Zhang-Lab 生信小课堂 第二一期

Applied Bioinformatics Club (ABC)

和趣求真 秉实生信

（张建伟生物信息学课题组 https://zhang.hzau.edu.cn）

在VSCode中远程编写和调试集群Linux程序

2025.04.11 二综C102 15:00 欢迎大家交流学习！

主讲人：刘健

2025/04/11

目录

1.使用VSCode远程连接集群的优点

VSCode：轻量级代码编辑器，根据需要下载对应的插件

直接在windows界面上操作linux文件，vscode能轻松远程下载插件，更加便捷。

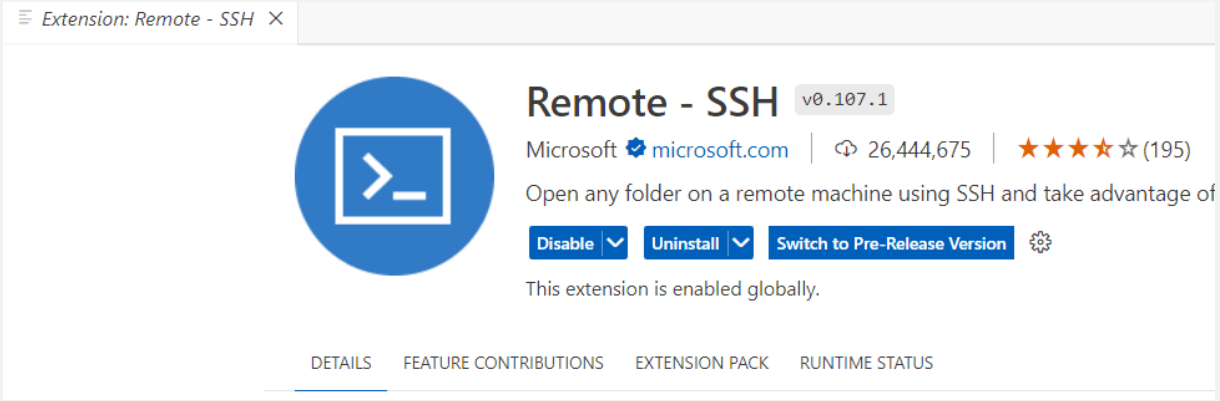
2.设置流程

查看集群上Glibc(Linux系统的基础api)的版本：ldd --version

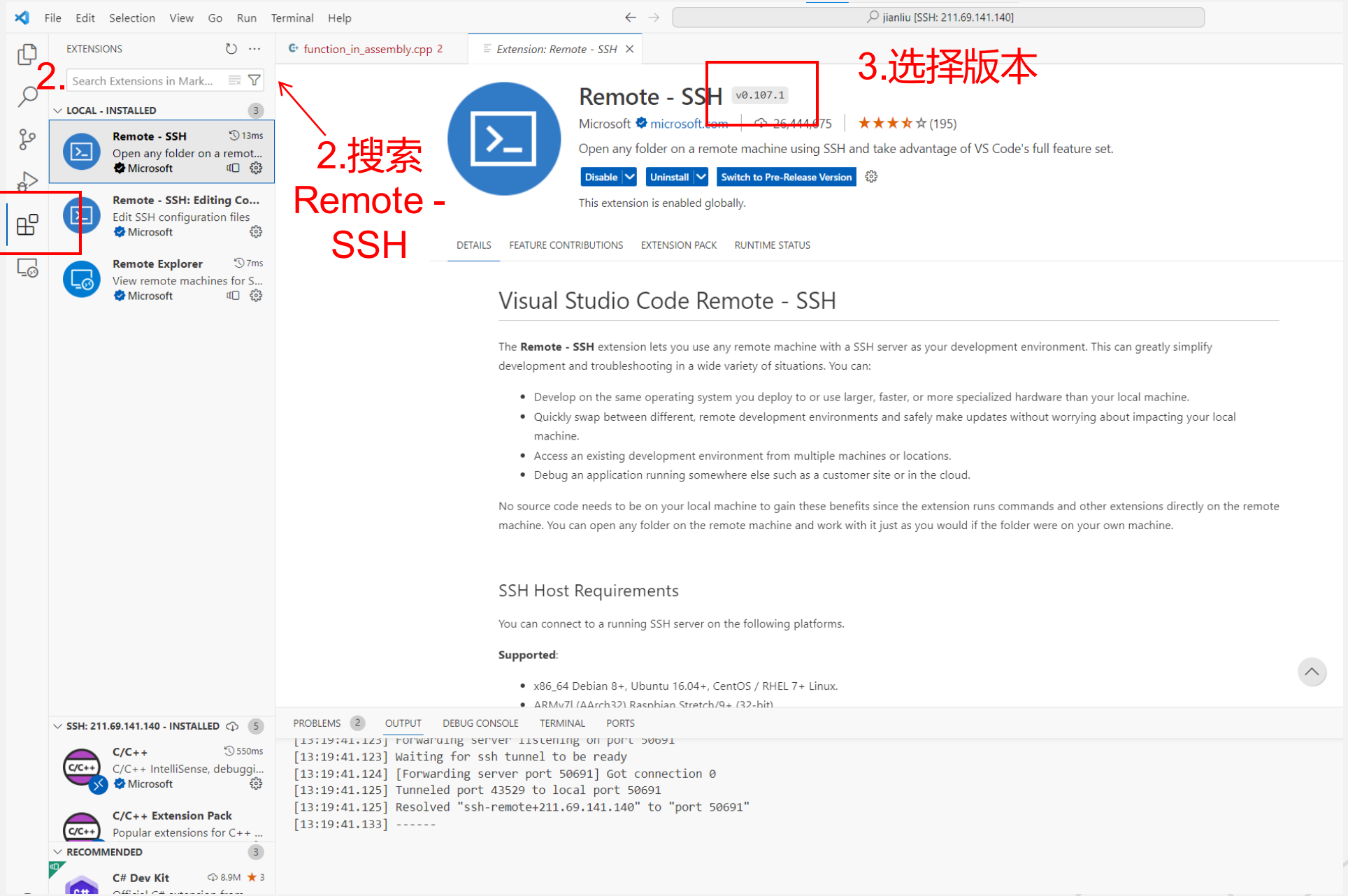
```
[jianliu@login02 ~]$ ldd --version
ldd (GNU libc) 2.17
Copyright (C) 2012 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
Written by Roland McGrath and Ulrich Drepper.
[jianliu@login02 ~]$
```

根据集群上Glibc的版本选择VSCode的版本和Remote ssh插件的版本：

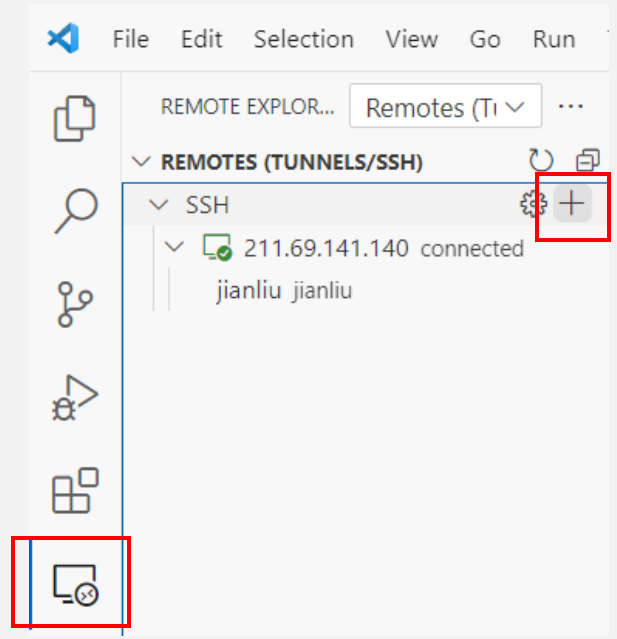
VSCode：1.58.2



2.设置流程


1.
2.
2.搜索
Remote -
SSH
3.选择版本
Remote - SSH
v0.107.1
Microsoft microsoft.com
Open any folder on a remote machine using SSH and take advantage of VS Code's full feature set.
Disable
Uninstall
Switch to Pre-Release Version
This extension is enabled globally.
DETAILS
FEATURE CONTRIBUTIONS
EXTENSION PACK
RUNTIME STATUS
Visual Studio Code Remote - SSH
The Remote - SSH extension lets you use any remote machine with a SSH server as your development environment. This can greatly simplify development and troubleshooting in a wide variety of situations. You can:
Develop on the same operating system you deploy to or use larger, faster, or more specialized hardware than your local machine.
Quickly swap between different, remote development environments and safely make updates without worrying about impacting your local machine.
Access an existing development environment from multiple machines or locations.
Debug an application running somewhere else such as a customer site or in the cloud.
No source code needs to be on your local machine to gain these benefits since the extension runs commands and other extensions directly on the remote machine. You can open any folder on the remote machine and work with it just as you would if the folder were on your own machine.
SSH Host Requirements
You can connect to a running SSH server on the following platforms.
Supported:
x86_64 Debian 8+, Ubuntu 16.04+, CentOS / RHEL 7+ Linux.
[13:19:41.123] Waiting for ssh tunnel to be ready
[13:19:41.124] [Forwarding server port 50691] Got connection 0
[13:19:41.125] Tunneled port 43529 to local port 50691
[13:19:41.125] Resolved "ssh-remote+211.69.141.140" to "port 50691"
[13:19:41.133] ------

2.设置流程



4.Remote explorer

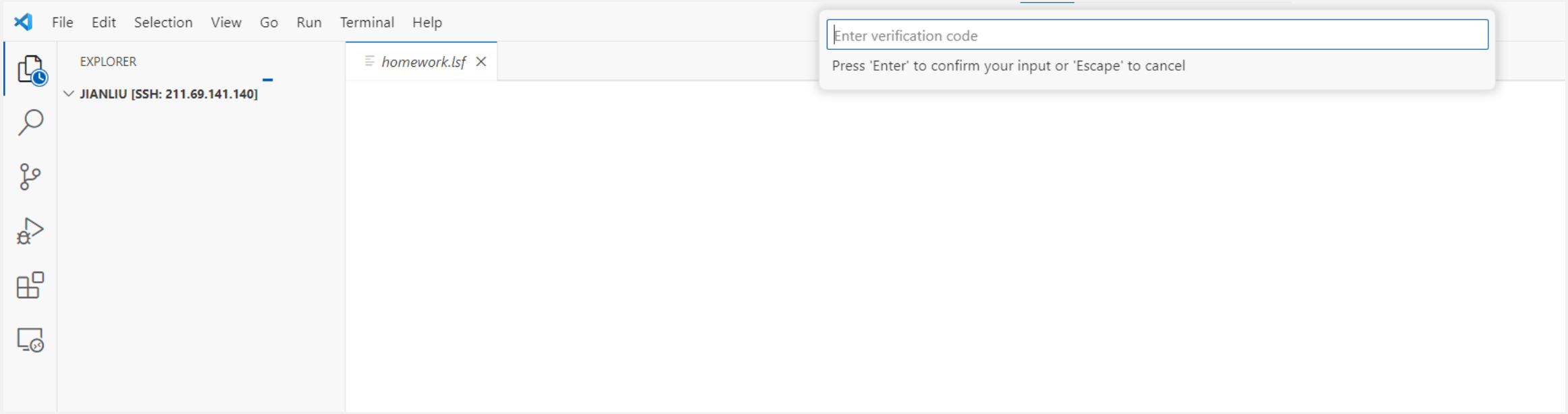
5.New remote

Remote -SSH插件默认会自动在集群安装VSCode server，连接成功！

6.选择主机类型：linux，输入端口号和主机地址

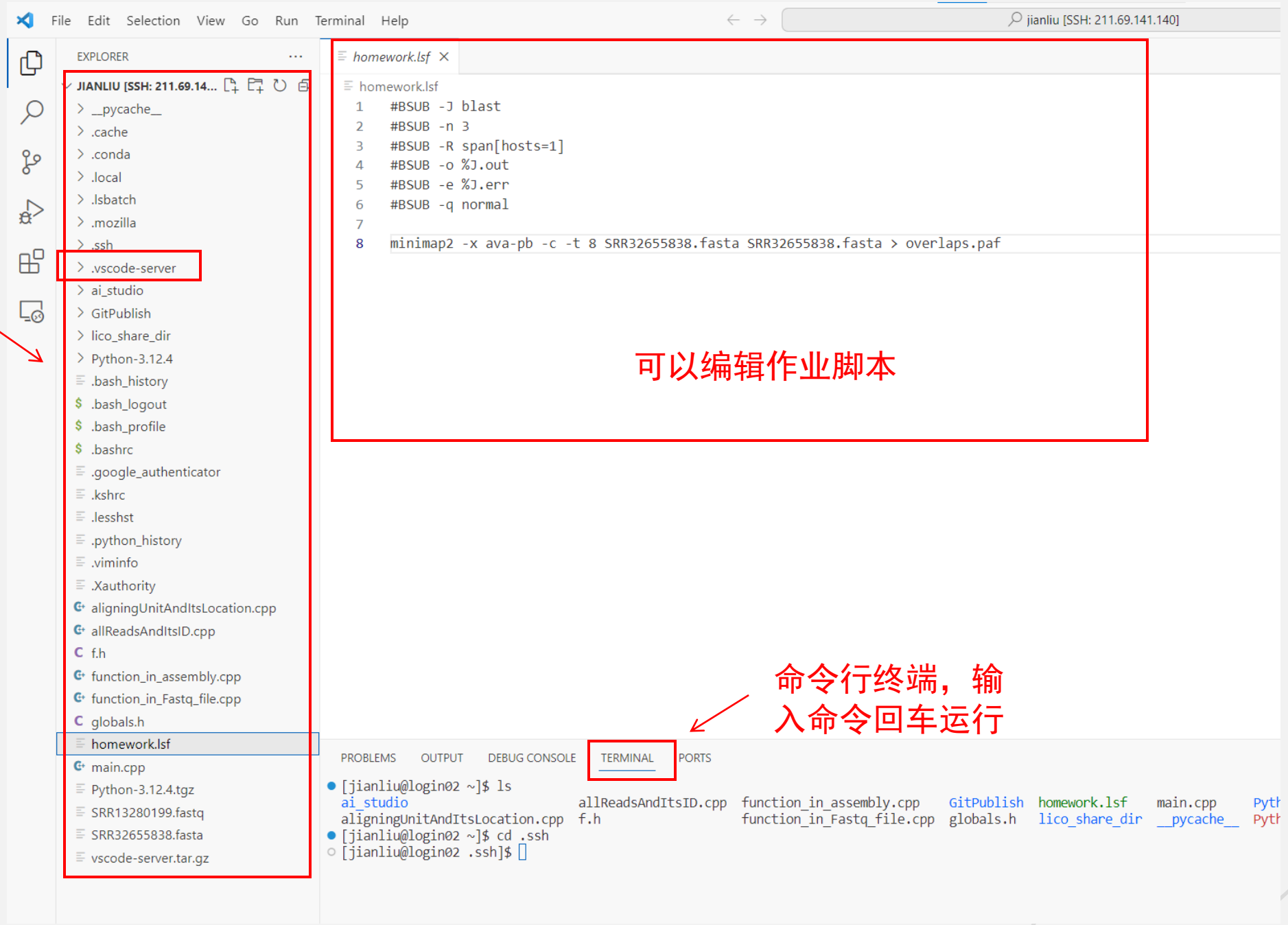
2.设置流程

登陆界面：


File
Edit
Selection
View
Go
Run
Terminal
Help
EXPLORER
JIANLIU [SSH: 211.69.141.140]
homework.lsf
Enter verification code
Press 'Enter' to confirm your input or 'Escape' to cancel

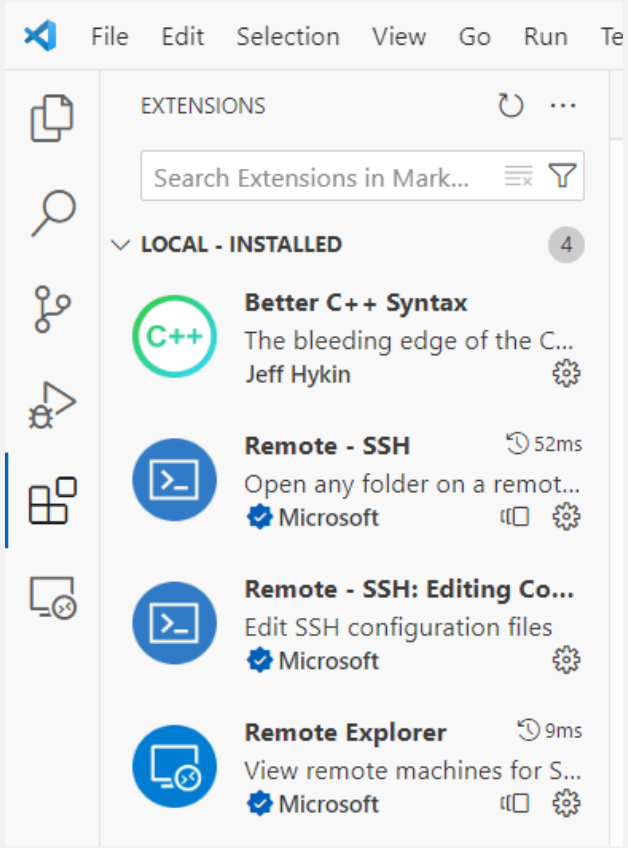
2.设置流程

集群文件列表，和windows操作方式一样，点击打开，右键删除或新建

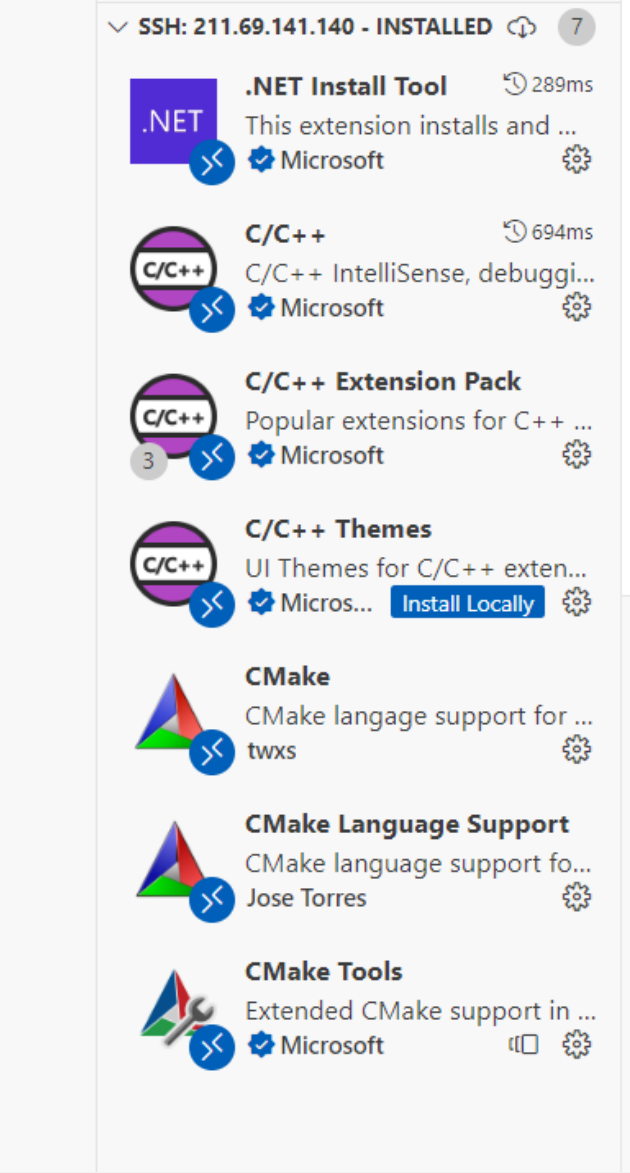


3.配置c++运行环境：下载插件

本地：



集群上：



3.配置c++运行环境：配置文件

名称不能更改

VSCode官方网址：https://code.visualstudio.com/docs/debugtest/tasks

tasks.json:

launch.json tasks.json

.vscode > tasks.json > [ ] tasks

```jsonc
{
    "version": "2.0.0",
    "tasks":
    //可以为不同场景定义多个任务
    [
        {
            "type": "cppbuild",
            "label": "genomeAssemblyAlgorithm",
            "command": "GCC", //使用GCC编译器
            "args":
            [
                "-fdiagnostics-color=always",
                "-g", // 生成调试信息
                "${file}",
                "-o", // 指定输出文件路径
                "${fileDirname}/${fileBasenameNoExtension}" //输出可执行文件路径
            ],
            "options":
            {
                "cwd": "${fileDirname}"
            },
            "problemMatcher":
            [
                "$gcc"
            ],
            "group":
            {
                "kind": "build",
                "isDefault": true  // 设为默认任务，按 Ctrl+Shift+B 直接运行
            },
            "detail": "compiler: /bin/gcc"
        }
    ]
}

```

launch.json:

launch.json tasks.json

.vscode > launch.json > ...

```jsonc
{
    "version": "0.2.0",
    "configurations":
    [
      {
        "name": "Remote C++ Debug",
        "type": "cppdbg",            // 调试器类型
        "request": "launch",          // 启动调试
        "program": "${fileDirname}/${fileBasenameNoExtension}", // 可执行文件路径
        "args": [],                   // 程序参数 (可选)
        "stopAtEntry": false,         // 是否在入口暂停
        "cwd": "${workspaceFolder}",  // 工作目录
        "environment": [],
        "externalConsole": false,     // 使用 VSCode 内置终端
        "MIMode": "gdb",              // 调试器模式 (gdb 或 lldb)
        "miDebuggerPath": "/usr/bin/gdb", // 远程机器的 gdb 路径
        "setupCommands":
        [
          {
            "description": "为 gdb 启用整齐打印",
            "text": "-enable-pretty-printing",
            "ignoreFailures": true
          }
        ],
        "preLaunchTask": "genomeAssemblyAlgorithm" // 必须与 tasks.json 中的 label 一致，确保调试前执行编译任务
      }
    ]
}
```

感谢观看！