

## به نام خدا

### ۱- نحوه استفاده از کلاستر

#### ۱-۱- نحوه دسترسی

- برای دسترسی بعد از اتصال به شبکه دانشگاه با آدرس (IP) زیر به سرور اصلی متصل می شوید:

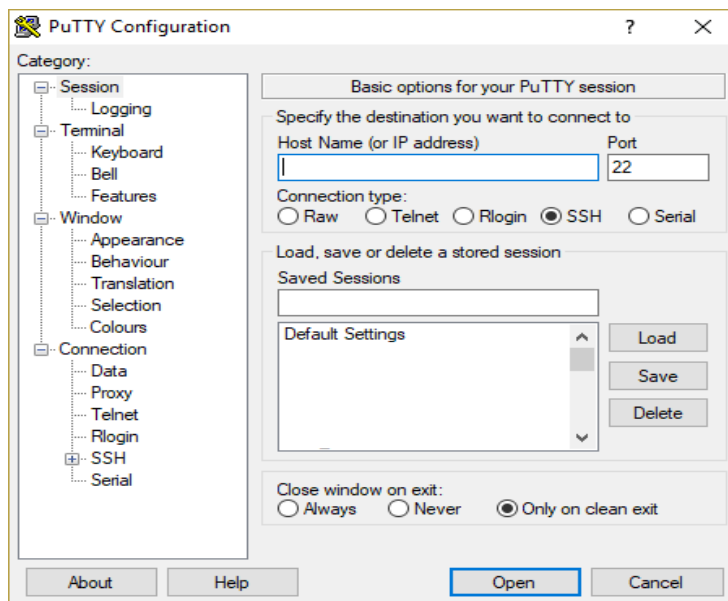
192.168.220.180

در ویندوز با استفاده از نرم افزار putty و در لینوکس با استفاده از دستور ssh می توانید به سرور متصل شوید.

نرم افزار putty را از سایت زیر می توانید دانلود نمایید:

<https://the.earth.li/~sgtatham/putty/latest/w32/putty-0.70-installer.msi>

بعد از دانلود و نصب و بعد از اجرا، پنجره زیر باز می شود:



در قسمت Host Name آدرس IP سرور کلاستر را وارد کنید. به عنوان مثال آدرس 192.168.220.180 که بعد

از وارد کردن آن می توانید با زدن کلید Open وارد سرور شوید که در این مرحله از شما نام کاربر و سپس رمز عبور

پرسیده می شود. بعد از وارد کردن آنها شما با موفقیت وارد کلاستر شده و می توانید از آن استفاده نمایید.

## ۱-۱-۱- نحوه تغییر رمز عبور برای اتصال ssh

برای تغییر رمز عبور بعد از ورود به ترمینال سرور، با زدن دستور زیر رمز عبور دسکریپشن ssh را می‌توان تغییر داد.

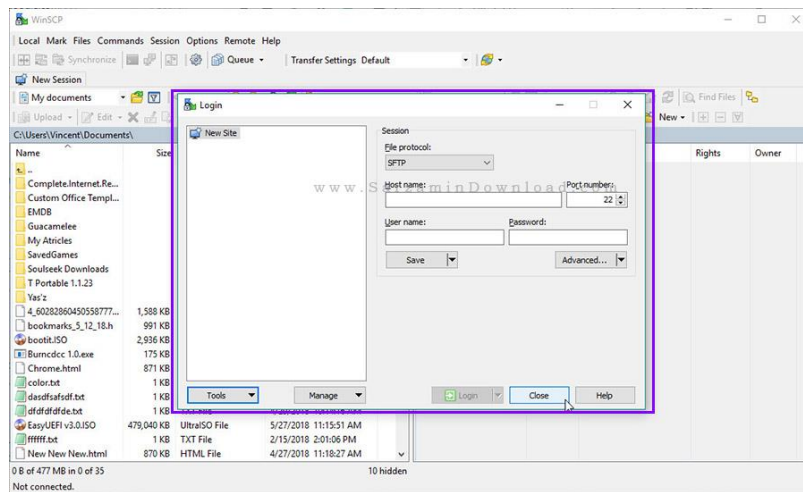
passwd

## ۱-۲- نحوه انتقال فایل‌ها از طریق سیستم عامل ویندوز

ابتدا آخرین نسخه نرم‌افزار WinSCP را با استفاده از لینک زیر دانلود کرده و سپس آن را در سیستم‌تان نصب و اجرا کنید.

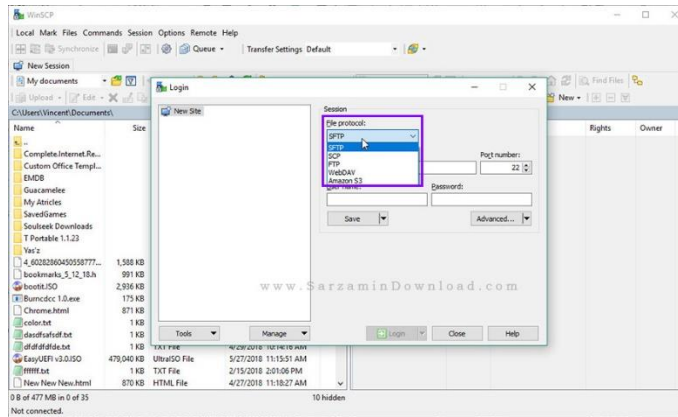
<http://sarmad.sbu.ac.ir/Downloads/winscp.exe>

به محض اجرای نرم‌افزار با پنجره کوچک مربوط به وارد کردن اطلاعات سرور جهت اتصال روبرو می‌شوید:



ابتدا پروتکل اتصال به سرور را از طریق کادر "Protocol" انتخاب کنید. بهتر است پروتکل "SFTP" را انتخاب

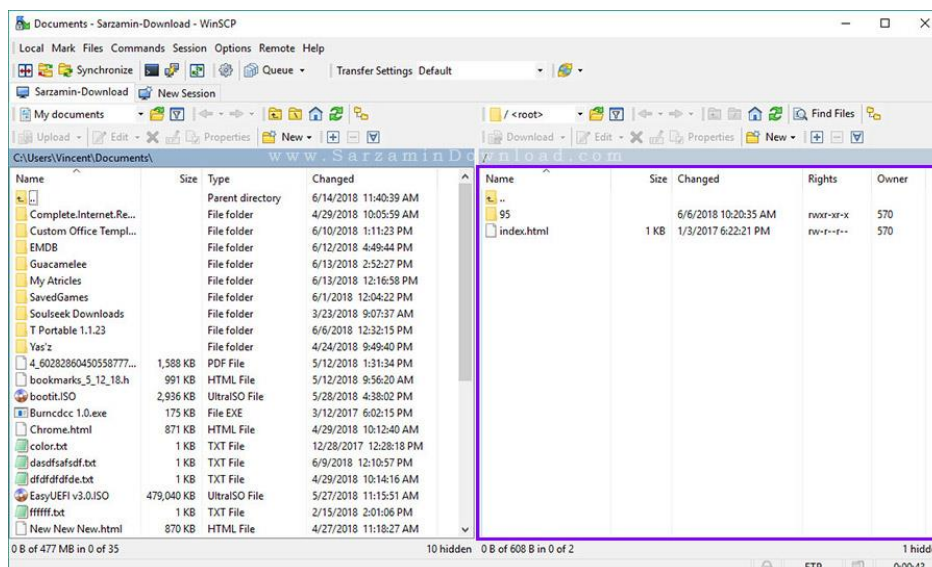
گردد تا اتصال به صورت رمزنگاری شده و امن صورت گیرد.



در ادامه آدرس سرور را در کادر "Host Name" قرار دارد برابر با 192.168.220.180 قرار دهید و همچنین پورت (Port) اتصال را نیز در کادر "Port" بنویسید (پورت پیشفرض اتصال SFTP پورت ۲۲ میباشد). همچنین نام کاربری و رمز عبور نیز برای اتصال را در کادر "User Name" و رمز عبورتان را در کادر "Password" وارد نمایید. در آخر "Save" را کلیک کنید تا این سرور در لیست سرور ها ذخیره شود. نهایتاً "Login" را کلیک کنید تا اتصال به سرور صورت گیرد.

Host name:	Port number:
<input type="text"/>	<input type="text" value="22"/>
User name:	Password:
<input type="text"/>	<input type="password"/>
<input type="button" value="Save"/>	<input type="button" value="Advanced..."/>

هم اکنون به سرور متصل شده و میتوانیم فایل های موجود در آن را مشاهده کنیم. برای انتقال فایل به سرور (آپلود) و یا دانلود فایل از سرور، تنها کافیست که فایل ها و یا پوشه های مورد نظر را از سمت راست نرم افزار به سمت چپ و یا بالعکس بکشید.



### ۳-۱- نحوه قرار دادن جابها و اجرای برنامه‌ها در سرورهای متصل به سامانه

به طور معمول در خوشه‌های<sup>۱</sup> محاسباتی، چند گره پردازشی<sup>۲</sup> وجود دارند که برنامه‌های کاربران را اجرا میکنند. بنابراین، برای این که کاربر بتواند برنامه خود را بر روی خوشه اجرا کند، باید یک گره پردازشی بیکار<sup>۳</sup> پیدا کند و سپس برنامه خود را بر روی آن گره اجرا کند. با توجه به این که این کار زمانبر و از حوصله کاربران خارج است و کاربر باید دائماً سامانه را زیر نظر داشته باشد تا فرد دیگری سامانه را در اختیار نگرفته باشد، یک گره پردازشی به عنوان گره اصلی انتخاب می‌شود تا نقش رابط بین کاربران و خوشه را به عهده بگیرد. به همین خاطر، یک برنامه بر روی گره اصلی وجود دارد که وظیفه آن مدیریت سایر گره‌ها است تا در صورت رسیدن یک درخواست اجرا (برنامه کاربر)، یک گره بیکار پیدا کند و اجرای برنامه را به آن گره محول کند. به این برنامه‌ها اصطلاحاً برنامه مدیر کار<sup>۴</sup> گفته میشود. وظیفه اصلی آن پیدا کردن منبع (گره) خالی برای اجرای برنامه‌ها است و برای این کار از یک زمانبند<sup>۵</sup> استفاده میکند تا در صورت پر بودن همه گره‌های پردازشی، برنامه‌ها را در صف قرار دهد و زمانبندی کند. با توجه به این که برنامه زمانبند از دید کاربران پنهان است، بیش از این نیازی به معرفی آن نیست. علاوه بر ساده سازی محیط استفاده از خوشه محاسباتی، برنامه مدیر کار یک نقطه قوت دیگر هم دارد و آن این است که در صورت به کار گیری آن، کاربر میتواند برنامه مورد نظر خود را به مدیر کار ارسال<sup>۶</sup> کند و بعد از آن اتصال خود به خوشه را قطع کند. در این صورت نیازی نیست که همیشه کاربر به خوشه متصل بماند تا برنامه اجرا شود. مدیر کار به صورت خودکار در پس زمینه<sup>۷</sup> برنامه کاربر را مدیریت میکند و نتایج خروجی تولید شده توسط برنامه کاربر را بر روی دیسک ذخیره میکند. با توجه به این که کاربران برنامه‌های خود را باید به مدیر کار ارسال کنند، لازم است نحوه تعامل با آن را فرا بگیرند. به طور عادی، کاربران برای اجرای برنامه‌های خود یک دستور را در لینوکس تایپ میکنند. برای ارسال برنامه به مدیر کار یک اسکریپت<sup>۸</sup> نوشته میشود که حاوی دو قسمت است. قسمت اول شامل تعدادی راهنما برای فرمان دادن به زمانبند است تا مشخصات برنامه در صف اجرا ثبت شوند. قسمت دوم حاوی دستور

---

<sup>1</sup> Cluster

<sup>2</sup> Processing node

<sup>3</sup> Idle

<sup>4</sup> Job manager

<sup>5</sup> Scheduler

<sup>6</sup> Submit

<sup>7</sup> background

<sup>8</sup> Script

اجرای برنامه است که به طور عادی در خط فرمان لینوکس نوشته میشود. یک اسکریپت نمونه به شکل زیر است. توجه کنید که قسمتهایی که با رنگ قرمز نوشته شدهاند، ثابت هستند و کاربران میتوانند بدون تغییر آنها را در

```
#PBS -q batch
#PBS -j oe
#PBS -l nodes=1:ppn=1
#PBS -N jobName
#PBS -o $PBS_JOBID.out
#PBS -e $PBS_JOBID.err
#PBS -l walltime=10:30:00
```

```
cd $PBS_O_WORKDIR
```

**Your Application command and Options**

اسکریپت‌های خود بنویسند. قسمتهایی که با رنگ آبی نوشته شدهاند، بسته به نیاز کاربر میتواند تغییر کند.

در این اسکریپت، ابتدا تعدادی راهنما (خطوطی که با # شروع می‌شوند) وجود دارند که به وسیله آنها به مدیر کار اعلام می‌شود، برنامه چه نیازها و مشخصاتی دارد. در این جا، به توضیح برخی از آنها که بیشتر کاربرد دارند، می‌پردازیم.

```
#PBS -N jobName
```

خط بالا، نام برنامه را به مدیر کار اعلام میکند. این نام دلخواه است و زمانی به کار می‌آید که می‌خواهیم از وضعیت اجرای برنامه در لیست برنامه‌های موجود در صف مطلع شویم.

```
#PBS -l nodes=1:ppn=1
```

در خط بالا کاربر برای اجرای این جاب تعداد هسته‌های پردازشی را تعیین می‌کند، با توجه به اینکه میزان منابع محدود است، مقدار انتخاب شده آن برای هر جاب، می‌تواند بین ۱ الی ۸ باشد.

```
#PBS -o $PBS_JOBID.out
#PBS -e $PBS_JOBID.err
```

دو خط بالا به مدیر کار اعلام میکند که خروجی تولید شده بر روی صفحه نمایش را در دو فایل مجزا بنویسد. به طوری که فایل با پسوند out حاوی اطلاعاتی است که به طور معمول بر روی صفحه نمایش نشان داده میشوند و فایل err. حاوی خطاهایی که در حین اجرای برنامه رخ داده است.

```
#PBS -l walltime=10:30:00
```

در خط بالا مدت زمان استفاده از منابع برای این جاب تعیین می‌گردد.

### your command

خط بالا همان دستوری است که به طور معمول کاربر در محیط لینوکس وارد می‌کند تا برنامه خود را با توجه به ورودی‌ها اجرا کند.

مثال شماره ۱) فرض کنید می‌خواهیم دستور date را به مدیر کار ارسال کنیم تا آن را بر روی یک گره اجرا کند و نتیجه (ساعت و تاریخ سیستم) را در یک فایل در مسیری که اسکریپت وجود دارد بنویسد. بنابراین اسکریپت مربوطه را به صورت زیر در یک آدرس دلخواه در پوشه کاربری خود ذخیره کنید.

```
#PBS -l nodes=1:ppn=1
#PBS -q batch
#PBS -N MyFirstExample
#PBS -o $PBS_JOBID.out
#PBS -e $PBS_JOBID.err

cd $PBS_O_WORKDIR

date > date.log
```

برای ارسال اسکریپت در صف، از دستور qsub استفاده می‌شود که باید نام اسکریپت را در ادامه این دستور قرار داد. اگر نام اسکریپت نوشته شده بالا، first\_example.sh باشد، دستور را به شکل زیر در ترمینال لینوکس اجرا می‌گردد.

```
[user_name@cluster ~]$ qsub first_example.sh
```

پس از اجرای دستور بالا، پیام زیر توسط زمانبند نمایش داده می شود، که به معنی قرار گرفتن اسکریپت جاب در صف با شماره چاپ شده در پیام است:

```
29788.cluster
```

با توجه به این که زمان اجرای این دستور کم است، می توان با استفاده از دستور ls از ساخته شدن فایل date.log اطمینان حاصل کنیم. همچنین به کمک دستور cat میتوانیم محتویات فایل را مشاهده کنیم.

```
[user_name@cluster ~]$ ls date.log
date.log

[user_name@cluster ~]$ cat date.log

Sun Mar  4 19:29:15 IRDT 2018
```

مثال شماره ۲) فرض کنید می خواهیم یک فایل نمونه از برنامه متلب را برای اجرا در سامانه قرار دهیم.

برای اجرای برنامه نمونه به پوشه متلب که پوشه کاربری قرار داده شده است، مراجعه کرده و سپس از دستور زیر استفاده می گردد:

```
[user_name@cluster ~]$ cd
[user_name@cluster ~]$ cd matlab/
```

برای ویرایش اسکریپت می توانید از دستور vim استفاده کنید

```
[user_name@cluster ~]$ vim job_Matlab2018.sh
```

بعد از استفاده از دستور بالا، محتویات اسکریپت مانند تصویر زیر نمایش داده می شود:

```

#!/bin/sh
#PBS -l nodes=1:ppn=8
#PBS -o $PBS_JOBID.out
#PBS -e $PBS_JOBID.err
#PBS -N Matlab18
#PBS -q batch
#PBS -l walltime=120:00:00
#Please config the Sanple_parfor.m for parallel compute
export MATLAB18=/share/Application/MATLAB/R2018a/
export PATH=$PATH:/usr/local/bin
cd $PBS_O_WORKDIR

# modify this section
export MFILE=sample_parfor.m

time $MATLAB18/bin/matlab < $MFILE &> ${PBS_JOBID}.Matout
~

```

با استفاده از کلید **i** شما می توانید محتویات این فایل را تغییر دهید و در انتها پس از تغییر می توانید با کلید **Esc** و سپس زدن عبارت زیر فایل را ذخیره کنید.

:w

و برای خروج از کلید **Esc** و بعد متن زیر را تایپ کنید و در انتها کلید **Enter** را فشار دهید:

:q!

در صورتی که استفاده کردن از ویرایشگر تحت لینوکس (**vim**) برای شما دشوار است می توانید با استفاده از برنامه **winscp** به پوشه متلب رفته و فایل بالا را با دوبر کلیک بر روی آن ویرایش کنید.

با توجه به نوع کار و فایل های ورودی بخش های آبی رنگ را ویرایش کنید:

1	#!/bin/sh
2	#PBS -l nodes=1:ppn=8
3	#PBS -o \$PBS_JOBID.out
4	#PBS -e \$PBS_JOBID.err
5	#PBS -N <b>Matlab18</b>
6	#PBS -q batch
7	#PBS -l walltime= <b>120:00:00</b>
8	
9	#Please config the Sanple_parfor.m for parallel compute
10	export MATLAB18=/share/Application/MATLAB/R2018a/
11	export PATH=\$PATH:/usr/local/bin
12	cd \$PBS_O_WORKDIR
13	
14	# modify this section
15	export MFILE= <b>sample_parfor.m</b>
16	
17	time \$MATLAB18/bin/matlab < \$MFILE &> \${PBS_JOBID}.Matout



همانطور که در مثال قبل توضیح داده شد، خطوط ابتدایی از شماره ۱ الی ۷، خطوطی هستند که مربوط به زمانبند و نحوه قرار دادن جاب در صف و تخصیص منابع به این جاب هستند، از اینرو، خط ۲ را می‌توانید با توجه به تعداد هسته‌هایی که برنامه شما نیاز به اجرا دارد بین ۱ الی ۸ انتخاب کنید. توجه داشته باشید که برنامه متلب به صورت پیش فرض تک هسته‌ای اجرا می‌گردد و در صورتی که برنامه نویس کد متلب از ویژگی‌های موازی استفاده کرده باشد از تعداد هسته‌های بیشتر از ۱ می‌تواند استفاده کند، بنابراین در صورتی که کد متلب را موازی نوشته‌اید، تعداد هسته‌ها یا مقدار عبارت  $ppn$  را برابر با ۱ قرار دهید تا بتوانید تعداد جاب‌های بیشتری را در سامانه قرار دهید و سایر کاربران نیز بتوانند بهترین استفاده و بهره را از سامانه داشته باشند. خط ۵، نام جاب است که می‌توانید هر عبارت دلخواه برای دنبال کردن وضعیت جاب باشد که کاربر به ازای هر جاب آن را می‌تواند تغییر دهد. خط ۷، مدت زمان استفاده این جاب از منابع پردازشی است که توسط کاربر انتخاب می‌گردد. توجه داشته باشید که مقدار آن را می‌توانید از یک دقیقه تا ۷۰۰ ساعت انتخاب کنید. همچنین لازم به ذکر است که مدت زمان انتخاب شده توسط کاربر یک زمان تقریبی از اجرای کار است کاربر می‌تواند تا دو یا سه برابر زمان تخمینی، مقدار آن را انتخاب نماید. مفهوم استفاده از این عبارت، تعیین کردن اولویت برای هر جاب با توجه به تعداد هسته‌های انتخابی کاربر است، که بر اساس آن به هر جاب اولویت داده می‌شود و از طرفی در صورتی که کاربر زمان را کوتاه انتخاب کرده باشد ولی مدت زمان اجرای جاب به بیش از آن زمان نیاز داشته باشد، زمانبند بدون توجه به اینکه جاب کاربر همچنان در حال اجرا است، پس از رسیدن به آن زمان، جاب را از وضعیت اجرا خارج می‌کند.

نکته ۱: مدت زمان اعلام شده در این بخش، فقط بعد از به اجرا رفتن برنامه ملاک است و زمان انتظار در صف از این زمان کسر نمی‌گردد.

در خط ۱۵، کاربر نام فایل متلب، کد اصلی یا Master را به درستی با رعایت حروف کوچک و بزرگ وارد می‌کند. دقت داشته باشید که نام فایل‌ها نباید شامل فاصله یا کارکترهای خاص باشد، لذا نام فایل‌های مورد استفاده را تا جای امکان بدون استفاده از فاصله و کارکترهای خاص ذخیره و در همان پوشه‌ای که این اسکریپت است، قرار دهید. توجه داشته‌باشید که در صورتی که تعداد از فایل‌های متلب دیگر در همان فایل اصلی متلب استفاده می‌گردد، همه آنها باید در این پوشه وجود داشته باشند.

نکته ۲: از آنجایی که اکثر کاربران که کد متلب را در سیستم عامل ویندوز ویرایش می کنند، دقت داشته باشند که مسیر دهی در سیستم عامل لینوکس با ویندوز متفاوت است. لذا برای یاد گرفتن نحوه آدرس دهی و تفاوت های آن می توانند از موتور جستجوی گوگل کمک گرفته و با نحوه آدرس دهی آشنا گردند.

نکته ۳: در صورتی که کاربران نیاز دارند تا اطلاعات مربوط به مقادیر متغیرها را بعد از انجام محاسبات ذخیره کنند می توانند در انتهای کد متلب از دستور زیر استفاده کنند، اما با توجه به اینکه امکان دارد حجم این فایل زیاد شود، پیشنهاد می گردد تا هر کدام از متغیرها و پارامترهای مهم را مانند دستور زیر در انتهای کد متلب خود قرار دهند.

```
save('workspace1.mat') % Will save all variables
save('workspace2.mat',X,Y,Z) % Will save the variables X, Y, Z
```

برای بارگذاری فایل به صورت کامند:

```
load('workspace1.mat') % Loads the saved variables
```

بعد از ویرایش اسکریپت و کد متلب خود و بارگذاری آنها در سامانه و در پوشه مخصوص به خود، با استفاده از برنامه putty وارد پوشه مورد نظر شده و با استفاده از دستور زیر فایل را در صف قرار می دهیم.

```
[user_name@cluster ~]$
[user_name@cluster ~]$ qsub job_Matlab2018.sh
297.cluster
```

بعد از qsub، برای اینکه وضعیت جاب را دنبال کنیم، می توانید از دستور qstat استفاده کنید. در خروجی این دستور مشاهده می گردد که کار با شماره 297 (همان شماره ای که در خروجی دستور qsub بود) نام Matlab18 را دارد (همان عبارتی که جلوی #PBS-N نوشته شده است) و در وضعیت اجرا (R) قرار دارد. در ستون سمت چپ وضعیت، سه عدد به شکل 00:01:00 نوشته شده است. این اعداد مدت زمان اجرای برنامه بر حسب هسته ساعت<sup>۱</sup> را نشان می دهند. مقدار زمان اجرا برنامه به مرور توسط مدیر کار به روز می شود. بنا به دلایلی، این به روز رسانی

---

<sup>۱</sup> هسته ساعت: به مدت زمان استفاده از یک هسته پردازشی در یک ساعت گفته می شود. به عبارتی در صورتی که برنامه ای از دو هسته به مدت یک ساعت از سامانه استفاده کند، مقدار آن با توجه به اینکه یک ساعت سپری شده است برابر با ۲ خواهد بود.

لحظه‌ای صورت نمی‌گیرد. می‌توانید با اجرای چند بار دستور qstat این موضوع را دنبال کنید. مادامی که وضعیت کار در حالت R قرار دارد، یعنی برنامه در حال اجرا است، مانند خروجی زیر:

```
[user_name@cluster ~]$ qstat

Job id          Name          User          Time Use S Queue
-----
279.cluster    Matlab18      user_name      0 R batch

[user_name@cluster ~]$ qstat

Job id          Name          User          Time Use S Queue
-----
279.cluster    Matlab18      user_name      00:01:00 R batch
```

بعد از اجرای برنامه پس از چند ساعت، با وارد کردن دستور qstat متوجه میشویم که وضعیت برنامه در حالت C به معنای پایان یافته تغییر کرده است .

در این هنگام میتوانیم با مشاهده محتویات فایل **Matout** شماره\_جاب از اجرای صحیح برنامه مطلع شویم.

```
[user_name@cluster ~]$ qstat

Job id          Name          User          Time Use S Queue
-----
279.cluster    Matlab18      user_name      00:08:43 C batch

[user_name@cluster ~]$ ls *.Matout
279.cluster.Matout

[user_name@cluster ~]$ cat 279.cluster.Matout

MATLAB is selecting SOFTWARE_OPENGL rendering.

                < M A T L A B (R) >
      Copyright 1984-2018 The MathWorks, Inc.
      R2018a (9.4.0.813654) 64-bit (glnxa64)
      February 23, 2018

To get started, type one of these: helpwin, helpdesk, or demo.
For product information, visit www.mathworks.com.

>> >> >> >> >> >> >> >>
```

نکته: اگر بنا بر هر دلیل قصد داشته باشیم که یک برنامه در حال اجرا را از صف خارج کنیم، از دستور qdel استفاده خواهیم کرد. برای این کار، شماره برنامه در حال اجرا (ستون اول دستور qstat) را باید به این دستور بدهیم.

این دستور عملاً باعث قطع اجرای دستور و کشتن<sup>۱</sup> آن می شود و وضعیت برنامه به C تغییر می یابد. دقت کنید که در این حالت ممکن است خروجی برنامه معتبر نباشد. یک بار دیگر، مثال شماره 2 را اجرا می کنیم و از این دستور برای حذف/کشتن برنامه در حال اجرا استفاده می کنیم. سپس با دستور qstat متوجه می شویم که برنامه از صف خارج شده است.

```
[user_name@cluster ~]$ qsub job_Matlab2018.sh
280.cluster.

[user_name@cluster ~]$ qstat
Job id          Name          User          Time Use S Queue
-----
280.cluster    Matlab18      user_name     00:00:00 R batch

[user_name@cluster ~]$ qdel 280

[user_name@cluster ~]$ qstat
Job id          Name          User          Time Use S Queue
-----
280.cluster    Matlab18      user_name     00:00:00 C batch
```

#### ۴-۱- نحوه دسترسی گرافیکی به کلاستر در سیستم عامل ویندوز

در برخی از برنامه ها نیاز خواهد بود تا با استفاده از محیط گرافیکی کار اجرا گردد، البته در سیستم های پردازش سریع باید کار به صورت خط فرمان اجرا گردد تا بتوان از نهایت سرعت اجرا، اطمینان از اجرای آن با توجه به اولویت و زمان بندی مناسب استفاده گردد. همچنین برای اجراهای طولانی استفاده از بخش قبل توصیه می گردد. لذا استفاده از این بخش تنها برای مدت زمان کوتاه و در بخش هایی که نیاز به انجام عملیات های تعاملی است پیشنهاد می گردد. این کار به صورت عادی با putty امکان پذیر نیست که در ادامه به چگونگی اجرا محیط گرافیکی برنامه های linux در putty می پردازیم.

برای انجام این کار به دو نرم افزار نیاز است:

#### ۱. Putty

نرم افزار بالا را می توانید از لینک زیر دانلود کنید:

<http://dl2.soft98.ir/soft/p-q/PuTTY.0.71.x64.zip?1553672946>

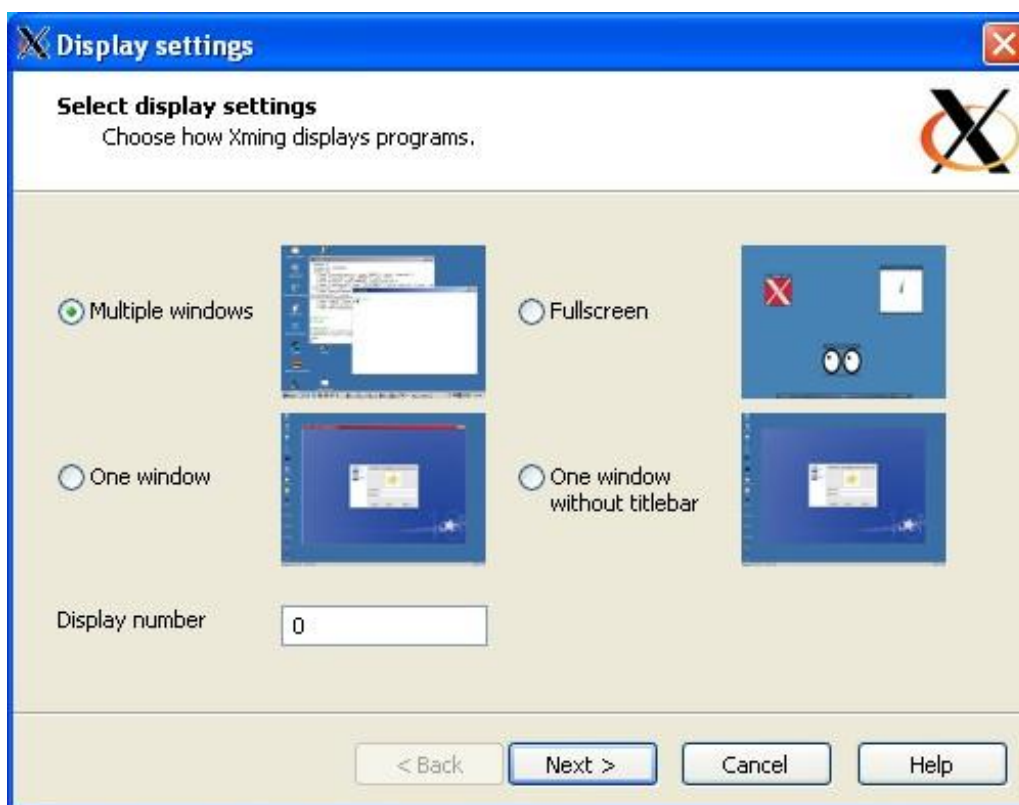
<sup>1</sup> Kill

## ۲. Xming

نرم افزار بالا را می توانید از لینک زیر یا با جستجو در گوگل و یافتن سایر لینکها دانلود کنید:

[http://sourceforge.net/project/downloading.php?group\\_id=156984&filename=Xming-6-9-0-31-setup.exe](http://sourceforge.net/project/downloading.php?group_id=156984&filename=Xming-6-9-0-31-setup.exe)

Xming یک پیاده سازی از X Window System بر روی سیستم عامل Windows است. سامانه پنجره X زیر ساخت هایی را برای اجرای یک رابط گرافیکی در سیستم عامل های مبتنی بر یونیکس فراهم می کند. بعد نصب هر دو برنامه ابتدا برنامه XLaunch را مانند تصاویر زیر اجرا می کنیم.



**Session type**

**Select how to start Xming**  
Choose session type and whether a client is started immediately.

Start no client  
This will just start Xming. You will be able to start local clients later.

Start a program  
This will start a local or remote program which will connect to Xming. You will be able to start local clients later too. Remote programs are started using PuTTY/SSH.

Open session via XDMCP  
This will start a remote XDMCP session. Starting local clients later is limited. This option is not available with the "Multiple windows" mode.

< Back   Next >   Cancel   Help

**Additional parameters**

**Specify parameter settings**  
Enter clipboard, remote font server, and all other parameters.

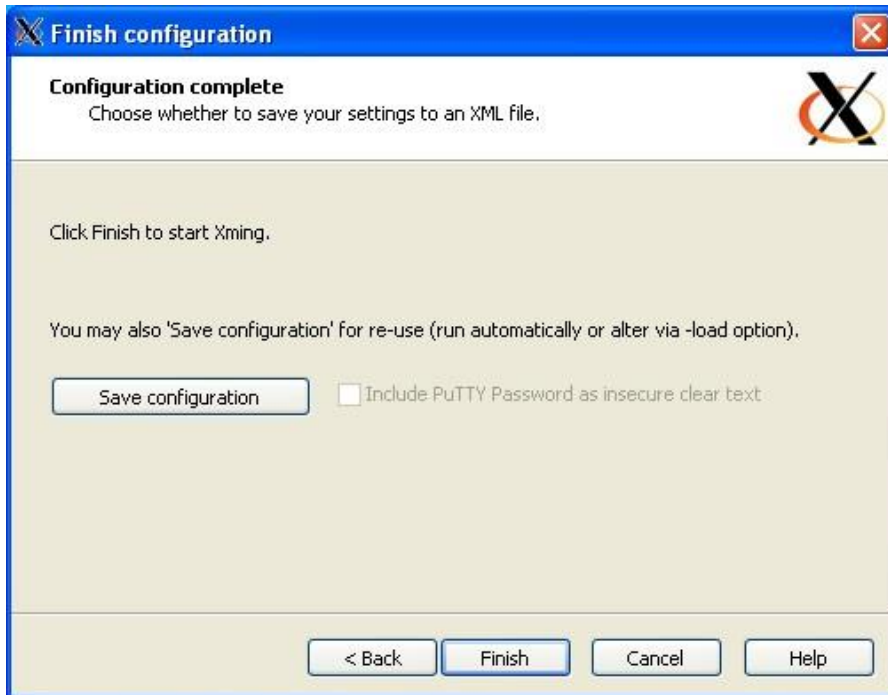
Clipboard                       No Access Control  
Start the integrated clipboard manager                      Disable Server Access Control

Remote font server (if any)

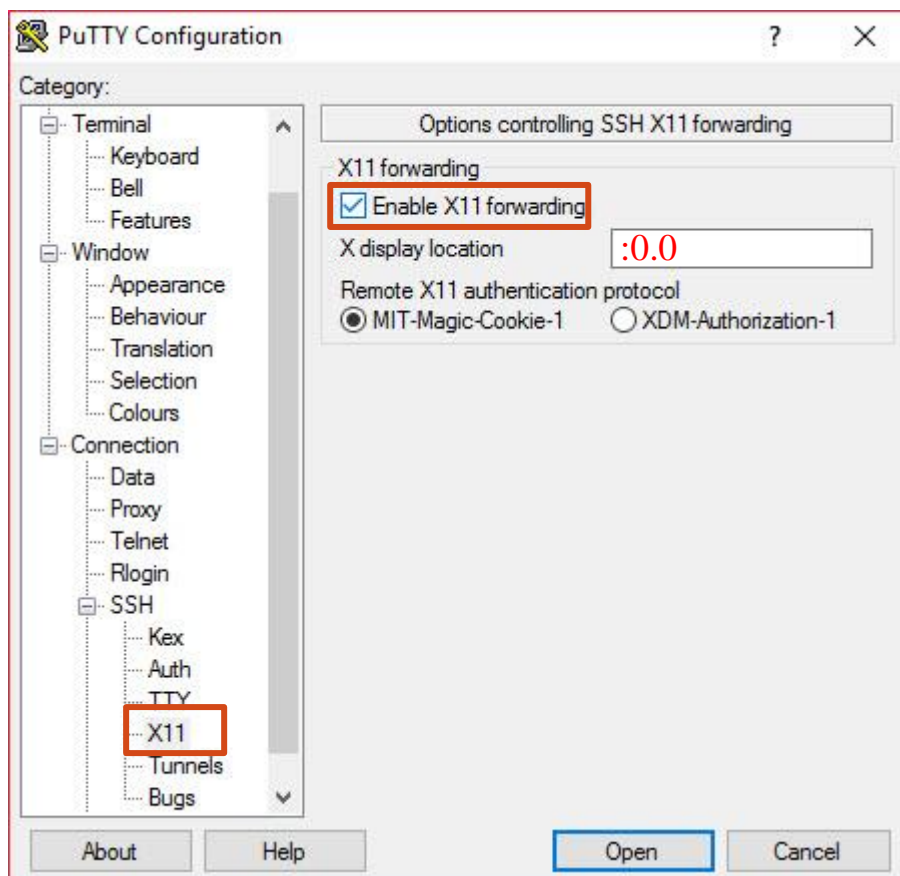
Additional parameters for Xming

Additional parameters for PuTTY or SSH

< Back   Next >   Cancel   Help



سپس برنامه putty را اجرا می کنیم و بعد از وارد کردن اطلاعات سرور مانند آی پی و پورت و ... به بخش x11 که در شکل زیر مشخص شده است مراجعه کرده و از قسمت ssh مانند شکل زیر قابلیت X را فعال کنید و سپس به سرور متصل شوید.



حال با استفاده از دستور زیر از زمانبند درخواست برقراری ارتباط با یکی از سرورهای پردازشی آزاد را ارسال کنید تا زمانبند با توجه به وضعیت سامانه به شما نوبت و اجازه ورود بدهد.

```
qsub -X -I -l nodes=1:ppn=2 -l walltime=00:10:00
```

در خط بالا، عبارت‌های آبی رنگ را می‌توانید متناسب با نوع جاب تغییر دهید. مقدار تعیین شده برای ppn، تعیین کننده تعداد هسته‌های مورد نیاز شما برای اجرای برنامه است، با توجه به اینکه در ارتباط تعاملی بیشتر زمان صرف کارهای گرافیکی و تک هسته‌ای است، بهتر است تعداد هسته‌ها را یک یا دو قرار دهید و در صورتی که قصد اجرا گرفتن دارید و نیاز به سرعت بالا و اجرای موازی دارید درخواست خود را با حداکثر ۸ هسته قرار دهید. در ادامه عبارت walltime مدت زمان برقرار بودن ارتباط با گره را نشان می‌دهد و از آنجایی که ارتباط به صورت تعاملی است، بهتر است مدت زمان کوتاه و به اندازه مورد نیاز انتخاب گردد تا در صورت وجود چندین جاب در صف، درخواست شما در صف باقی نمانده و بتوانید در کمترین زمان به سرور پردازشی متصل گردید.

توجه داشته باشید که در صورتی که منابع خالی وجود نداشته باشد، زمانبند، شما را همچنان در انتظار قرار می‌دهد و در صورت وجود منابع خالی همانند شکل زیر شما را به یکی از سامانه‌های پردازشی متصل می‌کند.

```
[user_name@cluster ~]$ qsub -X -I -l nodes=1:ppn=2 -l walltime=00:10:00
qsub: waiting for job 63.cluster.ac.ir to start
qsub: job 63.cluster.ac.ir ready
```

```
[user_name@compute-0-0 ~]$
```

به عنوان مثال پس از اتصال درخواست استفاده از سامانه برای اجرای کدهای متلب را داشته باشید، می‌توانید از

دستور زیر استفاده کنید تا برنامه متلب را پس از اتصال استفاده کنید:

```
[user_name@compute-0-0 ~]$ cd matlab
[user_name@compute-0-0 matlab]$ cat job_Matlab2018.sh
#!/bin/sh
#PBS -l nodes=1:ppn=1
#PBS -o $PBS_JOBID.out
#PBS -e $PBS_JOBID.err
#PBS -N Matlab18
#PBS -q batch
#PBS -l walltime=1:00:00
#Please config the Sample parfor.m for parallel compute
export MATLAB18=/share/Application/MATLAB/R2018a/
...
time $MATLAB18/bin/matlab < $MFILE && ${PBS_JOBID}.Matout
```

با توجه به خروجی دستور بالا، خطی را که هایلایت شده را کپی کرده و در قسمت خط فرمان مانند زیر قرار دهید.

```
[user_name@compute-0-0 matlab]$ export MATLAB18=/share/Application/MATLAB/R2018a/
```

و برای اجرای برنامه متلب دستور زیر را وارد کنید تا برنامه متلب به صورت گرافیکی اجرا گردد:



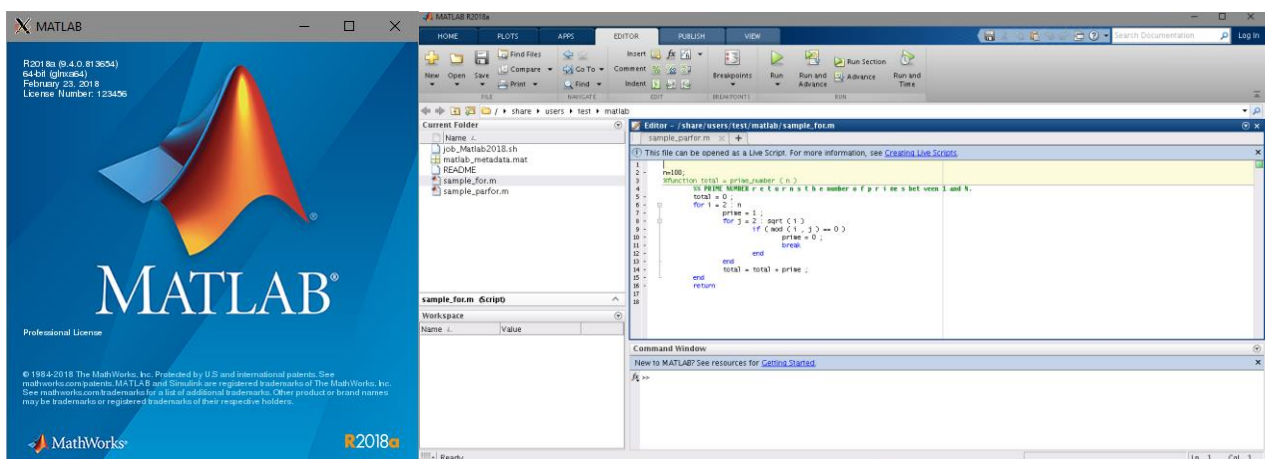
```
[user_name@compute-0-0 matlab]$ $MATLAB18/bin/matlab
```

امکان دارد پس از اجرای دستورات بالا خطای زیر صادر گردد:

```
PuTTY X11 proxy: unable to connect to forwarded X server: Network error: Connection refused
```

خطای بالا، نشان دهنده آن است که برنامه xlaunch به درستی اجرا نشده و یا firewall در سیستم شما اجازه دسترسی برنامه به این ارتباط را نمی‌دهد.

پس از آنکه برنامه به درستی اجرا گردید، شما به مدت زمانی که برای walltime تعیین کرده‌اید، می‌توانید از این برنامه استفاده کنید. توجه داشته باشید که اجرای برنامه‌ها به این روش، کمی تاخیر را به همراه دارد، که امری کاملاً طبیعی است.



در انتها پس از اجرای برنامه، برای خروج، می‌توانید از دستور exit استفاده کنید.

روش دوم، در صورتی که دسترسی به اکانت VNC دارید، می‌توانید با ابزار VNC به سامانه متصل شوید و در نهایت بعد از اتصال و باز کردن ترمینال (Terminal)، ادامه دستورات را از بخش ۱-۴-۱ دنبال کنید. به هر حال روش بالا (استفاده گرافیکی و اجرای برنامه‌های گرافیکی) به هیچ وجه پیشنهاد نمی‌گردد و روش بیان شده در بخش ۱-۳، به عنوان راه حل و استفاده بهینه از سامانه‌های پردازشی پیشنهاد می‌گردد. چرا که در استفاده به صورت گرافیکی، از سامانه پردازشی استفاده بهینه انجام نشده و برای کاربر الزامی است تا معادل دستورات گرافیکی را به صورت command پیدا کند و در نهایت برای انجام کارهای گرافیکی نتیجه را از سامانه خارج کرده و در رایانه خود برای تولید خروجی استفاده نماید.