.9)(**Simon 2010**). The clean data were used for an estimate of genome size as follows.

```
#quality check (fastqc, multiqc)
fastqc XX.fq.gz -o XX
multiqc ./
```

K-mers were counted and exported to histogram files using jellyfish (v2.3)(**Marçais and Kingsford 2011**) [key parameter: jellyfish histo reads.jf]. GenomeScope2.0(**Ranallo-Benavidez *et al.* 2020**) corresponding key parameters were applied to calculate the genome sizes using *k*-mers lengths of 21 and 25.

```
#How to estimate genome size using GenomeScope2.0

#REFERENCES:
#GenomeScope 2.0 for estimating genome size and heterozygosity of
ploidy genomes from WGS reads
#Original article ↓ Article
#Ranallo-Benavidez, T. R., Jaron, K. S., & Schatz, M. C. (2020).
#GenomeScope 2.0 and Smudgeplot for reference-free profiling of
polyploid genomes. nature communications, 11(1), 1432.
https://doi.org/10.1038/s41467-020-14998-3

git clone https://github.com/tbenavi1/genomescope2.0.git
cd genomescope2.0/
mkdir ~/R_libs
echo "R_LIBS=~/R_libs/" >> ~/.Renviron
#Rscript install.

Move raw data (fastq format) to the specified directory
First, analyze the fastq file.

jellyfish count -C -m 21 -s 1000000000 -t 12 *fq -o reads_k21.jf
jellyfish count -C -m 25 -s 1000000000 -t 12 *fq -o reads_k25.jf

#-m Length of mer
#-s Initial hash size
#-t Number of threads (1)

Output a histogram file.

jellyfish histo -t 12 reads.jf > reads.histogram

Output a graph of k-mer spectrum. k-mer_max recommended value is
1000.

~/Important_Software/genomescope2.0/genomescope.R -i reads.histo -
o output_dir2 -k 21 -p 2

-k kmer length used to calculate kmer spectra [default 21]
-i input histogram file
-o output directory name
-p ploidy (1, 2, 3, 4, 5, or 6) for model to use [default 2]

When running GenomeScope2, if you make a mistake in specifying the
number of ploidy, the estimated value will change.
```

```
If you are not sure about the ploidy and want to estimate the
number of ploidy and whether it is heteroploidy or homoploidy, use
smudgeplot first.

http://kazumaxneo.hatenablog.com/entry/2019/04/18/073000
```

③ De novo genome assembly and quality evaluation

4 The reads were assembled using ALGA (v1.0.3; **Swat *et al.* 2021**) with the default parameter --error-rate = 0.02. long DNA fragments 1 to 10 kb in length were combined, and gaps between them were filled with unknown bases (Ns) using Redundant (v0.14a;**Pryszcz and Gabaldón 2016**), a software program for scaffolding, with default parameter values.

```
#Using ALGA
#How to install #ALGA
(https://kazumaxneo.hatenablog.com/entry/2021/01/22/121538)
#From git.hub
#Depends on.
#CMake VERSION 2.8.7 or higher
#C++ 17 or higher
#Install the latest version of make
#First, check the version of make
make --version
GNU Make 3.82
#Built for x86_64-redhat-linux-gnu.
#Copyright (C) 2010 Free Software Foundation, Inc.
#License GPLv3+: GNU GPL version 3 or later
<http://gnu.org/licenses/gpl.html>
#This is free software: you are free to modify it and redistribute
it freely.
No #warranty to the fullest extent permitted by law.

If your #make version is 4 or lower, update to 4 or higher.
conda install make
make --version
Collect package metadata (current_repodata.json): done
Solution environment: done

## Package plan ##

## Environment location
/home/iceplant4561/anaconda3/envs/gappadder

# Add/update specifications.
# - make

# the following new packages will be installed.

# _libgcc_mutex conda-forge/linux-64::_libgcc_mutex-0.1-
conda_forge
# _openmp_mutex conda-forge/linux-64::_openmp_mutex-4.5-1_gnu
# libgcc-ng conda-forge/linux-64::libgcc-ng-11.2.0-h1d223b6_11
# libgomp conda-forge/linux-64::libgomp-11.2.0-h1d223b6_11
# make conda-forge/linux-64::make-4.3-hd18ef5c_1

#proceed ([y]/n)? y
```

```
#prepare transaction: done
#transaction validation: done
#execute transaction: done

make --version
#GNU Make 4.3
#Built for x86_64-conda-linux-gnu.
#Copyright (C) 1988-2020 Free Software Foundation, Inc.
#License GPLv3+: GNU GPL version 3 or later
<http://gnu.org/licenses/gpl.html>
#This is free software: you are free to modify it and redistribute
it.
No #warranty to the extent permitted by law.

#Update cmake
wget
https://github.com/Kitware/CMake/releases/download/v3.22.1/cmake-
3.22.1.tar.gz
tar zxvf cmake-3.22.1.tar.gz

#Build
cd cmake-3.22.1/
. /bootstrap
build

#pass through the path
echo 'export PATH=$HOME/cmake-3.22.1/bin/:$PATH' >> ~/.bashrc
Source ~/.bashrc

#Check cmake version
cmake --version
#Check cmake version.

#CMake suite is maintained and supported by Kitware
(kitware.com/cmake).

#How to install and build alga using c++ version 17 (see .2022 Jan
11 email from Mr. Ashizawa, National Institute of Genetics).
qlogin
Module load gcc/9.2.0
wget
https://github.com/swacisko/ALGA/archive/refs/tags/1.0.3.tar.gz
tar zxvf 1.0.3.tar.gz
cd ALGA-1.0.3/
```

```
# or
#git clone https://github.com/swacisko/ALGA.git
#cd ALGA/
#either is fine

mkdir build
cd build
cmake -DCMAKE_CXX_COMPILER=/opt/pkg/gcc/9.2.0/bin/c++ \?
-DCMAKE_C_COMPILER=/opt/pkg/gcc/9.2.0/bin/gcc ...

make -j 4
ls
#ALGA CMakeCache.txt CMakeFiles cmake_install.cmake Makefile

#ALGA build is now complete.

cd $HOME/WGS/iceplant_draft_contig

~/ALGA/ALGA --file1=output.0.fastq --file2=output.1.fastq --
threads=10 --output=Mc_draft_genome.fasta --error-rate=0.02

conda activate Redundans
 ~/New_redundans/redundans.py -v \
        -i
/home/iceplant4561/WGS/iceplant_draft_contig/output_1.fastq \

/home/iceplant4561/WGS/iceplant_draft_contig/output_2.fastq \
        -f
/home/iceplant4561/WGS/iceplant_draft_contig/Mc_draft_genome.fasta
-o more_scaffolding
```

The genome coverage of reads was estimated using the Mosdepth program (**Pedersen and Quinlan 2018**).

```
#Genome Coverage Calculations Using mosdepth
#Reference:
https://kazumaxneo.hatenablog.com/entry/2018/06/06/112849
#http://kazumaxneo.hatenablog.com/entry/2018/04/04/175133

#Mapping fastq data to thegenome using minimap2
nohup singularity exec /usr/local/biotools/m/minimap2:2.9--1 \.
minimap2 -t 10 -a -x sr \.
/home/iceplant4561/Agarie_group/Iceplant_shotgun_genome_assembly/m
ore_scaffolding/Mc_2nd_scaffold.filled.fa \?
/home/iceplant4561/Agarie_group/Iceplant_shotgun_genome_assembly/t
rimmed/Mc_musket_1.fastq \.
/home/iceplant4561/Agarie_group/Iceplant_shotgun_genome_assembly/t
rimmed/Mc_musket_2.fastq \ \?
> Mc_Genome.sam &.

#Conversion to bam file => sort
samtools view -@ 40 -bS Mc_Genome.sam > Mc_Genome.bam
samtools sort -@ 40 -o Mc_Genome_sort.bam Mc_Genome.bam
samtools index Mc_Genome_sort.bam #index place

singularity exec /usr/local/biotools/m/m/mosdepth -t 40 -n
Mc_Genome_sort.bam
```

The completeness of the assembled genome was evaluated based on the content of orthologs in higher plants, using the benchmarking universal single-copy orthologs (BUSCO) program (v5.0; **Manni *et al.* 2021**). The lineage dataset was embryophyta_odb10 (creation date: 2020-09-10, number of BUSCOs: 1614).

```
#List creation
singularity exec /usr/local/biotools/b/busco\:5.4.3--pyhdfd78af_0
busco --list-datasets

#genome
singularity exec /usr/local/biotools/b/busco\:5.4.3---pyhdfd78af_0
busco -m geno -i Complete_iceplant_genome.fasta -o out_dir -l
embryophyta_ odb10 -c 30

#Merge multiple busco data
generate_plot.py -wd BUSCO_summaries/
*It is necessary to store the short_summary* file under
BUSCO_summaries beforehand.
In short_summary.specific.embryophyta_odb10.*.txt, the * part is
the species name.
If you use
short_summary.specific.embryophyta_odb10.M.crystallinum.txt, it
will be separated by M. Use M_crystallinum.
*Maybe you can do it by tinkering with python scripts (230208)
```

We also searched for core genes in the genome sequences of nine other plant species: *Kewa caespitosa*, *Pharnaceum exiguum*, *Macarthuria australis*, *Solanum chaucha*, *Populus trichocarpa*, *Arabidopsis thaliana*, and *Oryza sativa* using BUSCO. The first three species belong to the same order, Caryophyllales, to which the ice plants belong. Genome information was obtained from the NCBI (see Supplementary Note 1 "Address to genome information"; **Supplementary_Information: Sato *et al.*, 2022a**). The number of bases, sequences, sequences in several base number ranges, and the maximum base length of the final draft genome sequences was calculated using **gVolante** (v2.0.0)(**Nishimura *et al.* 2017**). BLASTN (v2.2.31+; **McGinnis and Madden 2004**) was used to investigate the number of cDNA sequences identified by transcriptome (**Lim *et al.* 2019**), and registered DNA sequences (retrieved from **NCBI**, last accessed February 2022) were aligned to the final assembled genome sequence.

④ Phylogenetic tree creation among multiple plant species using 18S ribosomal DNA sequences

5 The 18S ribosomal genes were extracted using barrnap (v0.9; **Seemann 2018**) from the obtained genome sequences of the ice plant.

```
barrnap --kingdom euk --threads 12 genome.fasta
#Extract 18 S rDNA sequences from result fasta files.
```

As comparative objectives, 25 kinds of 18S ribosomal genes from general crops (Japanese radish [*Raphanus sativus*], Soybean [*Glycine max*], Japanese trefoil [*Lotus japonicus*], Barrelclover [*Medicago truncatula*], Adzuki bean [*Vigna angularis*], Banana [*Musa acuminata*], Barley [*Hordeum vulgare*], Sorghum [*Sorghum bicolor*], Bread wheat [*Triticum aestivum*], Maize [*Zea mays*], Apple [*Malus domestica*], Peach [*Prunus persica*], Coffee tree (Arabica var.) [*Coffea arabica*], Coffee tree (Robusta var.)[*C. canephora*], Clementine [*Citrus clementina*], Orange [*C. sinensis*], Poplar, Tobacco [*Nicotiana tabacum*], Tomato [*Solanum lycopersicum*], Eggplant [*S. melongena*], Potato [*S. tuberosum*] and Grape [*Vitis vinifera*]) were selected using the **SILVA database** (Release. 2020-08; **Pruesse *et al.* 2007**). After joining all ribosomal DNA sequences into one file, a molecular phylogenetic tree was created using implemented in **NGPhylogeny.fr** (**Lemoine *et al.* 2019**) (Released in 2019). SH-aLRT (Shimodaira-Hasegawa-approximate likelihood ratio test) (**Shimodaira and Hasegawa 1999**) was used to determine the molecular phylogenetic tree.

⑤ Detection of repetitive regions

6 Repetitive sequences were detected, and custom repeat libraries involving transposable elements and long terminal repeat-retro transposons were generated using RepeatModeler2 (v2.0.2; **Flynn *et al.* 2020**) and TEclass (v2.1.3)(**Abrusán *et al.* 2009**). Known repeat sequences were detected and classified in the assembled genome sequence with reference to the Repbase library(**Bao *et al.* 2015**) and the custom repeat libraries, using **RepeatMasker**(v4.1.2-p1; **Smit *et al.* 2013-2015**). The capital letters in the genome sequences were replaced with small characters as soft masking.

```
1) Creation of a repetitive array custom repeat library using
RepeatModeler2

conda create -n repeatmodeler RepeatModeler==2.0.3

conda activate repeatmodeler

BuildDatabase -name Mc
~/Important_Software/Agarie_group/ice_plant_genome/data/iceplant_g
enome.fasta

qsub -V -cwd -l medium -l s_rt=120:00:00 -l d_rt=120:00:00 -l
s_vmem=30G -l mem_req=30G -pe def_slot 30 -b y -e .
/error_log_repeatmodeler -N RepeatModeler \
RepeatModeler \ -database Mc_PacB
-database Mc_PacBio -pa 29 -genomeSampleSizeMax 370000000 \
-repeatmasker_dir ~/Important_Software/RepeatMasker/RepeatMasker \
-abblast_dir ~/Important_Software/ab-blast-20200317-linux-x64/

*Note! It will probably take two days time. Sleep at home.

Mc-families.fa and Mc-families.stk (Stockholm format) will appear
*Name specified with -name during BuilDatabase

2)TEclass is used to classify TEs classified by RepeatModeler in
detail.

singularity exec ~/Important_Software/teclass.sif TEclassTest.pl -
c TEclass-2.1.3c/classifiers -o ./ Mc-families.fa
cd _*
cp Mc-families.fa.lib ./
~/Agarie_group/ice_plant_genome/Repeat/RepeatMasker
cd ~/Agarie_group/ice_plant_genome/RepeatmaskerLib/

3)RepeatMasker
makeblastdb -in Mc-families.fa.lib -dbtype nucl -blastdb_version 4
*If you don't do it first, an error will occur.
RepeatMasker -pa 20 -html -gff -xsmall -lib Mc-families.fa.lib
Mc_scaffold.filled.fa
```

⑥ Search for genomic sequences coding transfer RNA (tRNA) and micro-RNA (miRNA)

7 The tRNA genes were identified in the draft common ice plant genome using tRNAscan-SE2.0 (v2.0.9)(**Chan *et al.* 2021**).

```
#Identification of tRNAs using tRNAscanSE-2.0
#Reference:https://kazumaxneo.hatenablog.com/entry/2019/05/07/073000

singularity exec /usr/local/biotools/t/trnascan-se\:2.0.9--pl5321hec16e2b_3 \
tRNAscan-SE \
~/Agarie_group/ice_plant_genome/data/iceplant_genome.fasta \
-E -o Mc_tRNA_output -f tRNA_structure -s isotype -m statistics -b bedfiles -j gff -a fastafile -l worklog --detail --thread 30
```

The tRNA data of other nine plant species—*Arabidopsis*, rice, tomato, poplar, horseradish, potato, grape, soybean, and coffee tree (robusta species)—were obtained from the PlantRNA database (**Cognat *et al.* 2013**). The percentages of arbitrary tRNAs against the total tRNAs in the genome were calculated and compared to the ice plants' values with those of the other species. Smirnov-Grubbs' outlier tests were performed to select tRNAs more significantly involved. The test statistic T was calculated using the following equation:

$$T = \frac{(Percentage\ of\ arbitrary\ tRNAs\ in\ the\ ice\ plant) - (Sample\ mean\ for\ all\ nine\ species)}{\sqrt{Sample\ variance}}$$

The miRNA loci in the genome sequence were identified using the cmscan command in infernal (v1.1.4; **Nawrocki and Eddy 2013**) using **Rfam**.

```
#Identification of small RNA using Infernal
#Reference: http://eddylab.org/infernal/Userguide.pdf
#http://http.ebi.ac.uk/pub/databases/Rfam/

#In this case, we will use "Searching the Rfam CM database with a
query sequence" on p. 29 #of the reference.

#Infernal uses the Singularity image file from the Institute of
Genetics.

#Use Covariant Model (CM). I'm not sure, but I'll try to do it as
written.

(1) Get the latest covariance model from Rfam
wget http://http.ebi.ac.uk/pub/databases/Rfam/14.8/Rfam.cm.gz
gunzip Rfam.cm.gz

(2) Get Rfam's clan information (like a family)
wget http://http.ebi.ac.uk/pub/databases/Rfam/14.8/Rfam.clanin
mv Rfam.clanin Rfam.14.8.clanin

(3) Make data available in cmpress
singularity exec /usr/local/biotools/i/infernal\:1.1.4--
pl5321hec16e2b_1 \c
cmpress Rfam.cm

(4) Search with cmscan. Do as written.
singularity exec /usr/local/biotools/i/infernal\:1.1.4--
pl5321hec16e2b_1 \cmpress
cmscan --rfam --cut_ga --nohmmonly --tblout Mc_genome.tblout --fmt
2 --clanin Rfam.14.8.clanin --cpu 30 \
Rfam.cm ~/Agarie_group/ice_plant_genome/data/iceplant_genome.fasta
> Mc_genome.cmscan
```

⑦ Gene prediction

8 The BRAKER2 pipeline (v2.1.5; **Brůna *et al.* 2021**) was used for the prediction of genes in the common ice plant genome. Amino acid sequences were translated from the transcriptome profile reported by **Lim *et al.* (2019)** and used as additional reference data for the prediction of genes. BRAKER2 was used with the default parameters (–softmasking).

#Annotation of genomes using BRAKER2
At this point, change the header of the fasta file of the masked genome.
In the bam2hints process of BRAKER2, if there is a whitespace (" " ← this) in the fasta header, the error message "The hints file is empty.
The hints file is empty. Maybe the genome and the RNA-seq file do not belong together" error occurs.
Reference: https://github.com/Gaius-Augustus/BRAKER#common-problems

How to change
Create a new line

```
cat Iceplant-genome_fasta_full_softmask.fasta | awk '/^>/ { print n $0; n = "" }! /^>/ { printf "%s", $0; n = "\n" } END{ printf "%s", n }' > A.fasta

mv A.fasta Iceplant-genome_fasta_full_softmask.fasta
```

Preparation for BRAKER below
Reference: https://qiita.com/drk0311/items/a3ac648f2780cfee57b1

Obtaining GeneMark-ES/ET/EP
http://exon.gatech.edu/GeneMark/license_download.cgi

Here, add your name and affiliation, and
GeneMark-ES/ET/EP ver 4.69_lic
        LINUX 64 kernel 2.6 - 3
and click on I agree to the terms of this license agreement to go to the next page.

Right-click here and click Download

Download the key as well

Transfer to linux via sftp

```
sftp iceplant4561@gw2.ddbj.nig.ac.jp
cd /home/iceplant4561/Important_Software

put gmes_linux_64.tar.gz
put gm_key_64.gz
```

```
Back to linux
cd /home/iceplant4561/Important_Software

Extract each
tar xzvf gmes_linux_64.tar.gz
gunzip gm_key_64.gz

#The license is valid for 200 days, so after 200 days, go back to
the above site, re-enter your registration information, agree to
the license, and then click on the "Download key 32_bit or 64_bit"
button.
#Please download the license key (gm_key_64.gz or gm_key_32.gz)
from the link "Please download key 32_bit or 64_bit".
#If you are using 64 bit now, the majority of users will probably
use 64 bit. Unzip the license key with gunzip, rename it to
.gm_key and save it in your home directory.
#Now, you can save the program anywhere you want, but I keep my
tools that cannot be managed by Anaconda in a directory named
"local" under my home directory (~ or /home/account name) and put
them there.
#local directory under your home directory (~ or /home/account
name) for tools that cannot be managed by Anaconda. The following
is a case of dropping the program files into the downloads folder
on Windows.

cp gm_key_64 ~/.gm_key
cd
/home/iceplant4561/Agarie_group/ice_plant_genome_from_GSA/BRAKER/g
mes_linux_64
. /check_install.bash
Checking GeneMark-ES installation

export
GENEMARK_PATH=/home/iceplant4561/Agarie_group/ice_plant_genome_fro
m_GSA/BRAKER/gmes_linux_64
source ~/.bashrc

Create a Docker image container for BRAKER (v2.1.5)

cd ~/Important_Software/
singularity build braker.sif docker://hamiltonjp/braker2:a765b80

cd ~/Agarie_group/ice_plant_genome/BRAKER/

species="M.crystallinum"
```

```
species_dir="${PWD}/${species}"

singularity exec /home/iceplant4561/Important_Software/braker.sif
\
braker.pl --genome=./Iceplant-genome_fasta_full_softmask.fasta \
            --species=${species}_braker2 \
            --workingdir=./braker2_out \
          --softmasking \
          --prot_seq=Proteins_from_iceplants.fasta \
          --gff3 \
          --epmode \
          --cores 45 \
          --
GENEMARK_PATH=~/Agarie_group/ice_plant_genome_from_GSA/BRAKER/gmes
_linux_64 \
            --
AUGUSTUS_CONFIG_PATH=~/Agarie_group/ice_plant_genome_from_GSA/BRAK
ER/Augustus/config/ \
          --useexisting
```

The total sequences, total bases, total amino acids, and N50 were computed based on the resulting fasta-format files containing information about the genes, coding sequences, and amino acids using seqkit (v2.0.0; **Shen *et al.* 2016**) [key parameter: seqkit stats]. Protein BLAST searches (*E*-value < 1e-5) were conducted using DIAMOND (v2.0.13.151; **Buchfink *et al.* 2021**) against the **NCBI**-non-redundant protein sequences (retrieved from **NCBI** in March 2022), **Uniprot-swissprot**(retrieved in March 18), **Ensemble TAIR10** (retrieved in March 2022), and NCBI poplar amino acid sequence databases(retrieved from **NCBI** in March 2022).

```
#Output statistics for FASTA files using seqkit stats
seqkit stats -a *.fa

#BLASTP using DIAMOND(ex. NCBI)
singularity exec /usr/local/biotools/d/diamond:2.0.9--hdcc8f71_0
diamond makedb --in nr --db nr

singularity exec /usr/local/biotools/d/diamond:2.0.9--hdcc8f71_0
diamond blastp --query protein_output.fa \
--db ~/blast/database_for_blast/nr.dmnd --max-target-seqs 1 \
--evalue 1e-5 --outfmt 6 --out blast_vs_ncbi.txt -b12 -c1 --
threads 30
```

⑧ Protein domain searches

9 The protein domains in the genome were identified using the Pfam (v33.1) database(**Mistry *et al.* 2021**) with *E*-value < 1e-3, using HMMER (v3.1b2; **Potter *et al.* 2018**).

```
hmmpress ~/Pfam_db/Pfam.hmm
hmmscan --domtblout Pfam_result.out -E 1e-3 --cpu 20 \
~/Pfam_db/Pfam.hmm Protein_braker.fasta

#hmmscan can't be used because Pfam.hmm files are big data.
#https://www.biostars.org/p/438243/
```

The protein databases of rice, maize, and poplar from the **NCBI** (last accessed February 2022) were used in the domain for a detailed classification of the PKinase family, the iTAK (v18.12) web tool (**Zheng et al. 2016**; last accessed February 2022) was utilized. The ratio of families with a high ratio of genes to total genes in the ice plant was compared with that of the same families in the other plants. For statistical analysis, we used Smirnov-Grubbs' outlier tests. The following equation was used to obtain the test statistic T:

$$T = \frac{(Percentage\ of\ arbitrary\ protein\ families\ in\ the\ ice\ plant) - (Sample\ mean\ for\ all\ nine\ species)}{\sqrt{Sample\ variance}}$$

Finally, BLASTP was used to compare proteins generated from the ice plant genome and those from *Arabidopsis*, rice, maize, and poplar and renamed TAIR10 ID. These IDs were subjected to gene ontology (GO) enrichment analysis using DAVID (updated in 2022; accessed on March 24; **Sherman *et al.* 2022**) based on a modified Fisher exact probability test with *E*-value < 0.05.

Protocol references

Plant materials and growth conditions described in MATERIALS

Agarie S, Kawaguchi A, Kodera A, Sunagawa H, Kojima H, Nose A, Nakahara T. 2009. Potential of the common ice plant, Mesembryanthemum crystallinum as a new high-functional food as evaluated by polyol accumulation. Plant Prod Sci. 12(1):37–46. doi:10.1626/pps.12.37.

Gamborg, OL., Miller, RA., Ojima, K. 1968. Nutrient requirements of suspension cultures of soybean root cells. *Exp. Cell Res*. 50(1): 151–158. https://doi.org/10.1016/0014-4827(68)90403-5

Winter K, Ltittge U, Winter E, Troughton JH. 1978. Seasonal shift from C3 photosynthesis to crassulacean acid metabolism in *Mesembryanthemum crystallinum* growing in its natural environment. *Oecologia (Berl)*. 34:225–237. https://doi.org/10.1007/BF00345168

②Clean read preparation and genome size estimation

Hunt M, Kikuchi T, Sanders M, Newbold C, Berriman M, Otto TD. 2013. REAPR: a universal tool for genome assembly evaluation. *Genome Biol.* 14(5):R47. doi:10.1186/gb-2013-14-5-r47.

Liu Y, Schröder J, Schmidt B. 2013. Musket: a multistage *k*-mer spectrum-based error corrector for Illumina sequence data. *Bioinformatics.* 29(3):308–315. doi:10.1093/bioinformatics/bts690.

Marçais G, Kingsford C. 2011. A fast, lock-free approach for efficient parallel counting of occurrences of *k*-mers. *Bioinformatics.*27(6):764–770. doi:10.1093/bioinformatics/btr011.

Ranallo-Benavidez TR, Jaron KS, Schatz MC. 2020. GenomeScope 2.0 and Smudgeplot for reference- free profiling of polyploid genomes. *Nat Commun.* 11:1432.

Simon A. 2010. FastQC: a quality control tool for high throughput sequence data. http://www.bioinformatics.babraham.ac.uk/projects/fastqc

③ De novo genome assembly and quality evaluation

Lim SD, Lee S, Choi WG, Yim WC, Cushman JC. 2019. Laying the foundation for crassulacean acid metabolism (CAM) biodesign: expression of the C4 metabolism cycle genes of CAM in *Arabidopsis*. *Front Plant Sci.* 10:101. doi:10.3389/fpls.2019.00101.

Manni M, Berkeley MR, Seppey M, Zdobnov EM. 2021. BUSCO: assessing genomic data quality and beyond. *Curr Protoc.*1(12):e323. doi:10.1002/cpz1.323.

McGinnis S, Madden TL. 2004. BLAST: at the core of a powerful and diverse set of sequence analysis tools. *Nucleic Acids Res.* 32:20–25. doi:10.1093/nar/gkh435.

Nishimura O, Hara Y, Kuraku S. 2017. GVolante for standardizing completeness assessment of genome and transcriptome assemblies. *Bioinformatics.* 33(22):3635–3637. doi:10.1093/bioinformatics/btx445.

Pedersen BS, Quinlan AR. 2018. Mosdepth: Quick coverage calculation for genomes and exomes. *Bioinformatics.*34(5):867–868. doi:10.1093/bioinformatics/btx699.

Pryszcz LP, Gabaldón T. 2016. Redundans: an assembly pipeline for highly heterozygous genomes. *Nucleic Acids Res.*44(12):e113. doi:10.1093/nar/gkw294.

Sato, R. Kondo, Y. Agarie, S. (2022) Supplementary_Information. figshare. Dataset. https://doi.org/10.6084/m9.figshare.21788624.v5 Swat S, Laskowski A, Badura J, Frohmberg W, Wojciechowski P, Swiercz A, Kasprzak M, Blazewicz J. 2021. Genome-scale de novo assembly using ALGA. *Bioinformatics.*37(12):1644–1651. doi:10.1093/bioinformatics/btab005.

④ Phylogenetic tree creation among multiple plant species using 18S ribosomal DNA
Lemoine F, Correia D, Lefort V, Doppelt-Azeroual O, Mareuil F, Cohen-Boulakia S, Gascuel O. 2019. NGPhylogeny.fr: new generation phylogenetic services for non-specialists. *Nucleic Acids Res.*47(W1):W260-W265. doi: 10.1093/nar/gkz303.
Pruesse E, Quast C, Knittel K, Fuchs BM, Ludwig W, Peplies J, Glöckner FO. 2007. SILVA: a comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB. *Nucleic Acids Res.* 35(21):7188–7196. doi:10.1093/nar/gkm864
Seemann T. 2018. barrnap 0.9: rapid ribosomal RNA prediction. https://github.com/tseemann/barrnap
Shimodaira H, Hasegawa M. 1999. Multiple comparisons of loglikelihoods with applications to phylogenetic inference. *Mol BiolEvol.* 16:1114–1116.

⑤ Detection of repetitive regions
Abrusán, G., Grundmann, N., DeMester, L., & Makalowski, W. (2009). TEclass--a tool for automated classification of unknown eukaryotic transposable elements. Bioinformatics (Oxford, England), 25(10), 1329–1330. https://doi.org/10.1093/bioinformatics/btp084
Bao W, Kojima KK, Kohany O. 2015. Repbase Update, a database of repetitive elements in eukaryotic genomes. *Mob DNA*. 6(1):4–9. doi:10.1186/s13100-015-0041-9.
Flynn JM, Hubley R, Goubert C, Rosen J, Clark AG, Feschotte C, Smit AF. 2020. RepeatModeler2 for automated genomic discovery of transposable element families. *Proc Natl Acad Sci U S A.*117(17):9451–9457. doi:10.1073/pnas.1921046117.
Smit, AFA, Hubley, R, Green, P. RepeatMasker Open-4.0. 2013-2015 <http://www.repeatmasker.org>

⑥ Search for genomic sequences coding transfer RNA (tRNA) and micro-RNA (miRNA)
Chan PP, Lin BY, Mak AJ, Lowe TM. 2021. TRNAscan-SE 2.0: improved detection and functional classification of transfer RNA genes. *Nucleic Acids Res.* 49(16):9077–9096. doi:10.1093/nar/gkab688.
Cognat V, Pawlak G, Duchêne AM, Daujat M, Gigant A, Salinas T, Michaud M, Gutmann B, Giegé P, Gobert A, *et al.* 2013. PlantRNA, a database for tRNAs of photosynthetic eukaryotes. *Nucleic Acids Res.* 41(D1):273–279. doi:10.1093/nar/gks935.
Nawrocki EP, Eddy SR. 2013. Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics.*29(22):2933–2935. doi:10.1093/bioinformatics/btt509.

⑦ Gene prediction
Brůna T, Hoff KJ, Lomsadze A, Stanke M, Borodovsky M. 2021. BRAKER2: automatic eukaryotic genome annotation with GeneMark-EP+ and AUGUSTUS supported by a protein database. *NAR Genom Bioinform.* 3(1):1–11. doi:10.1093/nargab/lqaa108.
Buchfink B, Reuter K, Drost HG. 2021. Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nat Methods.*18(4):366–368. doi:10.1038/s41592-021-01101-x
Shen W, Le S, Li Y, Hu F. 2016. SeqKit: a cross-platform and ultrafast toolkit for FASTA/Q file manipulation.*PLoS One.*11(10):e0163962. doi:10.1371/journal.pone.0163962.
Lim SD, Lee S, Choi WG, Yim WC, Cushman JC. 2019. Laying the foundation for crassulacean acid metabolism (CAM) biodesign: expression of the C4 metabolism cycle genes of CAM in *Arabidopsis*. *Front Plant Sci.* 10:101. doi:10.3389/fpls.2019.00101.

⑧ Protein domain searches

Mistry J, Chuguransky S, Williams L, Qureshi M, Salazar GA, Sonnhammer ELL, Tosatto SCE, Paladin L, Raj S, Richardson LJ, *et al.* 2021. Pfam: The protein families database in 2021. *Nucleic Acids Res.* 49(D1):D412–D419. doi:10.1093/nar/gkaa913.

Potter SC, Luciani A, Eddy SR, Park Y, Lopez R, Finn RD. 2018. HMMER web server: 2018 update. *Nucleic Acids Res.*46(W1):W200–W204. doi:10.1093/nar/gky448.

Zheng, Y, Jiao, C, Sun, H, Rosli, HG, Pombo, MA, Zhang, P, Banf, M, Dai, X, Martin, GB, Giovannoni, JJ, Zhao, PX, Rhee, SY, Fei, Z. 2016. iTAK: A Program for genome-wide prediction and classification of plant transcription factors, transcriptional regulators, and protein kinases. *Mol. Plant. 9*(12): 1667–1670. https://doi.org/10.1016/j.molp.2016.09.014