workreport(3rd)

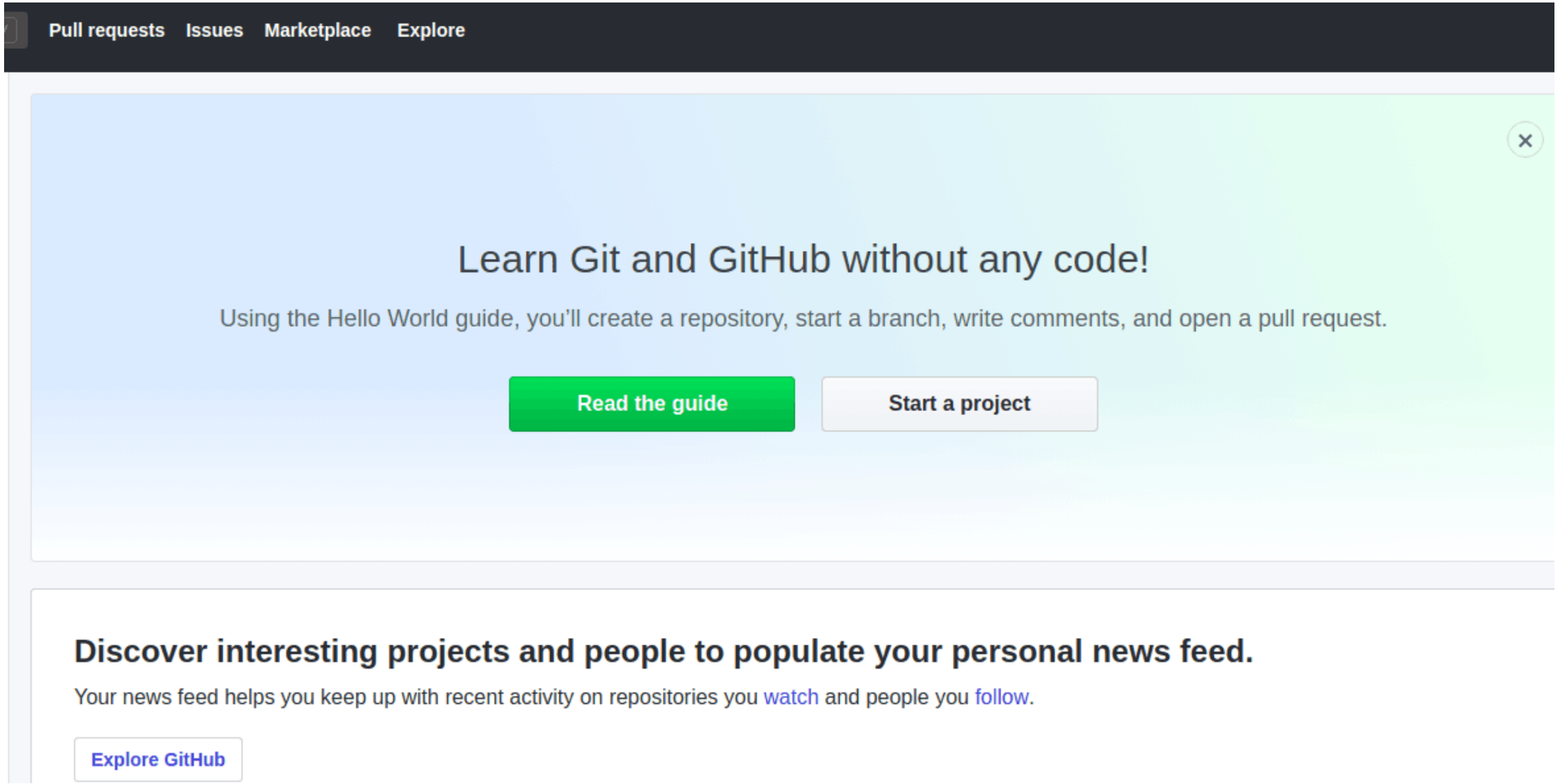
孙笳淋 杨焱辉

2019/7/14

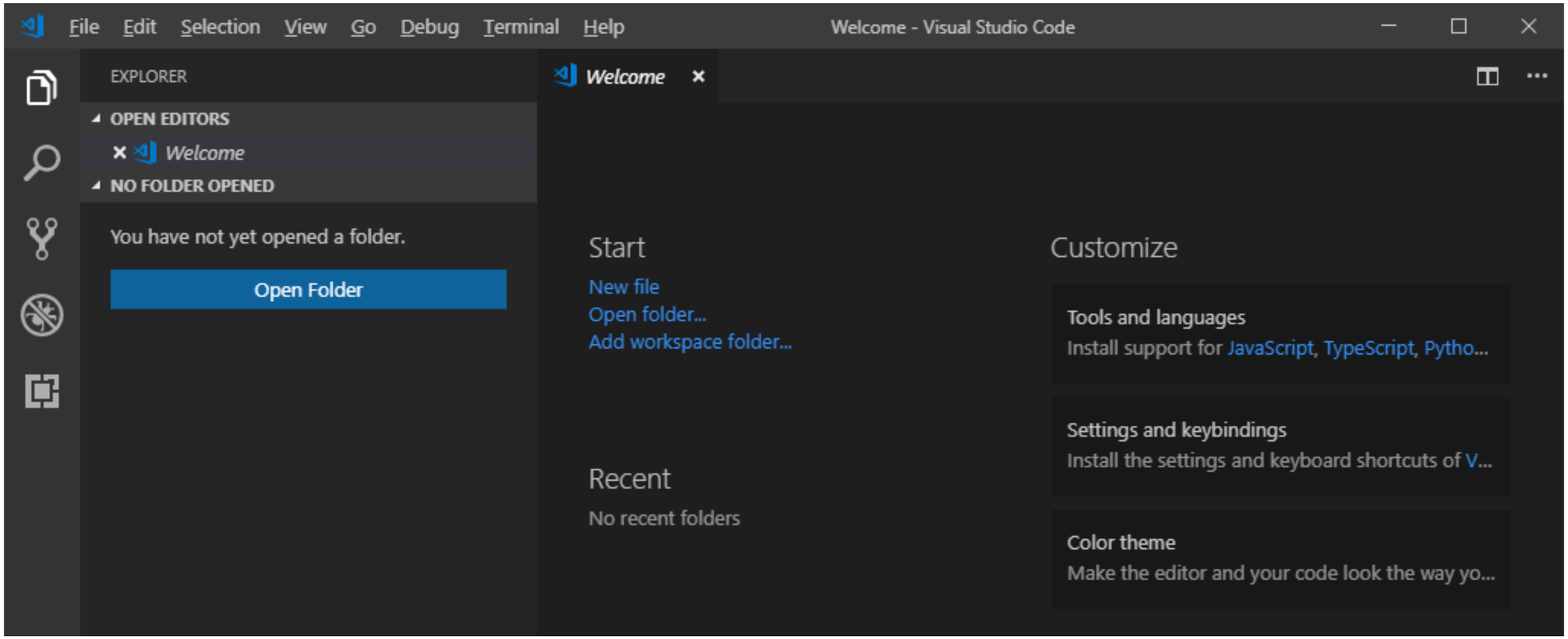
1.Accomplishment：Constructing Working Enviroment on

Ubuntu

- 1.1 A linux system(something went wrong with my HHD and I had to reinstall two systems)(Sun)
- 1.2 Register in github and get familiar with some basic operations following the instructions(repository,branch,pull request)

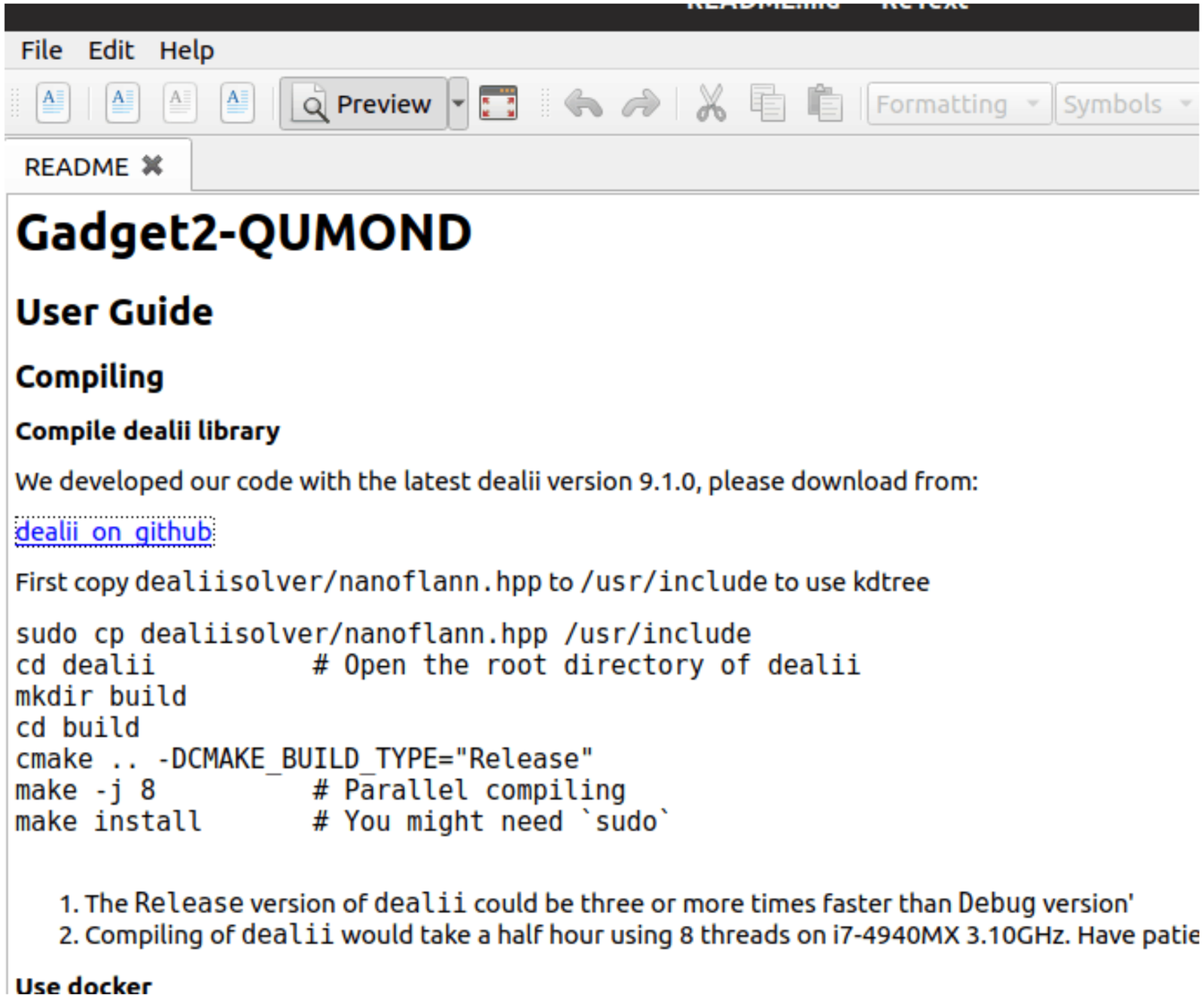


- 1.3 Install vscode (including C/C++ tools) and complete a simple 'hello world' programme in order to know how to encode, build and run a project via vscode.



- 1.4 Install Gadget2,(mpi,GSL and fftw),run a programme given in the parameter package to test whether it works and succeed to give output files
- 1.5 Install splash, and it was able to show the results above successfully in 2D patterns
- 1.6 Install GNUPLOT to realize data visualization

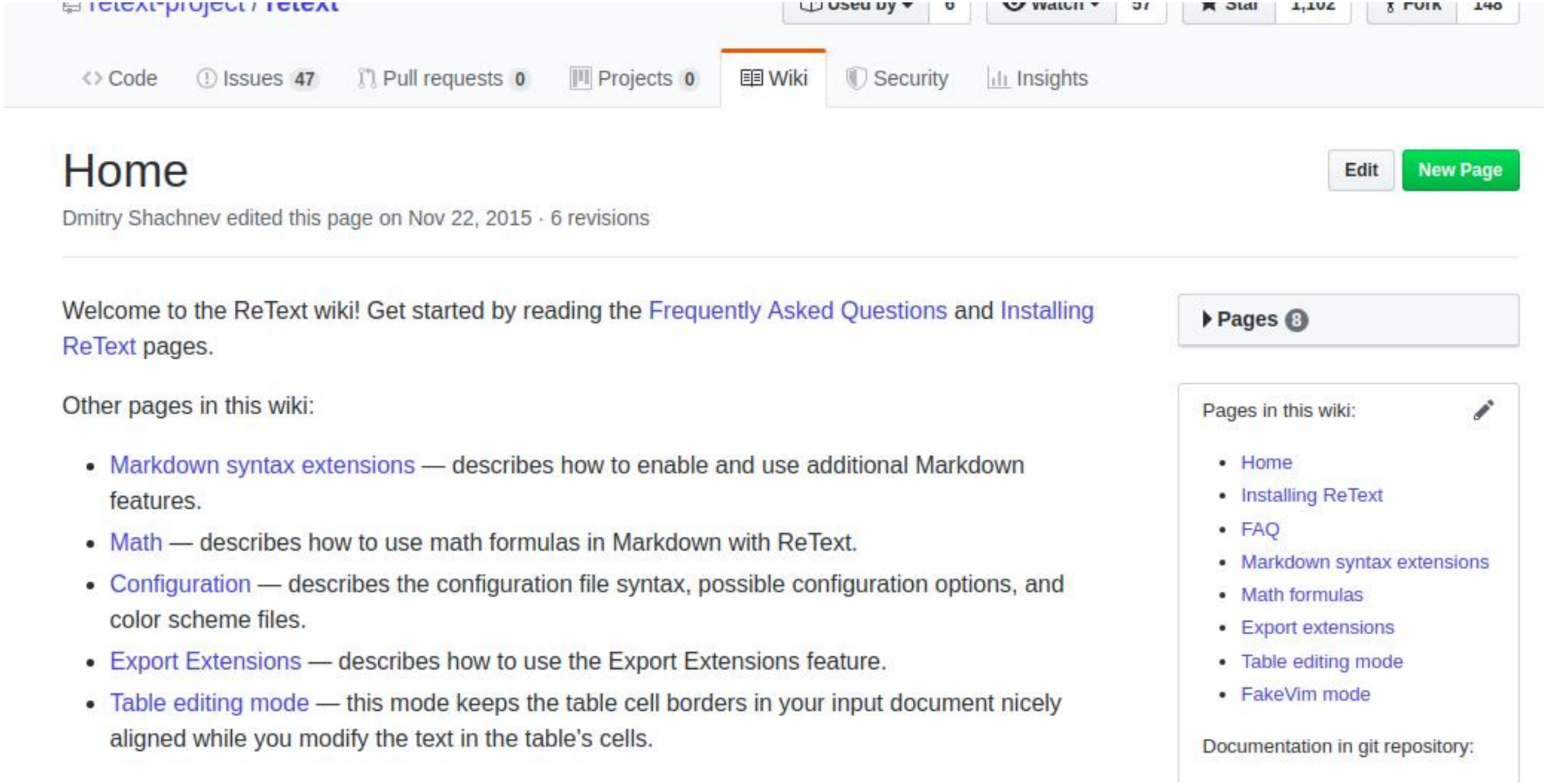
- 1.7 Install the G2-MOND package and deal.II package for later using, instructed by Meng



README.md

- 1.8 Installed Retext and learned syntax of markdown

Markdown learning: https://github.com/retext-project/retext/wiki



- 1.9 Learned the using of text editor vi

linux下终端常用命令和vi命令修改文件及保存的使用方法

2016年09月04日 10:38:16 DaveBobo 阅读数 29861

首先介绍一下Ubuntu下各个目录的一般作用：

/

这就是根目录，一台电脑有且只有一个根目录，所有的文件都是从这里开始的。举个例子：当你在终端里输入"/home"，你其实是
录）开始，再进入到home目录。

/root

系统管理员（root user）的目录。至于系统管理员的权限有多大我这里就不在废话了。因此，请小心使用root帐号。

/boot

系统启动文件，所有与系统启动有关的文件都保存在这里 。

/bin

这里是存放系统的程序。

/etc

主要存放了系统配置方面的文件。

/dev

https://blog.csdn.net/davebobo/article/details/52431014

- 1.10 Learned how to compile and run a simple C program on Linux

succeed in encoding,building and running a 'hello world' programme both using vscode and the linux terminal

https://blog.csdn.net/geeker_12/article/details/10911275

2.Reading document: Gadget 2 usrs guide

G2 usrs guide
2.1 install(details about installing)

- **mpi** ./configure make sudo make install
- **gsl** sudo apt install libgsl-dev
- **fftw** fftw 2.1.5

  ./configure --enable-mpi --enable-type-prefix --enable float

  make

  sudo make install

  ./configure --enable-mpi --enable-type-prefix

  make clean&&make&&sudo make install

(--enable-mpi :ensure parallel support enabled

--enable-type-prefix:obtain the libs in both single and double pricision version

after a make clean and reconfigure without(with)the --enable-float, the double(single) version will be installer following the single(double)version)

  fftw 3.3.8

  ./configure --enable-mpi

- **Gadget2** modify the makefile(the sources of lib and include) make and make install

G2 usrs guide
2.2things about running a simulation

start mpirun -np n filename

interrupt automatically write restart files before cpu-time is used up

interrupt mannually echo > stopfiles

restart from restart files mpirun -np n filename 1

(remember to add 1 to the end of the order or it will restart from the beginning,donot try different systems,make sure the n of processors is the same)

restrat from snapshots(ss) 1. specify ss as initial condition file

reset TIMEBEGIN

change the base filename

dont need operation 1

2.add 2 as a operation to the end of the file

set the ss name as the initial conditions file name

a series of ss files :give the base nam

in this way the number of processors can be different

G2 usrs guide
2.3.Types of simulations

1.cosmological integrations with comoving coordinates requires ComovingCoordinateON in the *.param file in the /Gadget2/parameterfiles

2.set the compile-time switch appropriately to adopt periodic boundary conditions and TreePM(TPM below) algorithm

3.to switch TPM algorithm on : passing the desired mesh-size at compile time (dis# the PMGRID at the latter parter of makefile)

4.using an explicit force split : the long-range force is computed with Fourier techniques, while the short-range force is done with the tree(tree method needs walking locally and both periodic and non-periodic boundary conditions are implemented for the tree method)

5.Pure SPH simulation can be done without self-gravity in periodic boxes whose dimensions in each direction is multiples of the basic boxes

SPH runs possiblly only in 2Ds with self-gravity (three coordinates are still stored for all particles and the computations are formally carried out as in the 3D case, except that all particles lie in one coordinate plane, etc. either have equal x-, y-, or z-coordinates)

G2 usrs guide
2.3Types of simulations(a summary of the types)

the following table adopt some abbreviations ,more information at Schematic overview of the different types of simulations possible with GADGET-2 page 8, table 1

Type of SImulation	Computational Methods	Remarks
Nt space	G:Tree,SPH(optional),vacuum boundcons	OMegaLambda:0
Per long box	No G,only SPH, per boundcons	NO G needs setting,LONG-x/y/z to scale the dime of box
Cos phy coor	G:Tree,SPH,vacuum boundaries	ComovingIntegrationON :0
Cos com coor	G:Tree,SPH,vacuum boundaries	ComovingIntegrationON :1
Cos,com per box	G:Tree with Ewald-correction,SPH, per boundaries	PERIODIC needs setting
Cos,com coor, TreePM	G:Tree with long range PM,SPH,vacuum boundaries	PMGRID needs setting
Cos,com per box,TreePM	G:Tree with long range PM,SPH, per boundaries	PERIODIC&PMGRID need setting
Cos,com coor,TreePM,Zoom	G:Tree with long range and intermediate range PM,SPH, vacuum boundaries	PMGRID&PLACEHIGHRESREGION need setting
Cos,per com box,TreePM,Zoom	G:Tree with long range and intermediate range PM,SPH, vacuum boundaries	PERIODIC,PMGRID&PLACEHIGHRESREGION need setting
Nt space,Tree PM	G: Tree with long range PM,SPH, vacuum boundaries	PMGRID needs setting

G2 usrs guide
2.4.Makefile options

1.**makefile**:compile-time sptions

parameterfile: run-time options

this allows the generation of highly optimised binaries by the compiler

disadvantage:different simulations may require different binary executables of GADGET-2

2.produce a separate executable for each running simulation in order to avoid danger caused by several simulations that are running concurrently and the risk of using the wrong code for a simulation:to make a copy of the whole simulation source code together with its makefile in the output directory of each simulation run, and then use this copy to compile the code and to run the simulation

3.once the makefile options is changed, a full compilation is neccessary ,

that's why the makefile itself has been added to the list of dependences of each source files, such that a complete recompilation should happen automatically when the makefile is changed and the command make is given

a command make clean followed by a make can enforce the code to be recomplied mannualy (make clean&&make)

G2 usrs guide
2.4.Makefile options

1.basic operation mode of code

- **PERIODIC**: periodic boundaries(we can also see this in the table given above or 2.3)
- **UNEQUALSOFTENINGS**:if adopting different gravitational softening lengths

2.default settings that are highly recommended

- **PEANOHILBERT**:bring the particles into Peano-Hilbert order after each domain decomposition, which improves cache utilisation and performance
- **WALLCLOCK**:a wallclock timer is used by the code to measure internal time consumption ,**Otherwise**, a timer that measures consumed processor ticks is used

G2 usrs guide
2.4.Makefile options

3.TreePM options:

- **PMGRID=128**:enabling the TreePM method**(long-range force is computed with a PM-algorithm, and the short range force with the tree)**

  parameter has to be set to the size of the mesh that should be used(etc,64,96,128)

note: 1.The mesh dimensions need not necessarily be a power of two, but the FFT is fastest for such a choice

2. If the simulation is not in a periodic box,then a FFT method for vacuum boundaries is employed, using a mesh with dimension twice that specified by PMGRID

- **PLACEHIGHRESREGION=1+8**:**(only work together with PMGRID) the long** One Fourier-grid is used to cover the whole simulation volume, allowing the computation of the large-scale force. A second Fourier mesh is placed on the region occupied by ‘high-resolution’ particles, allowing the computation of an intermediate-scale forcethe

  force on very small scales is computed by the tree method

the above methods is useful for ‘zoom-simulations’(where the majority of particles (the high-res particles) are occupying only a small fraction of the volume)

if types 0, 1, and 4 are the high-res particles, then the parameter should be set to PLACEHIGHRESREGION=1+2+16, (2^0,2^1,2^4)

Note: If a periodic box is used, the high-res zone is not allowed to intersect the box boundaries

G2 usrs guide
2.4.Makefile options

3.TreePM options:

- **ENLARGEREGION=1.1(110% of the initial,etc)**(the simulation will be interrupted if high-res particles leave this region in the course of the run):setting this parameter to a value larger than one, the high-res region can be expanded

If in addition SYNCHRONIZATION is activated, then the code will be able to continue even if high-res particles leave the initial high-res grid,and the code will update the size and position of the grid that is placed onto the high-resolution region automatically**(To prevent that this potentially happens every single PM step,assign a value SLIGHTLY larger than 1 to ENLARGEREGION)**

- **ASMTH=1.25**: override the value assumed for the scale that defines the long-range/short-range force-split in the TreePM algorithm (1.25 is the default value, given in mesh-cells)
- **RCUT=4.5**:override the maximum radius in which the short-range tree-force is evaluated (in case the TreePM algorithm is used). (4.5 is the default value, given in mesh-cells)

G2 usrs guide
2.4.Makefile options

4.single/double pricision

- **DOUBLEPRECISION**:the code store and compute internal particle data in double precision(output files are written by converting the values that are saved to single
  - precision)
- **DOUBLEPRECISION_FFTW**: use the double-precision version of FTTW(the latter has been explicitly installed with a ”d” prefix, and NOTYPEPREFIX_FFTW is not set. Otherwise the single precision version (”s” prefix) is used)

G2 usrs guide
2.4.Makefile options

- 5.Time Integrade options
- **SYNCHRONIZATION**:particles may only increase their timestep if the new timestep will put them into synchronisation with the higher time level

TreePM always proposes SYN be open.

- **FLEXSTEPS**:an alternative to SYN Particle timesteps are here allowed to be integer multiples of the minimum timestep that occurs among the particles, which in turn is rounded down to the nearest power of two devision of the total simulated time span,**result in a reduction of work-load imbalance losses**
- **PSEUDOSYMMETRIC**:anticipate timestep changes by extrapolating the change of the acceleration into the future,improving the long-term integration behaviour of periodic orbits as a rusult.

This option will become effectless if FLEXSTEPS is set

- **NOSTOP_WHEN_BELOW_MINTIMESTEP**:the code will not terminate when the timestep falls below the value of MinSizeTimestep specified in the parameterfile, useful for runs where one wants to enforce a constant timestep for all particles

Activate this option and set the MinSizeTimestep and MaxSizeTimestep to an equal value

- **NOPMSTEPADJUSTMENT:** the long-range timestep for the PM force computation is always determined by MaxSizeTimeStep, if not activated, , it is set to the minimum of MaxSizeTimeStep and the timestep obtained for the maxi_x0002_mum long-range force

G2 usrs guide
2.4.Makefile options

- 6.output options
- **HAVE_HDF5** the code will be compiled with support for input and output in the HDF5**(abbreviated for Hierachical Dada Format)** format. The HDF5 libraries and headers are required to beinstalled. **The HDF5 format can then be selected as format "3" in Gadget's *parameterfile*.**
- **OUTPUTPOTENTIAL:**This will force the code to compute gravitational potentials for all particles each time a snapshot file is generated and the values are included in the snapshot files. ***The computation of the values of the potential costs additional time.***
- **OUTPUTACCELERATION:**including the physical acceleration of each particle in snapshot files
- **OUTPUTCHANGEOFENTROPY:**including the rate of change of entropy of gas particles in snapshot files
- **OUTPUTTIMESTEP:**including the timesteps actually taken by each particle in the snapshot files

3.Plans for next week(Reading documents&Learning new things)

- 3.1 go on reading **G2 users guide(kind of tediously long and complicated)**(part finished,about 40% or maybe ,emmm,less done)& read G2 paper(0% done)

makefile options(more than 70% finished), parameterfile(0%),file names(0%) and file formats(0%) .....(0%)

3.Plans for next week(Reading documents&Learning new things)

- 3.2 read Tree documents
- 3.3 read VS Code “Get Started” (0% done)
- 3.4 read splash documents: learn how to realize data visualization(0% done)
- 3.5markdown: get familiar with it and learn how to type math formulas(0% done)