

# Hapl-o-Mat - Getting Started - Windows

Please also see the README.

## Hapl-o-Mat

Hapl-o-Mat is software for HLA haplotype inference coded in C++. Besides estimating haplotype frequencies via an expectation-maximization algorithm, it is capable of processing HLA genotype population data. This includes translation of alleles between various typing resolutions and resolving allelic and genotypic ambiguities. Both common formats for recording HLA genotypes, multiple allele (NMDP) codes and genotype list strings, are supported.

This guide explains how to use Hapl-o-Mat under Windows. For more information refer to our publications on Hapl-o-Mat:

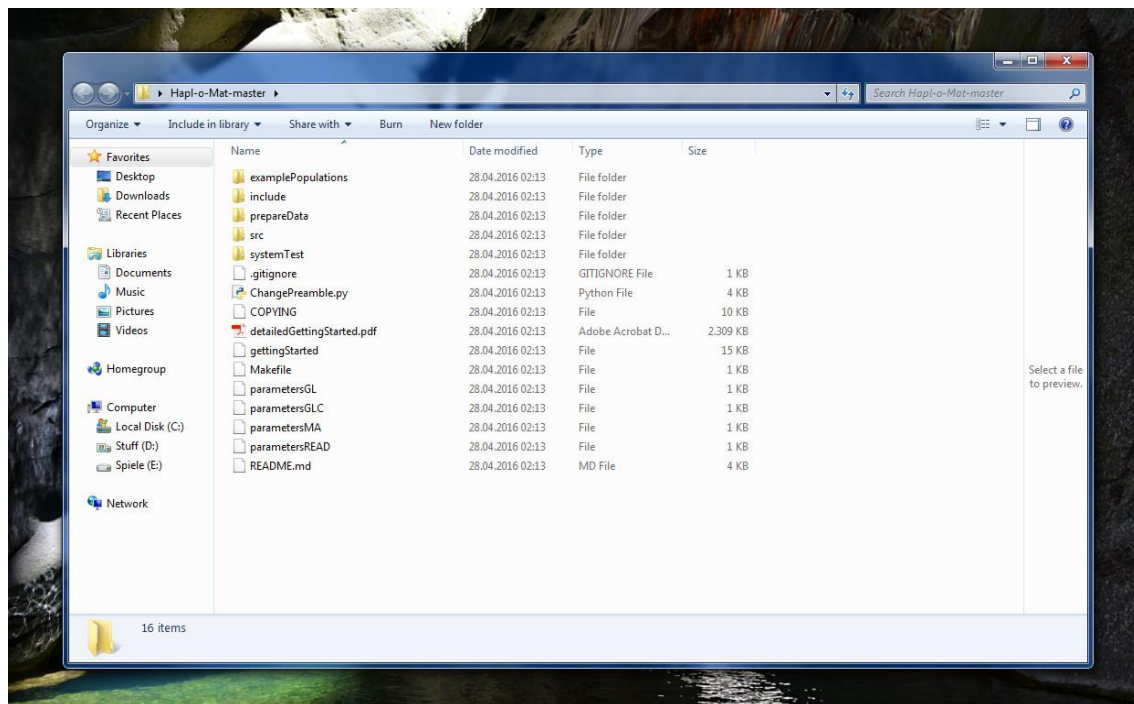
Journal article to come

C. Schaefer, A.H. Schmidt, J. Sauter: Hapl-O-mat: A Versatile Software for Haplotype Frequency Estimation. HLA (2016), 87, 236-320

## Getting Started

This guide is an introduction on how to use Hapl-o-Mat. In order to follow this guide, you need a Windows system and a C++ compiler supporting C++11. In this tutorial, we use Eclipse IDE for C/C++ Developers. You can get it here <https://www.eclipse.org/downloads/>.

After successfully downloading Hapl-o-Mat, look into the folder. You should see the following:



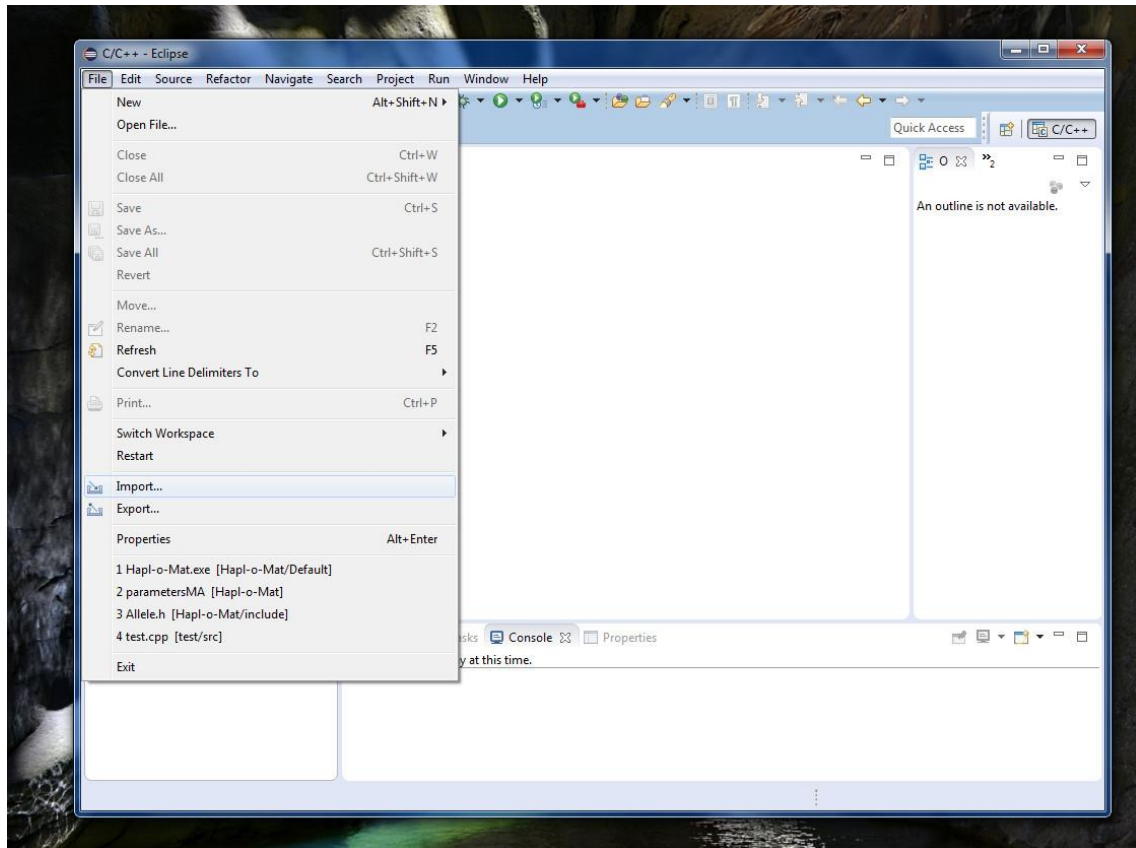
We give you some information on the files you find. Files which are important for using Hapl-o-Mat are marked as bold:

File name	Description
ChangePreamble.py	A python script to adapt the preamble in all files. You are not going to use it.
COPYING	The GNU General Public License.
detailedGettingStarted	Guides for using Hapl-o-Mat under Windows and Linux
<b>examplePopulations</b>	Some genotype population data we are going to work with in the section Tutorials.
gettingStarted	A shorter form of this tutorial
include	A part of Hapl-o-Mat's source code. If you do not want to change code, do not touch it
Makefile	Instructions for building Hapl-o-Mat.
<b>parametersGL, parametersGLC, parametersMA, parametersMA</b>	Parameter files for Hapl-O-mat. We are going to discuss this in section Parameters
<b>prepareData</b>	Here is everything to create the data required by Hapl-o-Mat
<b>README</b>	Read me
src	A part of Hapl-o-Mat's source code. If you do not want to change code, do not touch it
systemTest	Run the system test after changing code to check, if you broke something.

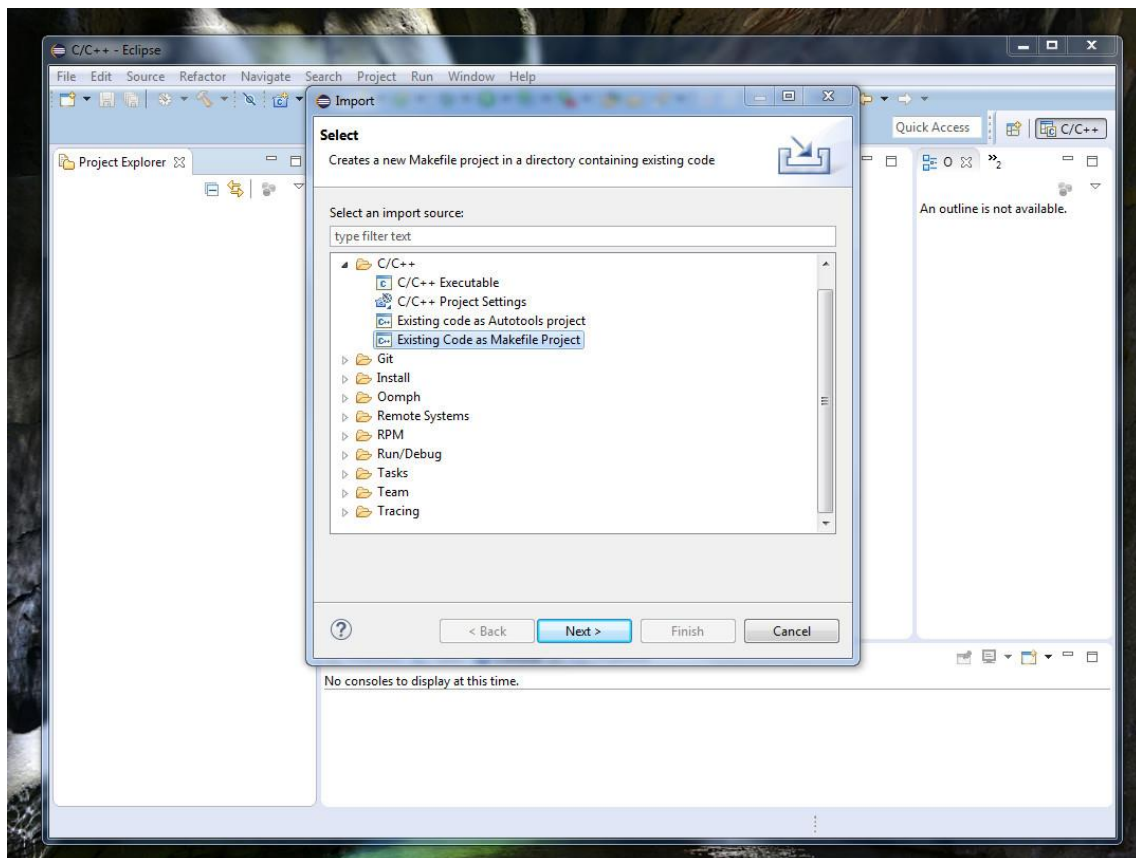
To estimate haplotype frequencies we only need to consider the folder prepareData and the files parametersGL, parametersGLC, parametersMA, and parametersREAD. To finish this tutorial we need the folder examplePopulations, too.

## Install Hapl-o-Mat

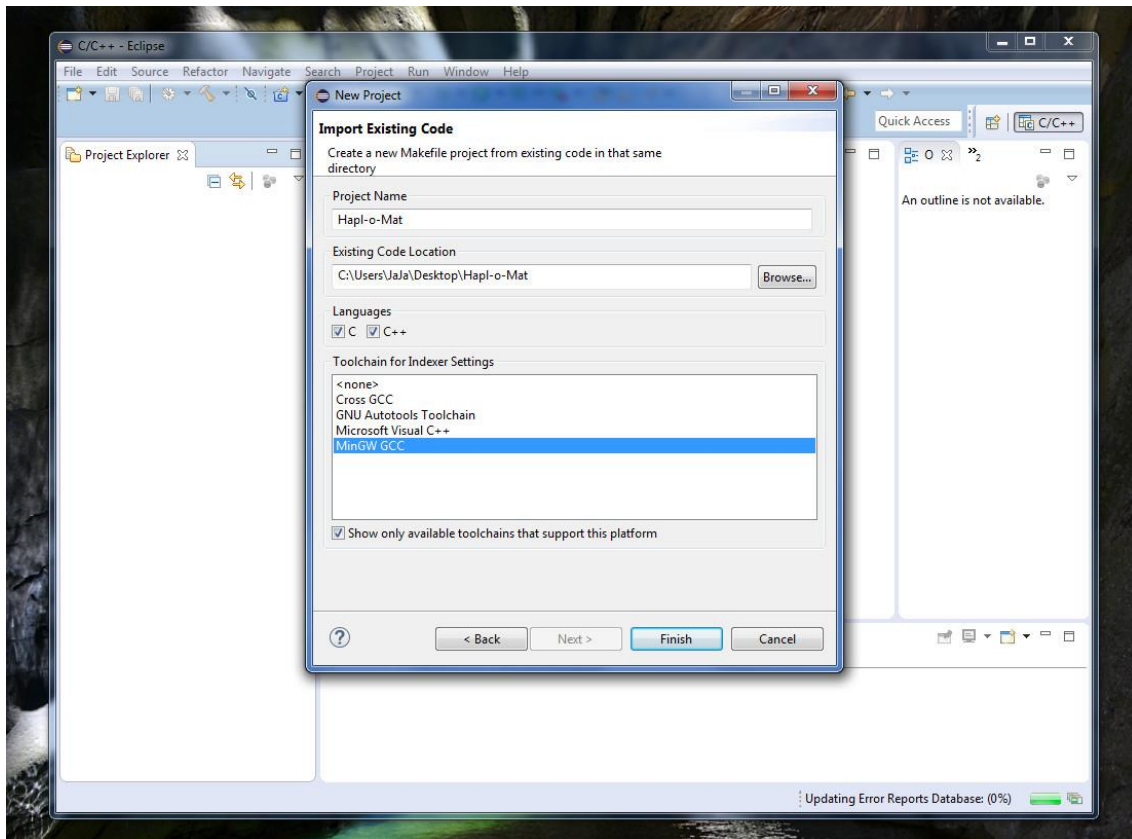
We compile Hapl-o-Mat using Eclipse. Start Eclipse and create a new project by clicking on File -> Import.



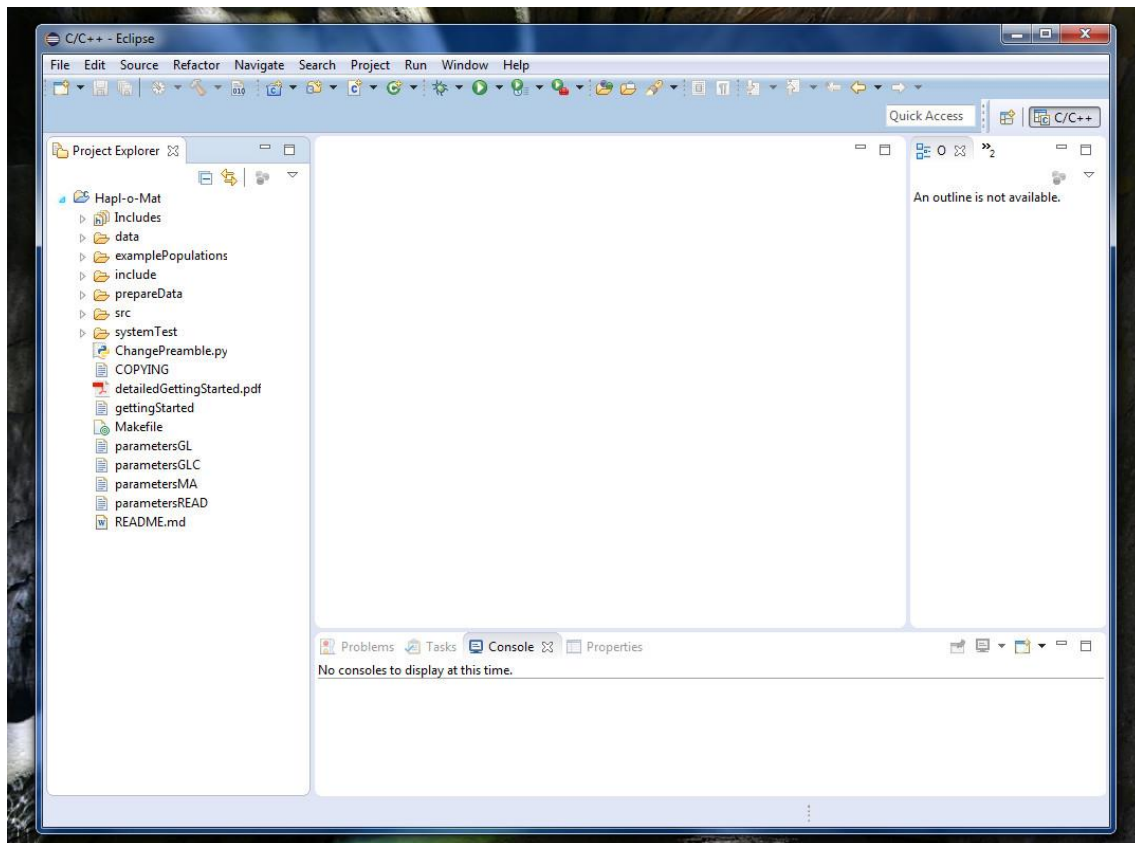
Choose Existing Code as Makefile Project in the pop-up window and then click Next.



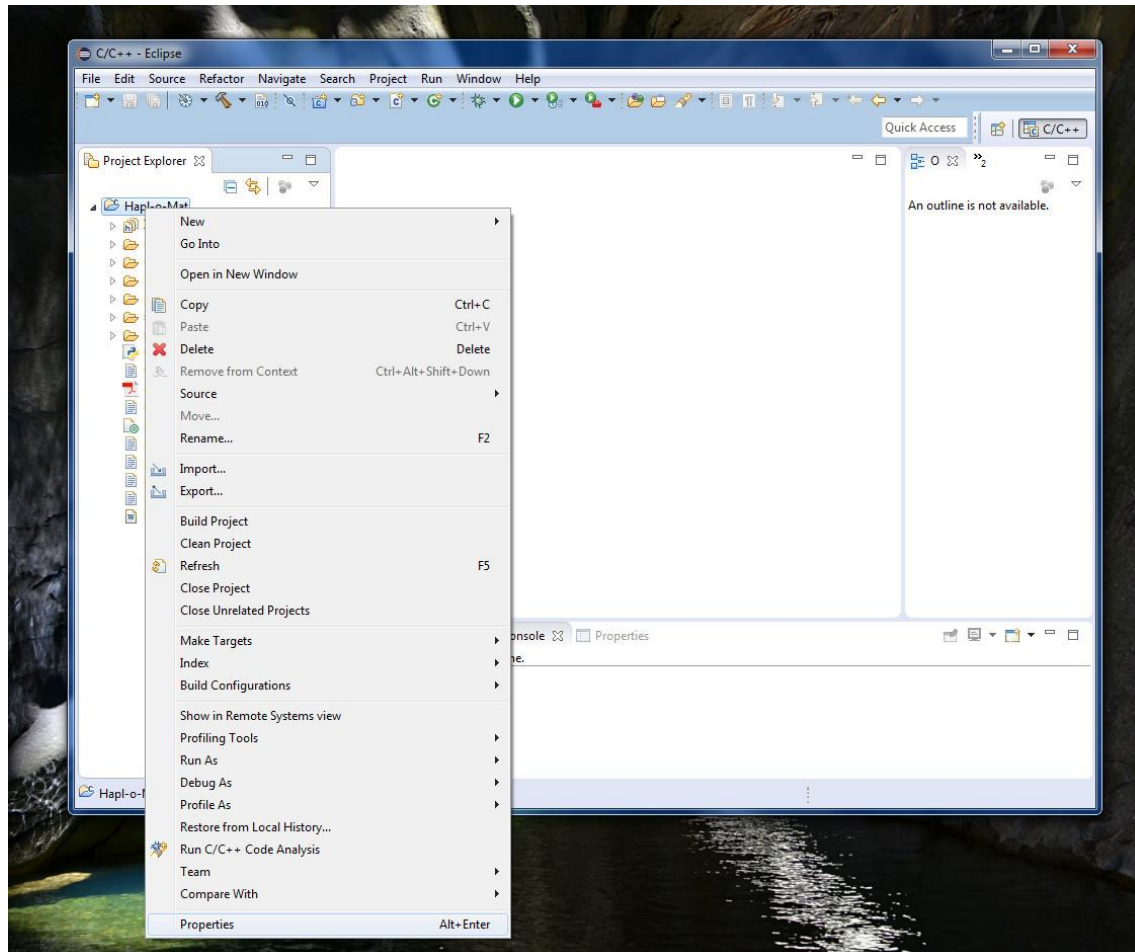
Enter the Project Name, e.g. Hapl-o-Mat, and browse to the location where you saved Hapl-o-Mat. Select MinGW GCC as Toolchain. Click Finish.



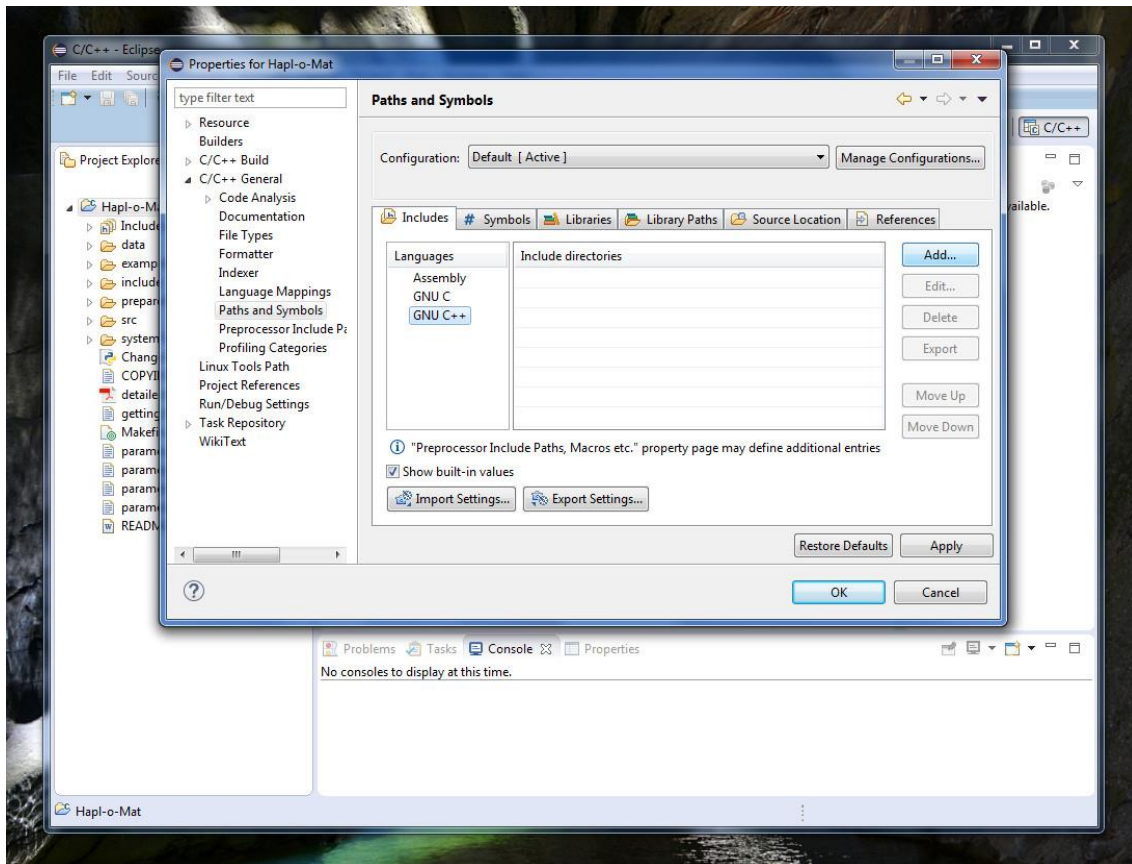
Now, you find all files and folders of Hapl-o-Mat in the Project Explorer. If your Eclipse does not show the Project Explorer, activate it via Window -> Show View -> Project Explorer.



Next, we have to adapt some settings. Right-click into the Project Explorer and choose Properties.

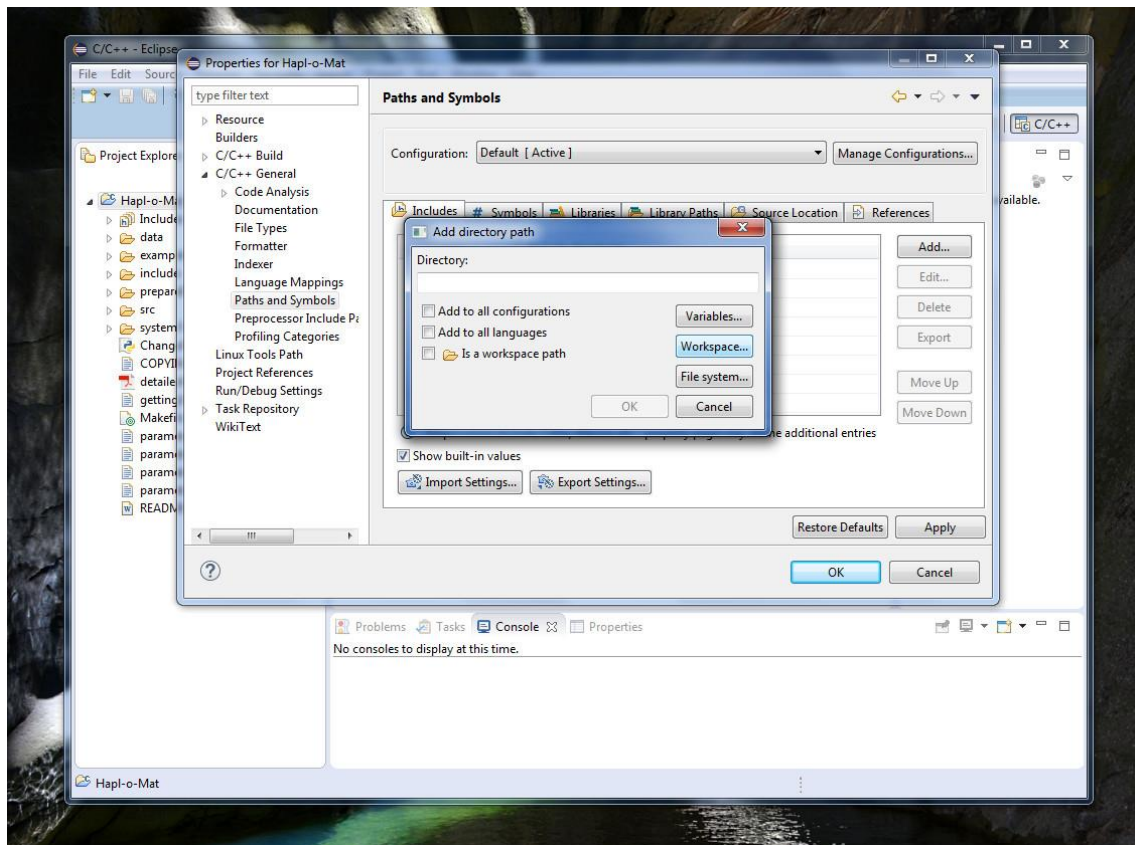


A new window pops up. In the left column choose C/C++ General -> Paths and Symbols. Then, click Add in the right column.

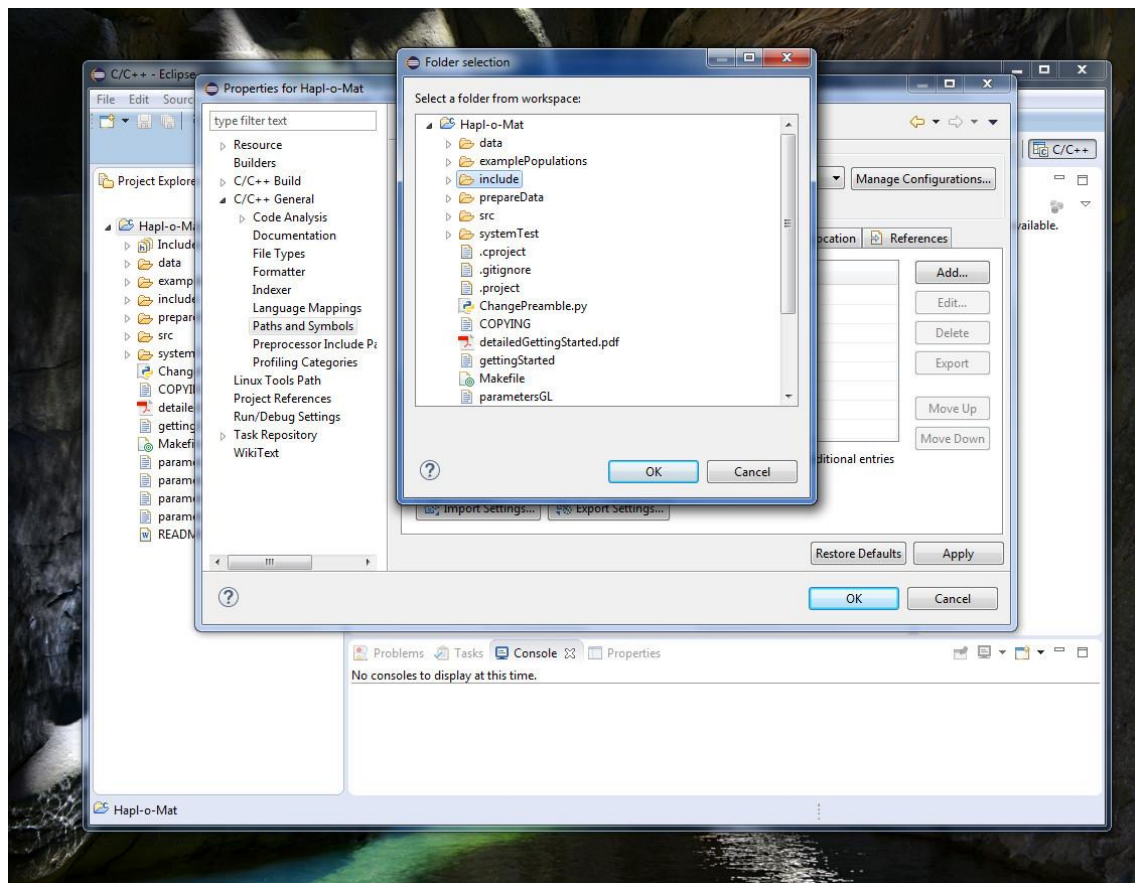


A new window pops up, where you click on Workspace....

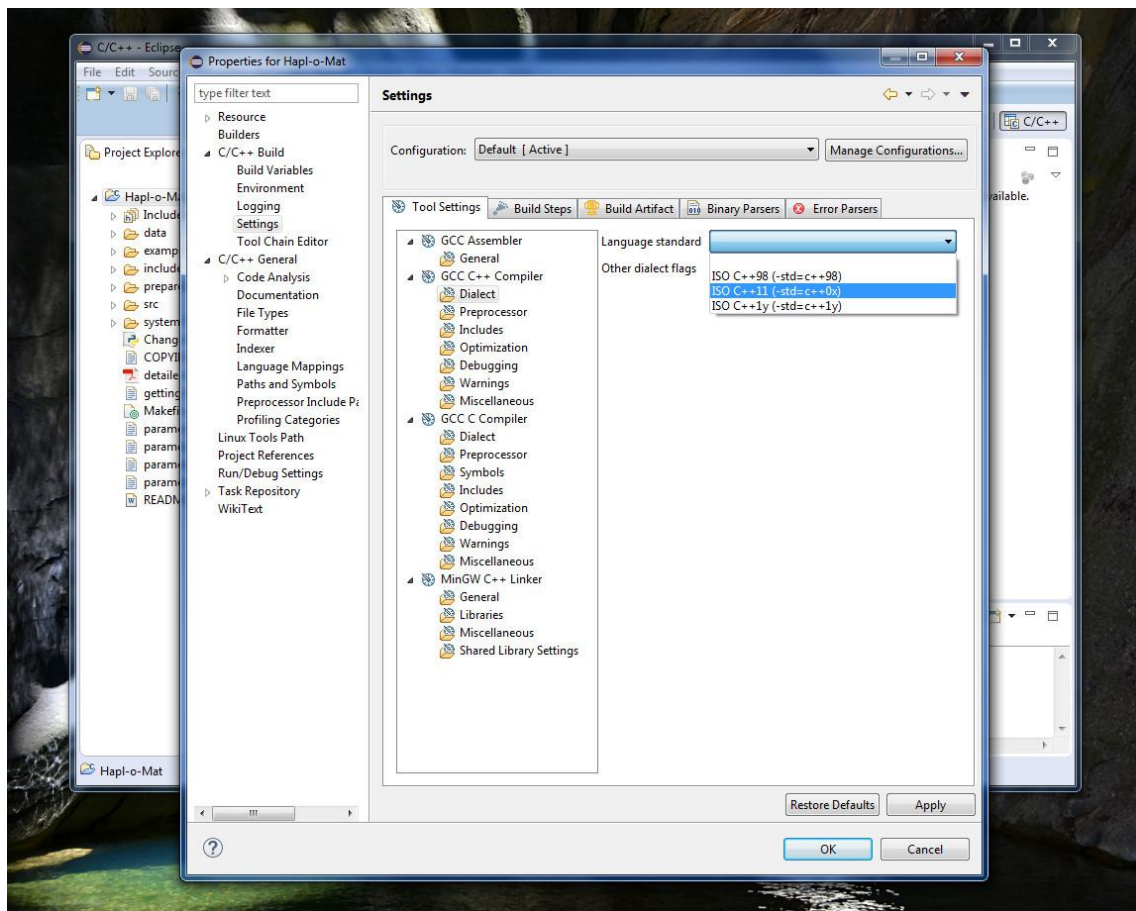




In the following window choose folder include.

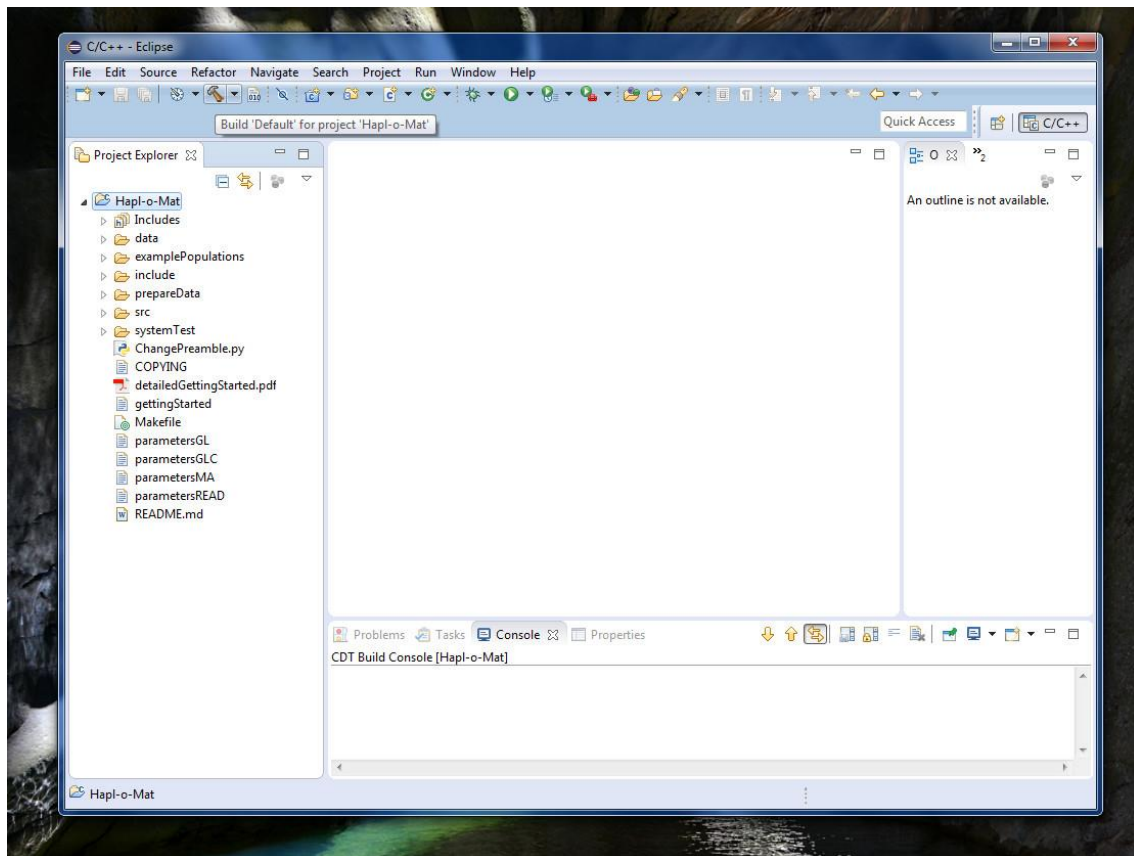


Click two times OK and then choose in the left column C/C++ Build -> Settings. In the right column choose GCC C/C++ Compiler -> Dialect and then choose ISO C++11(-std=c++0x) as Language standard. Click two times OK.

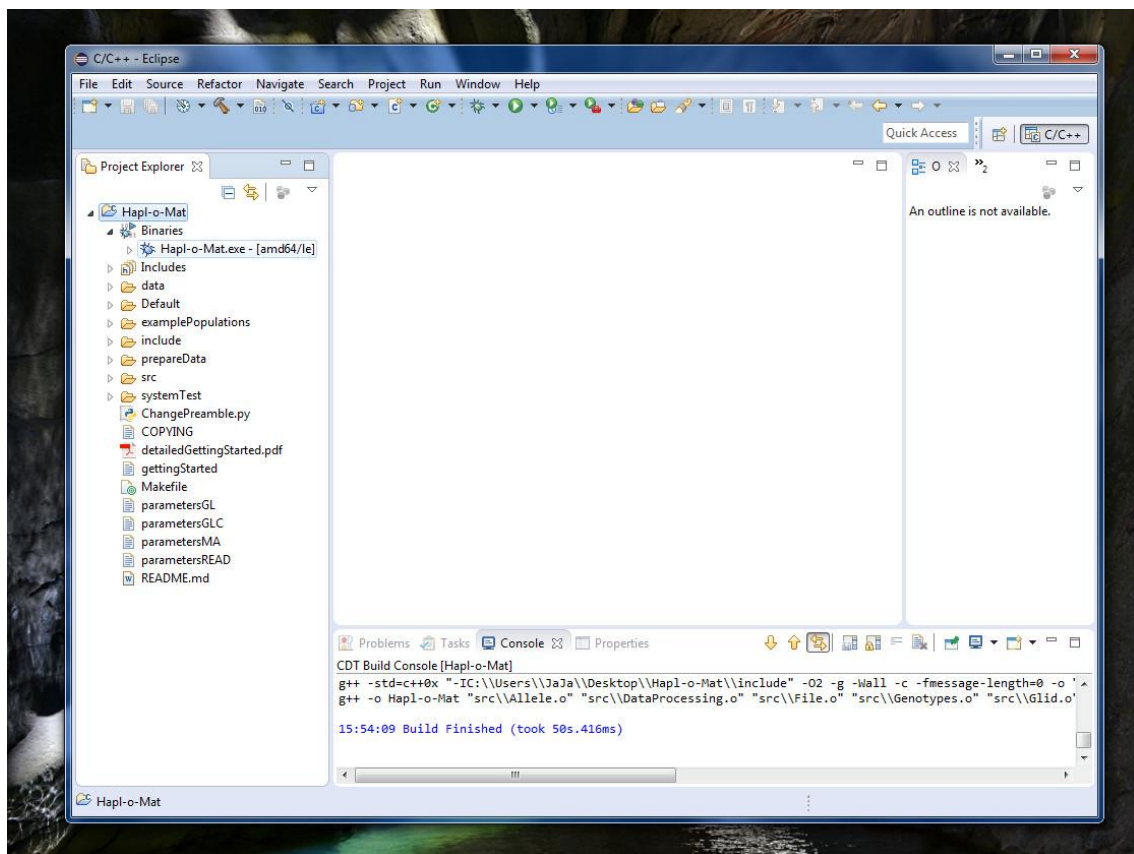


Finally, we compile Hapl-o-Mat by clicking the hammer-symbol in the upper navigation bar.

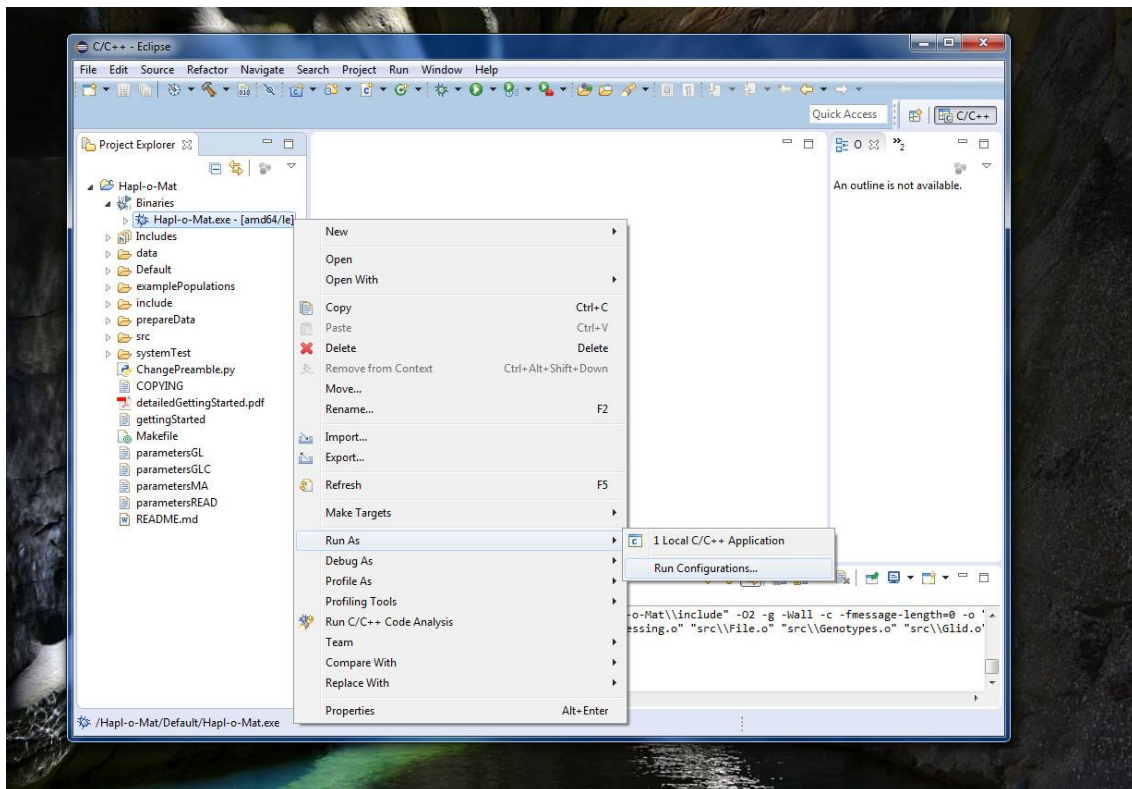




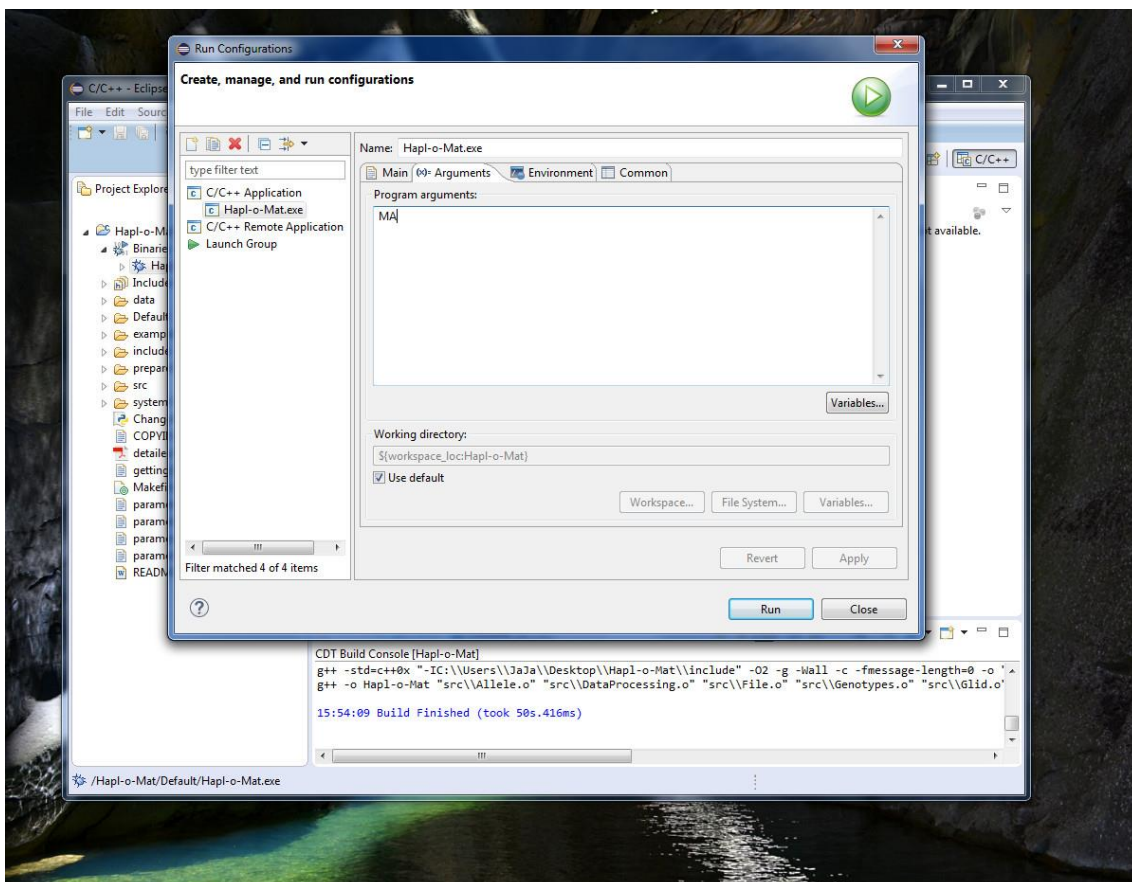
The executable Hapl-o-Mat.exe appeared.



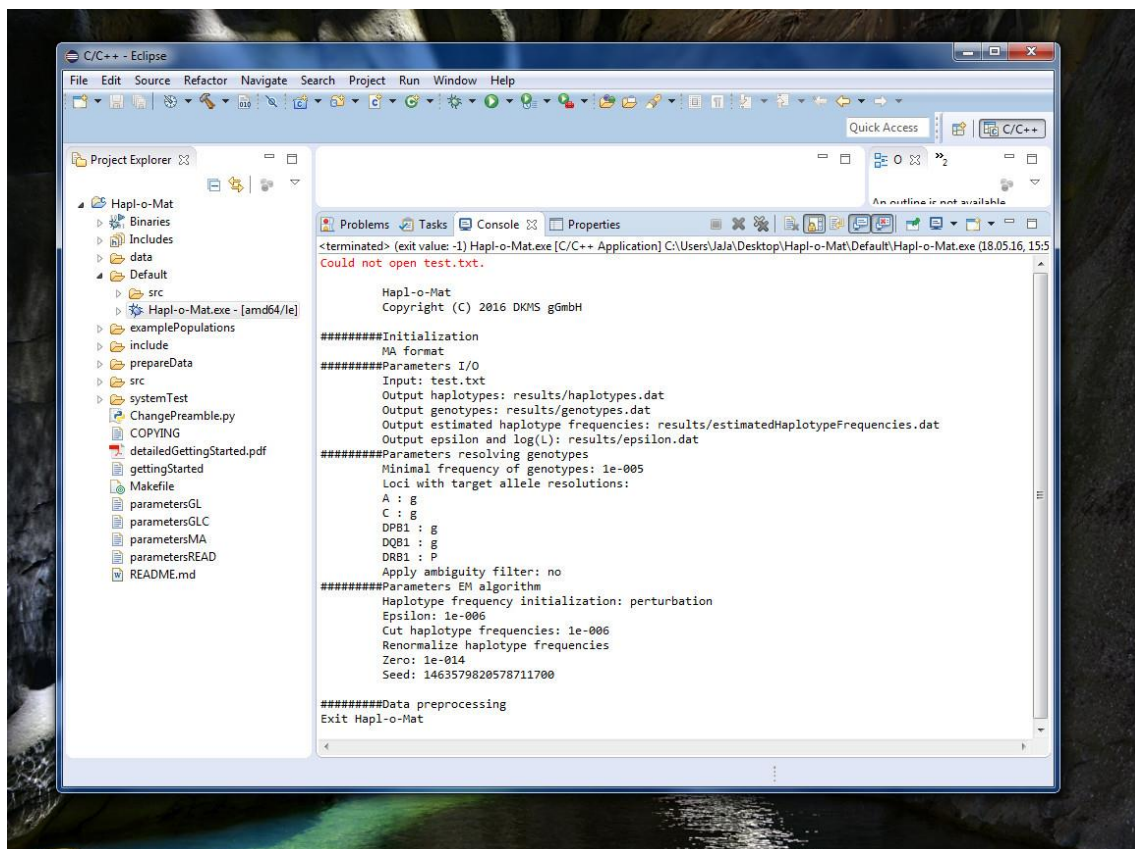
Right-click on it and choose Run As -> Run Configurations....



Click on the tab Arguments and enter MA into the field Program arguments.



Then click Run. Hapl-o-Mat runs now, but produces an error message, since we did not specify any input files. We tackle this later in the tutorial.



## Data Preparation

Hapl-o-Mat relies on information on the HLA nomenclature. This information is provided by data files, which we are going to create. As the HLA nomenclature evolves over time, e.g. by finding new alleles or adding new NMDP codes, it is important to update data from time to time. Hapl-o-Mat relies on the following files, which must be placed in the folder “Hapl-o-Mat/data” (we create the folder “data” later):

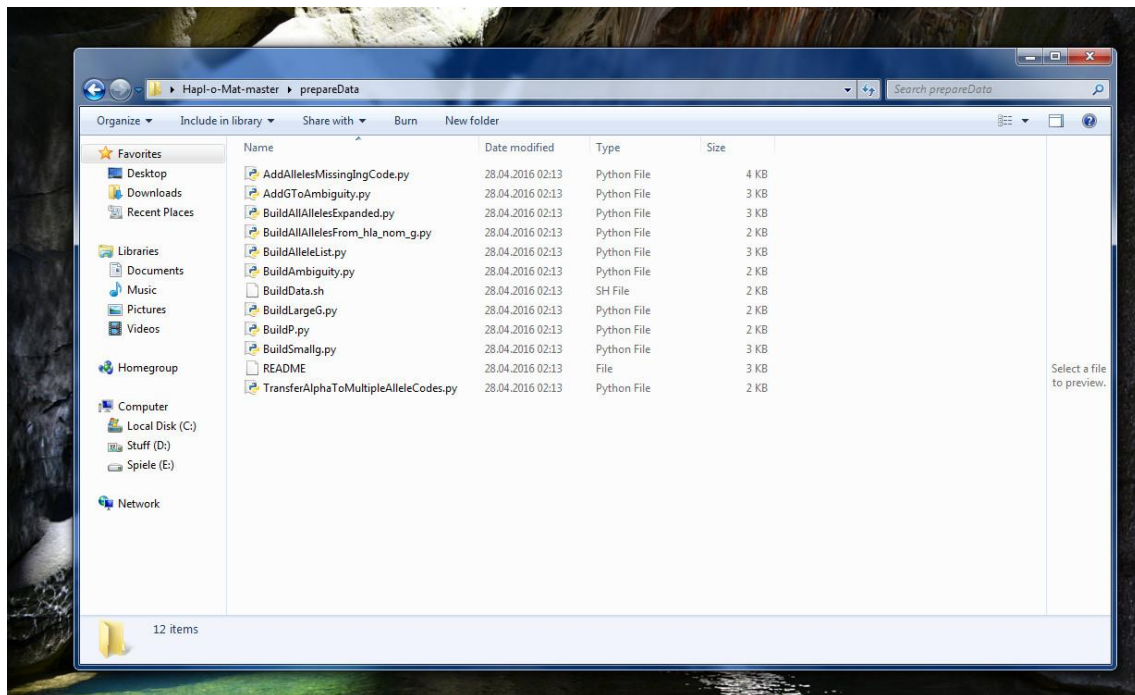
File name	Description
AllAllelesExpanded.txt	A list of relevant existing HLA alleles with their enclosed more-digit typing resolutions
AlleleList.txt	If your input data in GL format includes a missing single-locus genotype, it can be replaced by combining all alleles of the same locus from this file
Ambiguity.txt	Data basis for the ambiguity filter
LargeG.txt	A list of G-groups with their enclosed alleles in 8-digit resolution
MultipleAlleleCodes.txt	A list of multiple allele codes and their translation to alleles
P.txt	A list of P-groups with their enclosed alleles in 8-digit resolution
Smallg.txt	A list of g-groups with their enclosed alleles in 8-digit resolution

In the following we are going to create these data files.

## Download Data

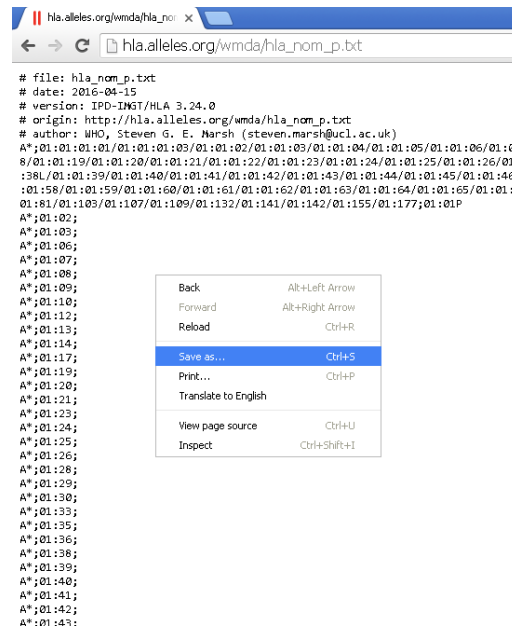
Check the folder “prepareData”.





You find a number of Python scripts (files ending with .py) which we are going to use to produce the data. But first we need some information from the web. Save all files in one folder, e.g. "InputData".

- 1) Go to the website [http://hla.alleles.org/wmda/hla\\_nom\\_p.txt](http://hla.alleles.org/wmda/hla_nom_p.txt) and save the file hla\_nom\_p.txt by right-clicking and choosing "Save as..."



- 2) Go to the website [http://hla.alleles.org/wmda/hla\\_nom\\_g.txt](http://hla.alleles.org/wmda/hla_nom_g.txt) and save the file hla\_nom\_g.txt (same as in 1))
- 3) Go to the website <https://bioinformatics.bethematchclinical.org/HLA-Resources/Allele-Codes/Allele-Code-Lists/Allele-Code-List-in-Alphabetical-Order/>. Click on "Alphabetical Allele Code List (ZIP) (new nomenclature)" and save alpha.v3.zip.

Allele Code List in Alphabetical Order

bioinformatics.bethematchclinical.org/hla-resources/allele-codes/allele-code-lists/allele-code-list-in-alphabetical-order/

MARROW DONOR PROGRAM THE MATCH

About Us

HLA Resources Search Strategies HLA Education Policies Contact Us

Haplotype Frequencies Allele Codes HaploStats HLA Typing HML Search Determinants

Allele Codes

Allele Code Nomenclature

Allele Code Lists

Allele Code List in Numerical Order

Allele Code List in Alphabetical Order

Non-Common and Well Documented Alleles

Allele Code Mailing List

HLA Resources > Allele Codes > Allele Code Lists > Allele Code List in Alphabetical Order

Allele Code List in Alphabetical Order

Example:  
AA 01/02/03/05  
AB 01/02  
AC 01/03

The allele code list in alphabetical order is provided in a variety of formats:

HTML Format (.html)

This format is recommended if you simply want to view the allele code list online.

- [Alphabetical Allele Code List \(HTML\) \(new nomenclature\)](#)  
**Note:** New window will open.
- [Alphabetical Allele Code List \(HTML\) \(old nomenclature\)](#)  
**Note:** New window will open.

Text Format (.txt)

Download a zip compressed file. Once extracted, the text file will be called "alpha.txt."

- [Alphabetical Allele Code List \(ZIP\) \(new nomenclature\)](#)  
**Note:** Extraction requires a data compression program such as WinZip.
- [Alphabetical Allele Code List \(ZIP\) \(old nomenclature\)](#)  
**Note:** Extraction requires a data compression program such as WinZip.

The self-extracting executable file has been removed as of 10/21/03. The allele code lists will no longer be available for download in this format. Please mail [new-allelecodes@nmdp.org](mailto:new-allelecodes@nmdp.org) with any questions or concerns regarding this change.

Extract the archive alpha.v3.zip. Afterwards, you can remove the archive.

- Go to the website <https://www.ebi.ac.uk/ipd/imgt/hla/ambig.html>. Click on "Download Excel" for the wanted release (usually the latest) and save ambiguity\_v<>.xls (replace <> by version).



Ambiguous allele combination: x

← → ↻ <https://www.ebi.ac.uk/ipd/imgt/hla/ambig.html>

<https://www.surveymonkey.co.uk/r/F5GHVGF>

## Ambiguous Allele Combinations Search Tool (Beta)

This search tool provides an alternative method of viewing the ambiguous combinations currently detailed in the downloads.

**STEP 1 - Enter the allele you wish to search for**

Query allele:

**STEP 2 - Search**

[Search Now](#)

## Download Ambiguous Allele Combinations files

The IPD-IMGT/HLA Ambiguous Allele Combinations files are available in XML and Microsoft Excel formats for the current release. The PDF versions of the files are no longer being made available from this website because the number of combinations increases rapidly with the release of novel alleles. For this reason, we would encourage our users to use the XML and Microsoft Excel formats where possible. Please note that Microsoft Office 2010 or later is required to view the files.

Older releases are also available in PDF format, as well as XML and Microsoft Excel.

Release	Date	PDF File	Excel File	XML File
3.24.0	2016-04-15	-	Pending	<a href="#">View XML</a>
3.23.0	2016-01-19	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.22.0	2015-10-10	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.21.0	2015-07-06	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.20.0	2015-04-19	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.19.0	2015-01-19	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.18.0	2014-10-10	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.17.0.1	2014-08-21	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.17.0	2014-07-17	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>
3.16.0	2014-04-14	-	<a href="#">Download Excel</a>	<a href="#">View XML</a>

Next we extract the information from the Excel sheet. Open ambiguity\_v<>.xls in Excel and save as ambiguity\_v<>.xslm to run macros.

ambiguity\_v3230.xls [Kompatibilitätsmodus] - Microsoft Excel

Datei Start Einfügen Seitenlayout Formeln Daten Überprüfen Ansicht

[Speichern](#)  
[Speichern unter](#)  
[Öffnen](#)  
[Schließen](#)

**Informationen**

Zuletzt verwendet

Neu

Drucken

Speichern und Senden

Hilfe

[Optionen](#)  
[Beenden](#)

### Informationen zu ambiguity\_v3230

\\tuesma02\users\cschaefer\Desktop\InputData\ambiguity\_v3230.xls

**Kompatibilitätsmodus**  
Einige neue Features sind deaktiviert, um Probleme beim Arbeiten mit früheren Versionen von Office zu verhindern. Die Konvertierung dieser Datei aktiviert diese Features, kann jedoch zu Layoutänderungen führen.

**Berechtigungen**  
Jeder kann diese Arbeitsmappe öffnen und beliebige Teile kopieren und ändern.

**Für die Freigabe vorbereiten**  
Bevor Sie diese Datei freigeben, machen Sie sich bewusst, dass sie Folgendes enthält:  

- Dokumenteigenschaften und Name des Autors
- Ausgeblendete Zellen
- Inhalte, die aufgrund des aktuellen Dateityps nicht auf Probleme bei der Barrierefreiheit geprüft werden können

**Versionen**  
Es sind keine früheren Versionen dieser Datei vorhanden.

**Eigenschaften**

Größe 29,1MB  
 Titel Titel hinzufügen  
 Kategorien Tag hinzufügen  
 Kategorien Kategorie hinzufügen

**Verwandte Datumsangaben**

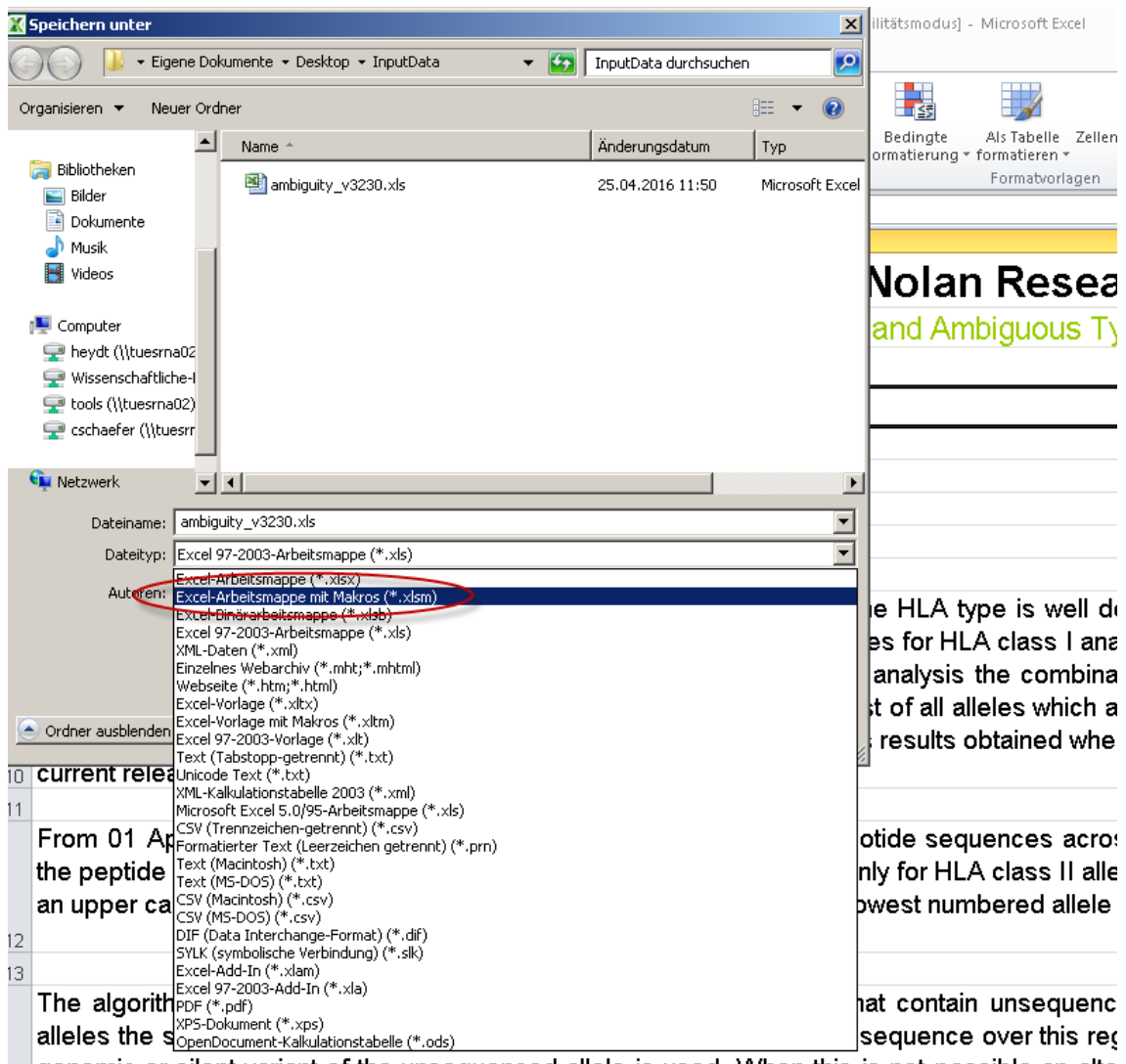
Letzte Änderung 19.01.2016 14:40  
 Erstellt 18.01.2016 12:36  
 Zuletzt gedruckt Nie

**Verwandte Personen**

Autor James Robinson  
 Autor hinzufügen  
 Zuletzt geändert von Anup Raushan Soormally

**Verwandte Dokumente**

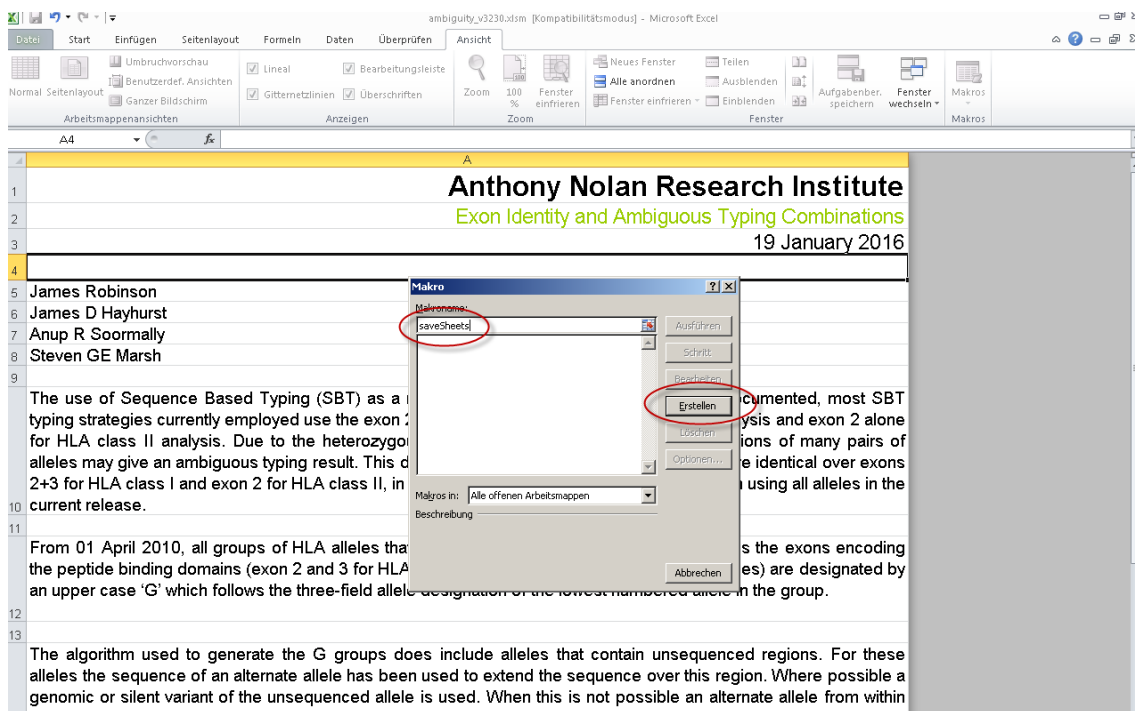
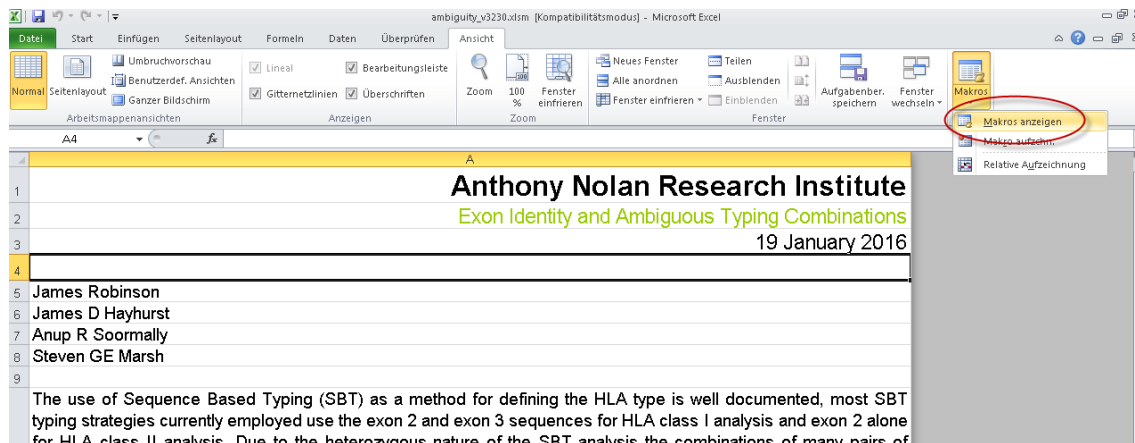
☐ Dateispeicherort öffnen  
[Alle Eigenschaften anzeigen](#)

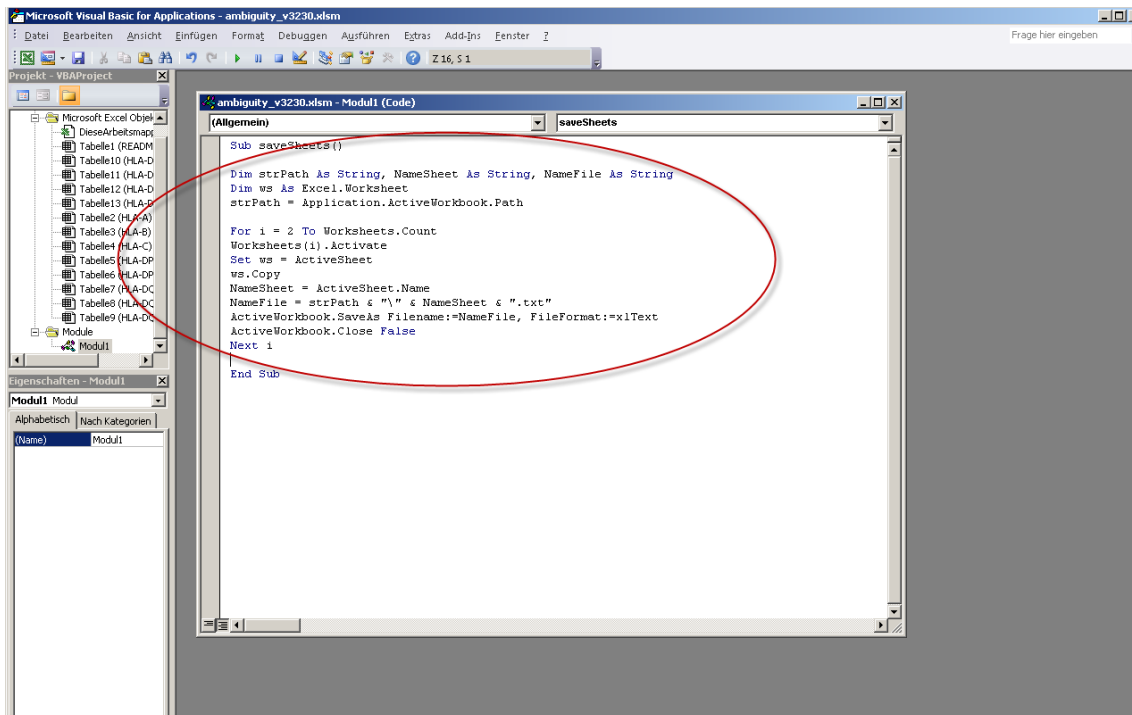


Now insert the following macro, which saves relevant information from the Excelsheets as text files:

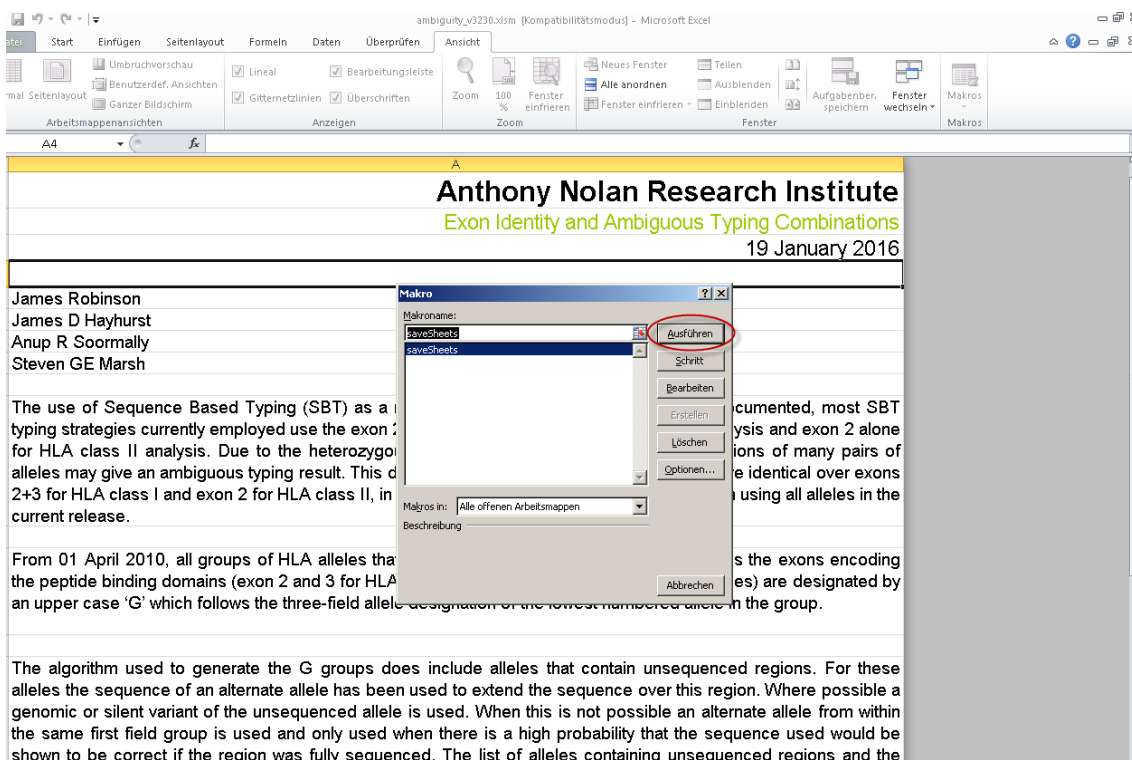
```
Sub saveSheets()
    Dim strPath As String, NameSheet As String, NameFile As String
    Dim ws As Excel.Worksheet
    strPath = Application.ActiveWorkbook.Path

    For i = 2 To Worksheets.Count
        Worksheets(i).Activate
        Set ws = ActiveSheet
        ws.Copy
        NameSheet = ActiveSheet.name
        NameFile = strPath & "\" & NameSheet & ".txt"
        ActiveWorkbook.SaveAs Filename:=NameFile, FileFormat:=xlText
        ActiveWorkbook.Close False
    Next i
End Sub
```





Close the window and execute the macro:



Some new text files should have appeared in your folder „inputData“. Afterwards you can remove the Excel file.

## Build Data for Hapl-o-Mat

Enter the folder InputData and copy all files to the folder “prepareData“. Then enter folder “prepareData“. Next, create the data required for Hapl-o-Mat by running the python scripts in the following order (you can easily get Python here <https://www.python.org/downloads/windows/>):

TransferAlphaToMultipleAlleleCodes.py  
 BuildAllAllelesFrom\_hla\_nom\_g.py  
 BuildAllAllelesExpanded.py  
 BuildP.py  
 BuildLargeG.py  
 BuildSmallg.py  
 BuildAmbiguity.py  
 AddGToAmbiguity.py  
 AddAllelesMissingIngCode.py

Then create the folder Hapl-o-Mat/data and copy the files

AllAllelesExpanded.txt  
 Ambiguity.txt  
 LargeG.txt  
 MultipleAlleleCodes.txt  
 P.txt  
 Smallg.txt

to the folder data.

## Input Genotype Data

Hapl-o-Mat infers haplotypes from population genotype data. It supports different formats of recording genotype data. To use Hapl-o-Mat your data should be in one of the following data formats:

Data format	Description
MA	Ambiguities are encoded by multiple allele (MA) codes. Except for the first line, input files hold an individual's identification number and genotype per line. Genotypes are saved allele by allele without locus name. Identification number and alleles are TAB-separated. The first line of the file is a header file indicating the name of the first column and the loci of the other columns. Same loci must be placed next to each other. For an example refer to "examplePopulations/populationData_a.dat".
GLC	Genotypes with or without ambiguities are saved by genotype list strings. Input files hold an individual's identification number and genotype per line. Identification number and single-locus genotypes are TAB-separated. For an example refer to "examplePopulations/populationData_b.dat"
GL	Genotypes with or without ambiguities are saved by genotype list (GL) strings. Population data is saved in two files. The pull-file contains an individual's identification number and a list of integer numbers, GL-ids, referring to its single-locus genotype. The GL-ids are separated from the identification number via ";" and from each other via ":". The second file, the glid-file, contains a translation from GL-ids starting with "1" to actual single-locus genotypes. GL-id and genotype are separated via ";". A GL-id of "0" is interpreted as a missing typing at the corresponding locus and does not require a translation in the glid-file. For an example refer to "examplePopulations/populationData_c.pull" and "examplePopulations/populationData_c.glid".



READ	Ambiguities are completely resolved and alleles are already translated to the wanted typing resolutions. The input data is of the format as Hapl-o-Mat records processed genotype data. This allows for easily repeating a run without the need to resolve genotype data again.
------	---

When compiling Hapl-o-Mat we assigned MA as argument to the executable. If you want to run another format, you have to change the argument to the appropriate format abbreviation. Right-click in the Project Explorer and select Run As -> Run Configurations.... Then, click on the tab Arguments and change MA to some abbreviation from above.

## Parameters

Each input format for genotype data requires a different set of parameters. The parameters are saved in the corresponding files "parametersMA", "parametersGLC", "parametersGL", and "parametersREAD". All input formats have the following parameters in common:

Parameter	Description
FILENAME_HAPLOTYPES	Name of the file which temporarily saves haplotype names
FILENAME_GENOTYPES	Name of the file which saves resolved genotypes.
FILENAME_HAPLOTYPEFREQUENCIES	Name of the file which saves haplotypes and estimated haplotype frequencies.
FILENAME_EPSILON_LOGL	Name of the file which saves stopping criterion and log-likelihood per iteration.
INITIALIZATION_HAPLOTYPEFREQUENCIES	Initialization routine for haplotype frequencies. It takes the following values: <ul style="list-style-type: none"> <li>"equal": All haplotype frequencies are initialized with the same frequency</li> <li>"numberOccurrence": Haplotype frequencies are initialized according to the initial number of occurrence of the haplotype</li> <li>"random": Haplotype frequencies are initialized randomly</li> <li>"perturbation": Haplotype frequencies are initialized as in numberOccurrence and then randomly modified by a small (&lt;10%) positive or negative offset</li> </ul>
EPSILON	Value for the stopping criterion, i.e. the maximal change between consecutive haplotype frequency estimations is smaller than the assigned value.
CUT_HAPLOTYPEFREQUENCIES	Estimated haplotype frequencies smaller than this value are removed from the output
RENORMALIZE_HAPLOTYPEFREQUENCIES	Takes values "true" and "false". If "true", normalize estimated haplotype frequencies to sum to one. Within machine precision, this becomes necessary, if estimated haplotypes are removed, e.g. via the option CUT_HAPLOTYPEFREQUENCIES

SEED	Set the seed of the used pseudo random number generator. If set to "0", the seed is initialized by the system time.
------	---

Depending on the input format (indicated in brackets), additional parameters are:

Parameter	Input format	Description
FILENAME_INPUT	MA, GLC, READ	The file name of the input population data
FILENAME_PULL	GL	The file name of the pull-file
FILENAME_GLID	GL	The file name of the glid-file
LOCI_AND_RESOLUTIONS	MA, GL, GLC	Loci included into analysis and desired typing resolution per locus. The list is separated by "," and contains the locus name followed by ":" and the desired typing resolution, e.g. A:g,B:4d,C:g. Supported typing resolutions and their abbreviations are g-groups (g), P-groups (P), G-groups (G), 2-digit fields (2d), 4-digit fields (4d), 6-digit fields (6d), and 8-digit fields (8d). Alleles are not translated via the option asltIs (applying the ambiguity filter includes an intrinsic translation to G-groups)
LOCIORDER	GL	Specify the order of loci the individual's GL-ids correspond to. Loci are separated via ",".
RESOLVE_MISSING_GENOTYPES	GL	Takes values "true" and "false". If set to true, a missing typing is replaced by a combination of all alleles from AlleleList.txt at the locus. Else, individuals with a missing typing are discarded from analysis
MINIMAL_FREQUENCY_GENOTYPES	MA, GL, GLC	Genotypes which split into more genotypes than the inverse of this number are discarded from analysis
DO_AMBIGUITYFILTER	MA, GL, GLC	Takes values "true" and "false". The option "true" activates the ambiguity filter
EXPAND_LINES_AMBIGUITYFILTER	MA, GL, GLC	Takes values "true" and "false". If set to "true", matching lines with additional genotype pairs in the ambiguity filter are considered

Whenever specifying a file name including folders, you have to create the folders before running Hapl-o-Mat.

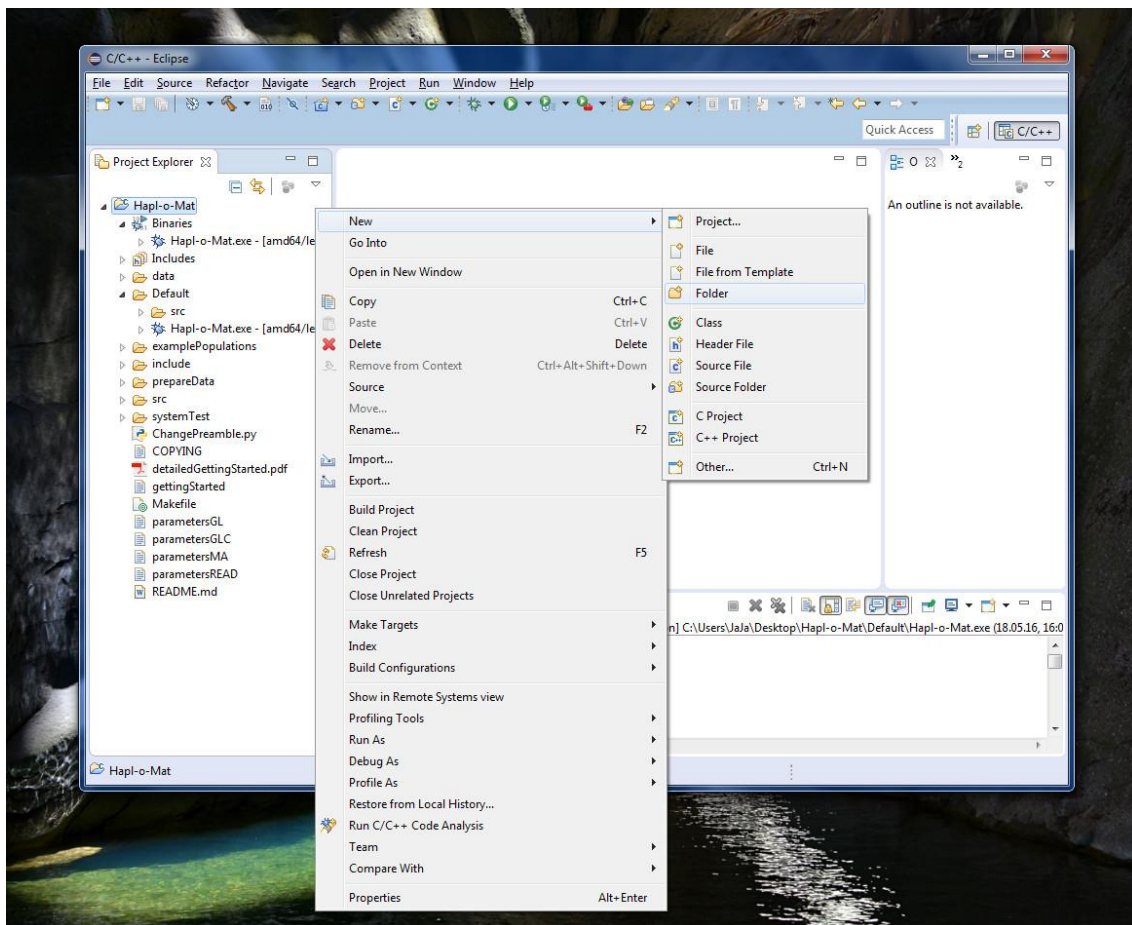
## Tutorials

We have everything ready to use Hapl-o-Mat. In the following we estimate haplotype frequencies from some included genotype data recorded in the input format MA.

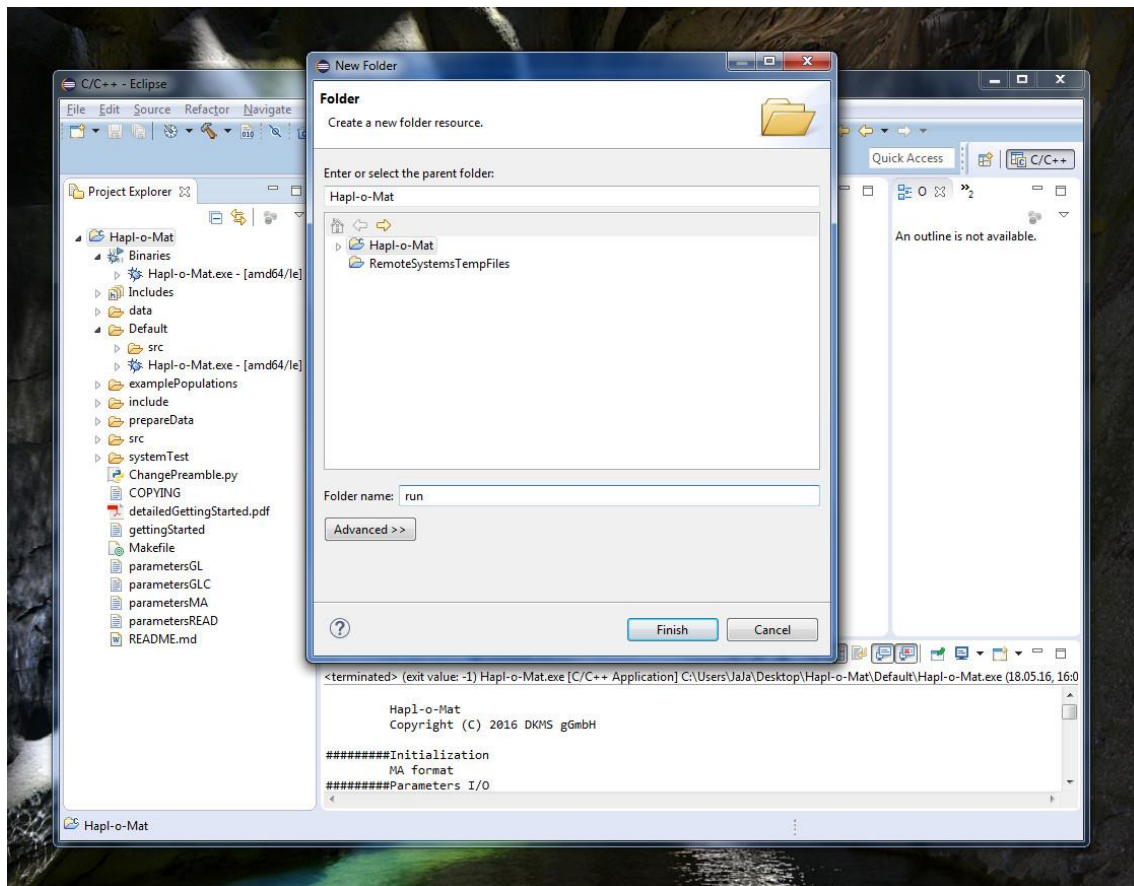
### Input Format MA

You find the relevant population data in "examplePopulations/populationData\_a.dat". As ambiguities are recorded as multiple allele codes, the input format is MA. We are going to infer three locus (A, B, DRB1) haplotypes from this data. Alleles at loci A and B shall be translated to typing resolution g and alleles at locus DRB1 to 4-digits typing resolution.

Create a folder by right-clicking in the Project Explorer and choosing New -> Folder.

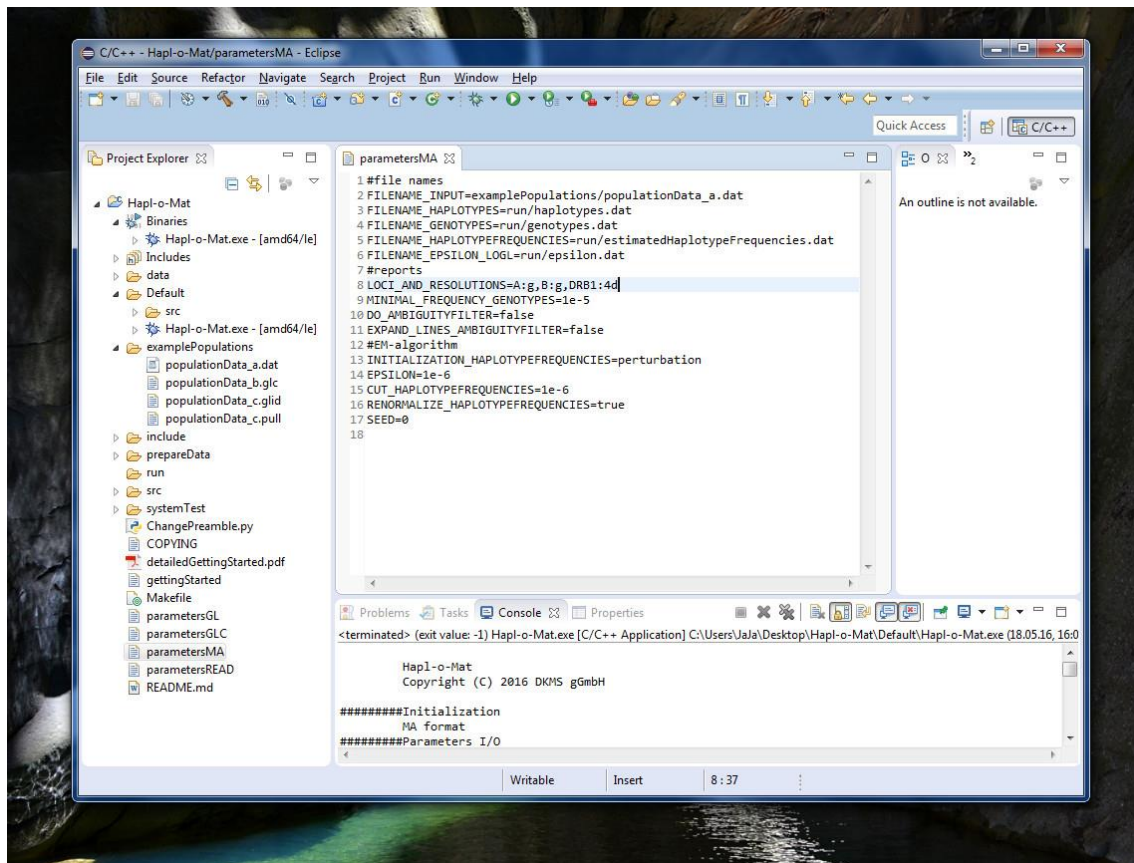


Name the folder run and then click Finish.

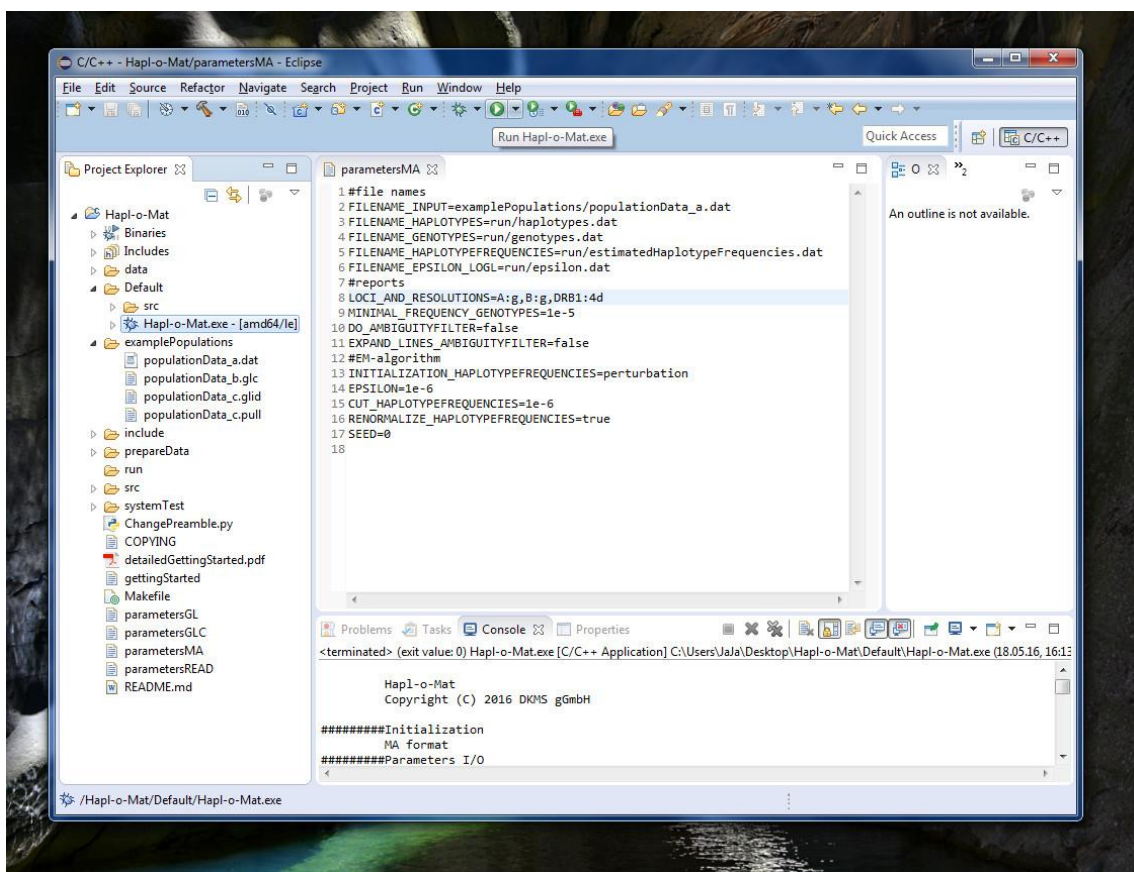


Next, open the parameter file parametersMA by double-clicking on it. Change its parameters according to:

```
#file names
FILENAME_INPUT=../populationData_a.dat
FILENAME_HAPLOTYPES=run/haplotypes.dat
FILENAME_GENOTYPES=run/genotypes.dat
FILENAME_HAPLOTYPEFREQUENCIES=run/hfs.dat
FILENAME_EPSILON_LOGL=run/epsilon.dat
#reports
LOCI_AND_RESOLUTIONS=A:g,B:g,DRB1:4d
MINIMAL_FREQUENCY_GENOTYPES=1e-5
DO_AMBIGUITYFILTER=false
EXPAND_LINES_AMBIGUITYFILTER=false
#EM-algorithm
INITIALIZATION_HAPLOTYPEFREQUENCIES=perturbation
EPSILON=1e-6
CUT_HAPLOTYPEFREQUENCIES=1e-6
RENORMALIZE_HAPLOTYPEFREQUENCIES=true
SEED=1000
```



Compute haplotype frequencies from the genotype input data by running Hapl-o-Mat. To this end, click on the green Play button in the upper navigation bar.





It produces some output on the screen including your chosen parameters, statistics on the resolved genotype data and the expectation-maximization algorithm, and the run time.

Now let's examine the results produced by Hapl-o-Mat. The results are saved in the folder run (press F5 to update, if they do not appear). We first look into the file with the resolved genotypes, "run/genotypes.dat".

```

1 1 NNN 1 A*26:04+A*29:67*B*07:218+B*54:16*DRB1*11:182+DRB1*13:14
2 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:01*DRB1*12:01+DRB1*14:23
3 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:01*DRB1*12:01+DRB1*14:23
4 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:14*DRB1*12:01+DRB1*14:23
5 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:14*DRB1*12:01+DRB1*14:23
6 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:19*DRB1*12:01+DRB1*14:23
7 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:19*DRB1*12:01+DRB1*14:23
8 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:01*DRB1*12:06+DRB1*14:23
9 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:01*DRB1*12:06+DRB1*14:23
10 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:14*DRB1*12:06+DRB1*14:23
11 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:14*DRB1*12:06+DRB1*14:23
12 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:19*DRB1*12:06+DRB1*14:23
13 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:19*DRB1*12:06+DRB1*14:23
14 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:01*DRB1*12:10+DRB1*14:23
15 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:01*DRB1*12:10+DRB1*14:23
16 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:14*DRB1*12:10+DRB1*14:23
17 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:14*DRB1*12:10+DRB1*14:23
18 2 III 0.098765432098765 A*02:570+A*24:02g*B*13:07N+B*14:19*DRB1*12:10+DRB1*14:23
19 2 III 0.012345679012346 A*02:570+A*24:50*B*13:07N+B*14:19*DRB1*12:10+DRB1*14:23
20 3 NNN 1 A*02:570+A*29:67*B*13:07N+B*35:54*DRB1*13:121+DRB1*14:23
21 4 NIN 0.333333333333333 A*02:77+A*66:10*B*14:01+B*15:154*DRB1*16:30+DRB1*16:30
22 4 NIN 0.333333333333333 A*02:77+A*66:10*B*14:14+B*15:154*DRB1*16:30+DRB1*16:30
23 4 NIN 0.333333333333333 A*02:77+A*66:10*B*14:19+B*15:154*DRB1*16:30+DRB1*16:30
24 5 NNN 1 A*03:217+A*29:36*B*15:01g*B*35:54*DRB1*11:95+DRB1*14:23
25 6 NNN 1 A*02:454+A*03:93*B*15:316+B*35:54*DRB1*04:52+DRB1*08:18
26 7 NNN 1 A*24:02g+A*30:13*B*18:19+B*39:27*DRB1*13:116+DRB1*14:23

```

```

<terminated> (exit value: 0) Hapl-o-Mat.exe [C:\C++ Application] C:\Users\Jala\Desktop\Hapl-o-Mat\Default\Hapl-o-Mat.exe (18.05.16, 17:15)
Different haplotypes: 876

Memory requirement haplotypes [MB]: 0.014064
Memory requirement genotypes [MB]: 0.028336

#####EM algorithm
Used 111 steps
Sum haplotype frequencies: 1
Sum output haplotype frequencies: 1

```

The first column corresponds to the individual's identification number. The second column indicates how ambiguities per single-locus genotypes have been resolved. If no ambiguity occurred or no additional genotypes are formed, the type is N. If an ambiguity occurred and was resolved via building all possible allele combinations, the type is I. Activating the ambiguity filter gives additional types: A, if one matching line in the ambiguity file was found, and M if multiple matching lines were found. The third column gives the frequency of the genotype and the fourth column the genotype itself. The genotype is saved in the GL format. If an individual's genotype splits into a set of genotypes, each genotype is written to one line starting with the same identification number. The corresponding frequencies become non-integer and sum to one.

The evolution of the stopping criterion and log-likelihood while iterating expectation and maximization steps is written to "run/epsilon.dat". The first column is the stopping criterion and the second one the not normalized log-likelihood.

```
1 1.63633915820329e-002 -8.4550108234294e+002
2 6.67894860870962e-003 -7.98264226478726e+002
3 2.52753700703970e-003 -7.93704761564617e+002
4 2.37110567041985e-004 -7.93448324807849e+002
5 3.80564892720948e-004 -7.93251538797204e+002
6 7.94531713717693e-004 -7.92615113678754e+002
7 1.25134929891374e-003 -7.91248930961018e+002
8 1.34419828966542e-003 -7.89708560775000e+002
9 7.87340220656511e-004 -7.88654214928568e+002
10 8.67044594283474e-004 -7.87769554535044e+002
11 8.09622574005869e-004 -7.86925312040129e+002
12 3.61677286719145e-004 -7.8659902363331e+002
13 3.63679053477610e-005 -7.86573664584397e+002
14 2.48608336182350e-005 -7.86573278860631e+002
15 2.26157003604885e-005 -7.86573068307770e+002
16 2.06790142148333e-005 -7.86572886196057e+002
17 1.89950287026487e-005 -7.86572727464074e+002
18 1.75202776769197e-005 -7.86572588152510e+002
19 1.62204268588893e-005 -7.86572465121285e+002
20 1.50680365537375e-005 -7.8657235851826e+002
21 1.40409442342254e-005 -7.86572258303325e+002
22 1.31210768740814e-005 -7.86572170806667e+002
23 1.2235665948050e-005 -7.86572091905070e+002
24 1.15460835489242e-005 -7.86572020694022e+002
25 1.08683267890028e-005 -7.86571955975155e+002
26 1.02516314710883e-005 -7.86571897020685e+002
```

```
<terminated> (exit value: 0) Hapl-o-Mat.exe [C/C++ Application] C:\Users\Jala\Desktop\Hapl-o-Mat\Default\Hapl-o-Mat.exe (18.05.16, 17:15)
Different haplotypes: 876

Memory requirement haplotypes [MB]: 0.014064
Memory requirement genotypes [MB]: 0.028336
*****EM algorithm
Used 111 steps
Sum haplotype frequencies: 1
Sum output haplotype frequencies: 1
```

The inferred haplotypes including estimated frequencies are listed in "run/hfs.dat". Haplotypes are saved in the GL format. This is the file you were aiming at. It is sorted by descending frequency and already normalized if you activated the corresponding option (we did in this tutorial).

```
1 A*29:25-B*15:101-DRB1*16:30 0.0600000000000000
2 A*11:01g-B*40:01g-DRB1*07:56 0.0400000000000000
3 A*24:02g-B*15:27-DRB1*14:85 0.0300000000000000
4 A*11:01g-B*15:31g-DRB1*11:182 0.0300000000000000
5 A*68:70-B*37:19-DRB1*13:52 0.0300000000000000
6 A*30:13-B*18:12-DRB1*11:147 0.0250000000000000
7 A*03:02g-B*35:32-DRB1*09:02 0.0250000000000000
8 A*32:33-B*54:16-DRB1*11:182 0.0250000000000000
9 A*30:73N-B*14:08-DRB1*14:01 0.02013358702001
10 A*03:93-B*35:54-DRB1*04:52 0.0200000000000000
11 A*26:04-B*54:16-DRB1*11:182 0.0200000000000000
12 A*03:217-B*15:01g-DRB1*11:95 0.0200000000000000
13 A*03:217-B*35:51-DRB1*14:85 0.0200000000000000
14 A*02:570-B*13:07N-DRB1*14:23 0.0200000000000000
15 A*02:454-B*15:31g-DRB1*08:18 0.0200000000000000
16 A*66:10-B*15:154-DRB1*16:30 0.0150000000000000
17 A*66:10-B*51:01g-DRB1*04:52 0.0150000000000000
18 A*32:73-B*51:114-DRB1*07:56 0.0150000000000000
19 A*03:239-B*27:98-DRB1*03:103 0.0150000000000000
20 A*11:109N-B*55:40-DRB1*13:116 0.0150000000000000
21 A*29:36-B*15:154-DRB1*04:129 0.0150000000000000
22 A*02:163-B*37:28-DRB1*16:30 0.0150000000000000
23 A*02:258-B*42:14-DRB1*13:121 0.0150000000000000
24 A*68:94N-B*39:27-DRB1*03:103 0.0150000000000000
25 A*23:06-B*08:145-DRB1*16:30 0.0150000000000000
26 A*68:70-B*35:32-DRB1*13:66 0.0150000000000000
```

```
<terminated> (exit value: 0) Hapl-o-Mat.exe [C/C++ Application] C:\Users\Jala\Desktop\Hapl-o-Mat\Default\Hapl-o-Mat.exe (18.05.16, 16:13)
*****Parameters I/O
Input: examplePopulations/populationData_a.dat
Output haplotypes: run/haplotypes.dat
Output genotypes: run/genotypes.dat
Output estimated haplotype frequencies: run/estimatedHaplotypeFrequencies.dat
Output epsilon and log(L): run/epsilon.dat
*****Parameters resolving genotypes
Minimal frequency of genotypes: 1e-005
```