

Hinweise zum Mathe-CIP-Pool

Stand: 24. Oktober 2020

Wir empfehlen Ihnen die Installation einer Entwicklungsumgebung auf Ihrem eigenen Rechner oder Laptop, z.B. Code::Blocks (Windows, Linux) bzw. XCode (Mac). Hinweise dazu finden Sie auf der Webseite der Veranstaltung unter der Adresse https://www.igpm.rwth-aachen. de/cpp. Eine solche lokale Installation hat den Vorteil, dass sie auch ohne eine stabile Internetverbindung funktioniert und die Benutzung der Regel komfortabler ist als das Einloggen auf einem Server.

Alternativ können Sie aber auch den Rechnerpool der Mathematik nutzen, indem Sie sich von zuhause dort einloggen. Aktuelle und allgemeine Hinweise zur Benutzung des Rechnerpools der Mathmatik (Mathepool) erhalten Sie unter der Adresse https://elearning.igpm.rwth-aachen.de/mathepool.

Die Rechner im Mathepool (CIP-Pool Raum 242 im Hauptgebäude bzw. CIP-PoolRaum 003 in der Pontdriesch) laufen mit dem Betriebssystem LINUX. Aufgrund der Pandemiesituation können die Rechner des Mathepools z.Zt. nur *remote* (d.h. per Fernzugriff von zuhause aus) genutzt werden. Die dazu nötigen Schritte sind im Folgenden beschrieben.

1 Account

Um sich auf den Servern anzumelden, brauchen Sie einen Account beim Mathepool der Mathematik (auch CIP-Pool genannt).

Falls Sie in den letzten Semestern als Mathematikstudierende das Begleitpraktikum belegt haben, sollten Sie einen solchen Account bereits besitzen, andernfalls wird Ihnen kurz nach der Anmeldung zum Modul einer angelegt werden.

Um dann für diesen Account ein Passwort festzulegen, besuchen Sie die folgende Seite:

www.igpm.rwth-aachen.de/mathepool-pw

Dort melden Sie sich mit Ihrer TIM-Kennung an und gelangen dann auf die folgende Seite:



Bitte beenden Sie den Browser, um sich auszuloggen

Anschließend können Sie ein neues Passwort festlegen, welches Sie zur Anmeldung am Mathepool-Server benötigen.

2 Zugang

Um nun mit Ihrem Account auf den Mathepool-Server zuzugreifen, müssen Sie als erstes eine Verbindung herstellen. Da dies vom Betriebssystem auf Ihrem Rechner abhängt, folgen Sie bitte dem entsprechendem Abschnitt.

2.1 Windows

Wir nutzen unter Windows die Software MobaXterm. Diese hat eine freie Version, die alles enthält, was wir brauchen. Zum Herunterladen gehen Sie auf die folgende Webseite und wählen Sie unter 'Home Edition' 'Download Now' aus. Anschließend wählen Sie die 'Installer Edition' (grüner Kasten).

mobaxterm.mobatek.net/download.html

Hinweis Vor der Installation muss die heruntergeladene Datei entpackt werden. Dazu klicken Sie zuerst auf die Datei und wählen dann oben rechts das Feld 'Alle extrahieren' aus. Anschließend können Sie die Installation mit einem Doppelklick auf die .msi-Datei starten Falls die Datei nicht vollständig entpackt wird, erhalten Sie bei der Installation eine Fehlermeldung und die Installation schlägt fehl.

Starten Sie nach der Installation das Programm. Es sollte sich nun ein solches Fenster öffnen:



igpm

Klicken Sie dann oben links auf den 'Session'-Button, um eine neue Verbindung zu starten. Als Typ wählen Sie 'SSH' und geben unter 'Remote Host'

ein. Bestätigen Sie die Erstellung mit einem Klick auf 'OK'. Anschließend sollte sich ein neuer Tab mit der gerade erstellten Verbindung öffnen. Für die Anmeldung geben Sie nun Ihre Matrikelnummer in dem Tab ein und bestätigen Sie dies mit der Enter-Taste. Sie werden nun nach dem gerade erstellten Passwort gefragt.

Hinweis

- Bei der Eingabe des Passwortes wird dieses bewusst nicht angezeigt. Nach Eingabe betätigen Sie zur Bestätigung die Enter-Taste.
- Dieser Schritt ist nur **einmal** notwendig. Wenn Sie das Programm ein weiteres Mal öffnen, ist es ausreichend, auf der linken Seite unter 'User Sessions' auf die bereits erstellte Session zu klicken.
- Falls Sie gefragt werden, ob Sie Ihr Passwort speichern möchten, ist dies nur möglich nach Einrichtung eines sogenannten Masterpasswortes. Falls Sie den genutzten Computer mit anderen Personen teilen, wird das Speichern von Passwörtern nicht empfohlen.
- Falls die Firewall fragt, ob der Zugriff durch MobaXterm zugelassen werden soll, erlauben Sie dies.

Session settings					×
SSH Telnet Rsh Xa	Marchard Mar	D 🤌 💉 TP SFTP Serial	👰 🎽 🔇 File Shell Brows	💉 😵 🛛 ser Mosh Aws S3 V	III VSL
Basic SSH settings	athepool.rwt	username	Port 22	x V	
Advanced SSH settings	Terminal settings	Network settings	★ Bookmark settings		
	Secure Shell (S	SSH) session			
	o c	ок 🛛 😣	Cancel		



Nun sollten Sie die folgende Ansicht sehen:

Damit ist die Verbindung zum Mathepool hergestellt. Sie befinden sich nun im sogenannten Terminal (auch Shell oder Konsole genannt). Die Zeile mit dem Cursor (weißer Block) nennt sich Kommandozeile, auf der nun Linux-Befehle eingegeben werden können. Sie können nun fortfahren, wie in Abschnitt 3 beschrieben.

2.2 Linux

Unter Linux ist es nicht notwendig, weitere Software zu installieren, vorausgesetzt folgendes ist installiert:

- eine grafische Desktop-Umgebung (wie Gnome oder KDE),
- ein SSH-Client (typischerweise schon vorinstalliert).

Um sich zu verbinden, geben Sie folgendes im Terminal ein:

ssh -X <matrikelnummer>@mars.mathepool.rwth-aachen.de

Anschließend öffnet sich eine Verbindung zum Server. Das weitere Vorgehen entnehmen Sie bitte dem Abschnitt 3.

2.3 Mac OS

In Mac OS ist die Vorinstallation von XQuartz notwendig. Die Vorgehensweise hierzu wird ausführlich an anderen Stellen beschrieben, daher wird hier nicht näher darauf eingegangen.

Um eine Verbindung herzustellen, starten Sie ein Terminal und geben den folgenden Befehl ein:

ssh -X <matrikelnummer>@mars.mathepool.rwth-aachen.de

Danach öffnet sich die Verbindung zum Server und nach der Eingabe Ihres Passwortes fahren Sie fort, wie in Abschnitt 3 beschrieben.

3 Arbeiten mit Linux

Der Mathepool-Server läuft unter dem Betriebssystem Linux, was einige Vorteile fürs Programmieren mit sich bringt (z.B. sind Compiler, Debugger etc. standardmäßig bereits installiert). Allerdings entstehen dadurch gerade für Linux-Neulinge einige Herausforderungen. Im Folgenden werden wir versuchen, Ihnen das System etwas näher zu bringen.

Statt einer grafischen Benutzeroberfläche wie bei den meisten anderen Betriebssystemen kann unter Linux auch rein textbasiert in einem sogenannten Terminal gearbeitet werden. Nach einer gewissen Eingewöhnungsphase liegt der Vorteil des Terminals in einem beschleunigten Arbeiten im Vergleich zur Bedienung mit der Maus. Außderdem ist das Terminal die einzige Option, falls eine grafische Benutzeroberfläche nicht zur Verfügung stehen sollte.

Wenn Sie sich mit dem Mathepool-Server verbunden haben, befinden Sie sich automatisch im Terminal (auch Shell oder Konsole genannt). Die Zeile mit dem Cursor (weißer Block) nennt sich Kommandozeile, auf der nun Linux-Befehle eingegeben werden können. Am Anfang der Kommandozeile steht der Ordner angegeben, in dem man sich gerade befindet. Beim Starten einer Shell befindet man sich immer im eigenen Home-Ordner, abgekürzt durch das Zeichen '~'. Sieht die Kommandozeile so aus wie hier,

123456@mars:~/Documents\$

dann befindet sich der Benutzer mit Account 123456 auf dem Server mars gerade im Ordner Documents. Um dann dort etwas zu tun, muss ein Befehl auf der Kommandozeile eingegeben werden. Dies können einfachste Linux-Kommandos bis hin zum Aufruf ganzer Programme sein.

Geben Sie im Terminal auf der Kommandozeile den Befehl

codeblocks &

ein, um Code::Blocks zu starten. Es öffnet sich dann ein entsprechendes Fenster, in dem Sie arbeiten können. Der GNU C++-Compiler g++ ist ebenfalls installiert. Hinweise, wie Sie diesen Compiler in der Shell benutzen können, finden Sie in der Kurzanleitung zu C++.

Einige weitere wichtige Befehle sind im Folgenden kurz zusammengefasst:

3.1 Shell-Befehle

mkdir Name	Erstelle ein Verzeichnis Name (make directory).
rmdir Name	Entferne das (leere) Verzeichnis <i>Name</i> (remove directory).
ls	Zeige die Dateien im aktuellen Verzeichnis an (list).
cd Name	Wechsle in das Verzeichnis Name (change directory).
cd	Wechsle in das übergeordnete Verzeichnis.
cd	Wechsle ins Home-Verzeichnis.
cd -	Wechsle in das zuletzt besuchte Verzeichnis.
cp Quelle Ziel	Kopiere die Datei Quelle nach Ziel (copy).
rm Name	Lösche die Datei Name (remove). Vorsicht: Weg ist weg!
mv Name Verzeichnis	Verschiebe die Datei Name ins Verzeichnis Verzeichnis (move).
mv Name1 Name2	Benenne die Datei Name1 in Name2 um.
gedit Name	Bearbeite die Textdatei Name mit dem Editor gedit.
man Befehlsname	Zeige Hilfe zum Befehl <i>Befehlsname</i> an (manual).
g++ -o Prog Name1 Name2 etc.	Übersetze die cpp-Dateien Name1 Name2 etc. und füge
	sie zum Programm <i>Prog</i> zusammen.
kdbg Prog	Debugge das Programm <i>Prog.</i>
	5



3.2 Optionen

Die meisten der Befehle haben eine ganze Reihe zusätzlicher Optionen, mit denen man ihr Verhalten verändern kann. Die Optionen werden fast immer zwischen dem Befehl und dem nachfolgenden Dateinamen eingefügt. Um sie von Dateinamen zu unterscheiden, beginnen sie mit einem Minuszeichen. Beispielsweise zeigt 1s -1 die Dateien mit zusätzlichen Informationen (Größe, Datum etc.) an. Mit 1s -t kann man sich die Dateien nach dem Datum sortieren lassen. Optionen, die nur aus einem Buchstaben bestehen, kann man oft zusammenfassen: 1s -1t oder 1s -t1. Wenn nach einer Option noch eine weitere Angabe (wie zum Beispiel -o *Prog* bei g++) folgt, dann funktioniert das allerdings nicht.

3.3 Befehle in den Hintergrund schieben

Normalerweise nimmt die Shell erst dann wieder Befehle entgegen, wenn der letzte eingegebene abgearbeitet wurde. Bei Befehlen wie 1s ist das auch ganz vernünftig. Wenn Sie allerdings einen Editor mit Ihrem selbst geschriebenen Programm gestartet haben, bleibt das Fenster so lange blockiert, bis Sie ihn wieder verlassen haben. Damit das nicht passiert (schließlich wollen Sie ein Programm nach einer Änderung übersetzen und testen, ohne jedes mal den Editor zu verlassen), können Sie stattdessen nach dem Befehl ein & eingeben. Durch das Und-Zeichen & am Ende wird der Befehl in den "Hintergrund" verschoben, und die Shell ist sofort für den nächsten Befehl bereit.