

全国高校生プログラミングコンテスト

CHaserOnline

ステップアップヒント3

13 クライアント起動用シェルスクリプトの作成

クライアントを起動するには大変多くの引数やオプションが必要です。これらを毎回入力するのは大変なことですし、また間違いの元にもなります。

それらを解消するシェルスクリプトを作成することで簡単にクライアントを起動できるようになります。テキストエディタを利用し次のように入力してください。

なお、ユーザ名・パスワード・プロキシサーバアドレスおよびプロキシサーバのポート番号は各自の環境に差し替えて記入してください。

例1) 毎回ルーム番号を入力する例(Client01.sh)

```
#!/bin/sh
#./CHaserOnlineClient2019public003.o
    http://localhost:8080/CHaserOnline003/user/ -u cool -p cool

#./CHaserOnlineClient2019public003.o
    http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool -p cool

./CHaserOnlineClient2019public003.o
    http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/
    -u cool -p cool -x 192.168.000.000:8080
```

1行ずつ説明します。

```
#!/bin/sh
```

1行目はこれからシェルスクリプトが始まることを示します。

```
#./CHaserOnlineClient2019public003.o
```

```
    http://localhost:8080/CHaserOnline003/user/ -u cool -p cool
```

改行せずに空白
をあける

改行する

次の行は左端に「#」がついています。「#」をつけるとその行のみコメント扱いとなり、命令が書かれていても実行はしません。

コメントを外した場合は、ローカルサーバでクライアントを動かす場合のコマンドになっています。

なお、紙面の都合上改行していますが、実際には改行せずに1行で記入してください。

```
#./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool -p cool
```

改行せずに空白をあける

改行する

次の行もコメントになっていますが、自宅などプロキシサーバを経由せずにオンライン接続するコマンドです。

```
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool -p cool
-x 192.168.000.000:8080
```

改行せずに空白をあける

改行する

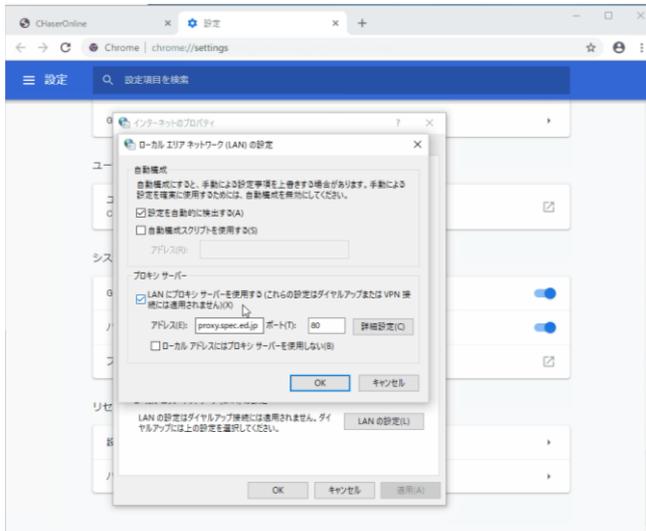
改行せずに空白をあける

指定されているプロキシアドレスとポート番号

次の行は「#」がついていません。実際に動作させるのはこの行だけとなります。学校などプロキシサーバ経由でインターネットへ接続している場合のコマンドです。

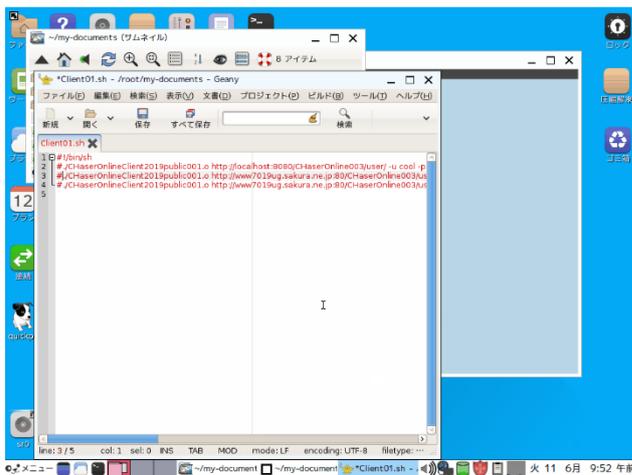
クライアントを実行する環境に合わせて「#」を付け替えることで、簡単に接続先を変更することができるようになります。

参考動画



066Crome のプロキシを調べる

<https://www.youtube.com/watch?v=OKXzfhhw1Aw>



077 シェルでプロキシのあり・なし切り替え

<https://www.youtube.com/watch?v=Cfz-p-K98DE>

例2) ルーム番号を順番に接続する例(Client02.sh)
※ルームがあいているかどうかの確認が別途必要です。

```
#!/bin/sh
for room in 1915 1923 1931
do
  #./CHaserOnlineClient2019public003.o
    http://localhost:8080/CHaserOnline003/user/ -u cool -p cool -r $room

  #./CHaserOnlineClient2019public003.o
    http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool -p cool -r $room

  ./CHaserOnlineClient2019public003.o
    http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/
    -u cool -p cool -x 192.168.30.○○○:8080 -r $room
  sleep 3
done
```

このシェルスクリプトは「接続がうまくいく」ことを前提に書かれています。
それでは1行ずつ説明します。

```
#!/bin/sh
```

1行目はこれからシェルスクリプトが始まることを示します。

```
for room in 1915 1923 1931
do
  .
  .
  .
done
```

2行目、3行目および10行目はセットで使用します。

2行目の「for」コマンドで変数「room」の中に「in」以降に列記した文字を順に代入します。変数「room」へ文字を代入するたびに、3行目から10行目の「do～done」を繰り返します。

```
#./CHaserOnlineClient2019public003.o
    http://localhost:8080/CHaserOnline003/user/ -u cool -p cool -r $room
```

4行目はローカル接続用に準備されたものになります。「-r \$room」とすることで「for コマンド」で順次変化する変数「room」をオプションに指定しています。

```

#./CHaserOnlineClient2019public003.o
    http:// www7019ug.sakura.ne.jp:80/CHaserOnline003/user/
    -u cool -p cool -r $room

./CHaserOnlineClient2019public003.o
    http:// www7019ug.sakura.ne.jp:80/ CHaserOnline003/user/
    -u cool -p cool -x 192.168. 000.000:8080 -r $room

```

改行せずに空白をあける

改行する

改行せずに空白をあける

この行はルーム番号を指定している部分以外は例1と同様になります。

```
sleep 3
```

この行は次のコマンドまでの待ち時間を設定します。数字は秒数を表すので3秒待つてループに戻ります。

Client02.sh はクライアントがある程度出来上がったときに、複数のルームで動作検証をするときに実行すると良いでしょう。

次にシェルスクリプトを実行可能な属性に変更します。

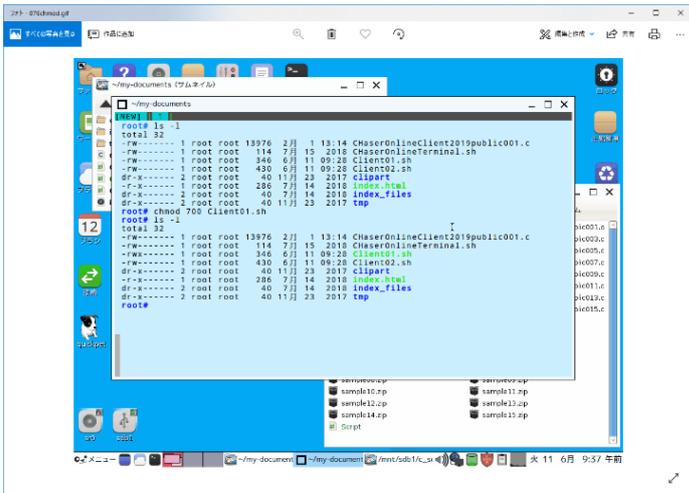
端末のコマンドラインにて次のコマンドを実行します。

```

chmod 700 Client01.sh
および
chmod 700 Client02.sh

```

参考動画



076chmod
<https://www.youtube.com/watch?v=EYEK4j0hdAs>

シェルスクリプトを実行するには端末のコマンドラインで以下のように実行します。

```
./Client01.sh
```

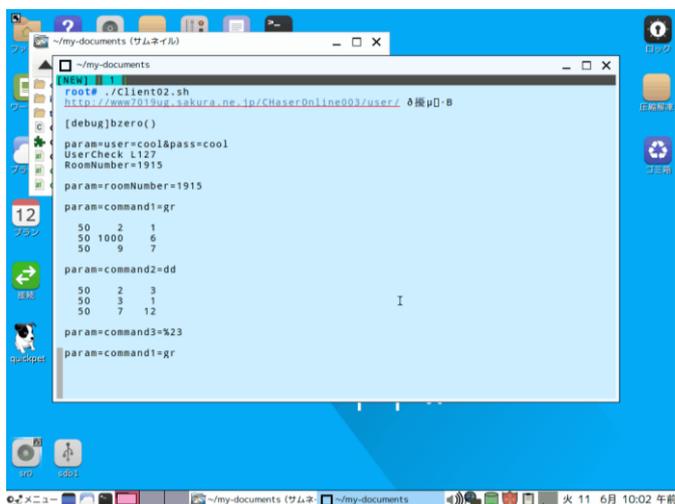
または

```
./Client02.sh
```

Client01.sh や Client02.sh などファイル名の前に「./」を入れるのを忘れないようにしてください。

なお、作品を第二次予戦に提出するときには、対戦処理の都合上シェルスクリプトからの実行が必須になります。(Windows 環境ではバッチファイルで代用してください)

参考動画



078Client02.sh の実行 2

https://www.youtube.com/watch?v=8t9q8geQL_E

例3) 自分のクライアントだけで「8人用ルーム」で対戦する例(Client03.sh)

```
#!/bin/sh
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
sleep 3
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool01 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
sleep 3
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool02 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
sleep 3
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool03 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
sleep 3
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool04 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
sleep 3
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool05 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
sleep 3
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool06 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
sleep 3
./CHaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/CHaserOnline003/user/ -u cool07 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
```

このシェルスクリプトは公開ユーザを使って8人用ルームで対戦するようになっています。
「cool」ユーザの所を各自のID・パスワードに変更して実行してください。
それでは1行ずつ説明します。

```
./ChaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/ChaserOnline003/user/ -u cool -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
```

改行せずに空白をあける

改行する

改行せずに空白をあける

一人目のクライアントを起動させています。
クライアントプログラムはサンプル3になっていますが、必要に応じて差し替えてください。ユーザは一人目として「cool」を記載してあります。各自のユーザ名・パスワードに書き換えてください。
ルーム番号は 1908 番を指定しています。必要に応じて変更してください。
行末の「&」はバックグラウンドで実行することを意味します。
こうすることでコマンドが終了していなくても次の行を実行することができるようになります。

```
sleep 3
```

この行は次のコマンドまでの待ち時間を設定します。数字は秒数を表すので3秒待つて次のコマンドに進みます。
ネットワークのスピードによってはこの秒数を調整しないと意図した順番にクライアントが接続されない場合があります。

```
./ChaserOnlineClient2019public003.o
http://www7019ug.sakura.ne.jp:80/ChaserOnline003/user/ -u cool01 -p cool -x
192.168.xxx.xxx:8080 -r 1908 &
```

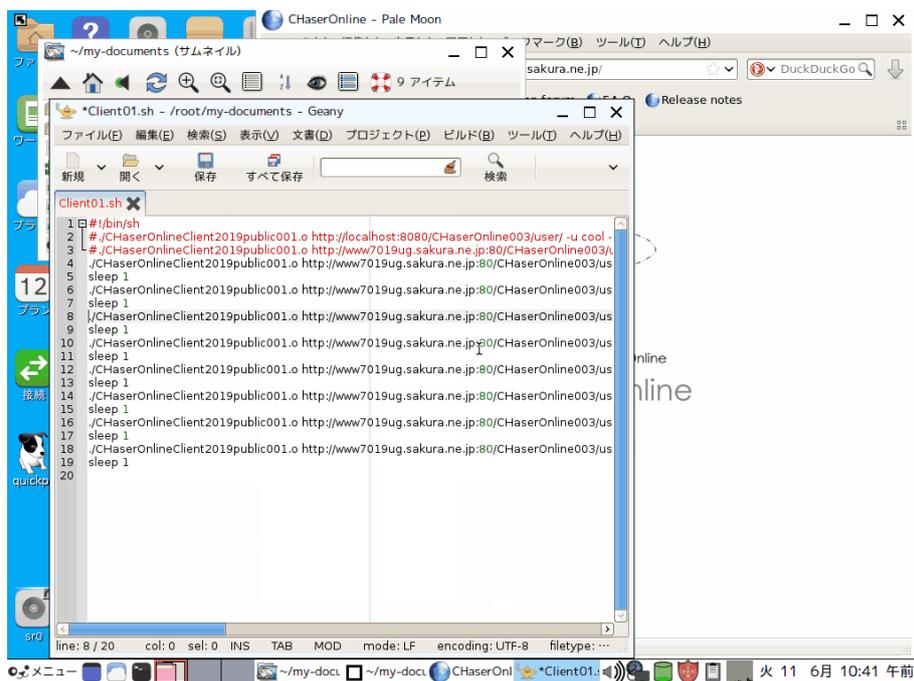
改行せずに空白をあける

改行する

改行せずに空白をあける

この行以降の7行は対戦ユーザ用となっています。
ユーザは cool01 から cool07 になっていますが、必要に応じて変更してください。

参考動画



079 自分のプログラム同士で複数人対戦

<https://www.youtube.com/watch?v=ocFvvCLCkVY>

14 サンプルプログラム 3

(1) GetReadyMove を発行するプログラム

```
/*-----  
GetReady を発行する  
-----*/  
do{  
    //printf("%n%n%ndeb191 GetReady%n");  
    strcpy(param, "command1=");  
    if(ActionCount == 9){  
        if(mode==51){  
            if(ActionReturnNumber[6]==5){  
                GetReadyMode = 1;  
            }  
        }  
        else if(mode==53){  
            if(ActionReturnNumber[6]==5){  
                GetReadyMode = 1;  
            }  
            else if(ActionReturnNumber[3]==1){  
                GetReadyMode = 3;  
            }  
            else if(ActionReturnNumber[3]==2){  
                GetReadyMode = 1;  
            }  
        }  
    }  
    else{  
        GetReadyMode = 0;  
    }  
}
```

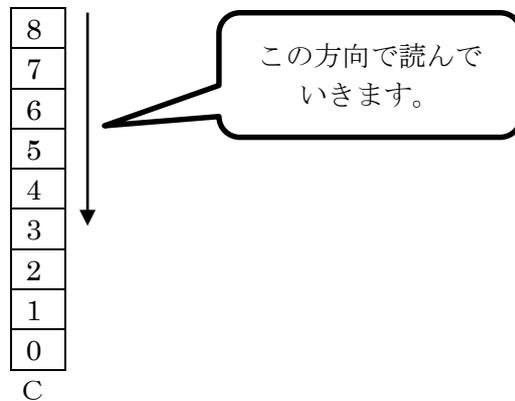
①Action の周囲情報が 9 個の時に判定する。

```
.  
.   
if(ActionCount == 9)
```

② pu3su の周囲情報の 6 つ目が化石なら上へ

```
.  
.br/>if(mode==51){  
    if(ActionReturnNumber[6]==5){  
        GetReadyMode = 1;  
    }  
}
```

「ActionReturnNumber[6]」は矢印の方向から考えて行くので「2 番目」の周囲情報をつかむことができる。



① 下記の例にて「p 1 3 s 1」の周囲情報を読んで、動作します。

```
.  
.br/>}  
else if(mode==53){  
    if(ActionReturnNumber[6]==5){  
        GetReadyMode = 1;  
    }  
    else if(ActionReturnNumber[3]==1){  
        GetReadyMode = 3;  
    }  
    else if(ActionReturnNumber[3]==2){  
        GetReadyMode = 1;  
    }  
}
```

動作例

0	1	2	3	4	5	6	7	8	C
---	---	---	---	---	---	---	---	---	---

6番が化石の時は上へ移動します。

3番がターゲットの時は左へ移動します。

3番が土の時は上へ移動します。

(2) 「Action を発行する」プログラム

```
/*-----  
Action を発行する  
-----*/  
do{  
    strcpy(param, "command2=");  
    if(count == 9){  
        if(returnNumber[5]==5){  
            mode = 51;  
        }  
        else if(returnNumber[3]==3){  
            mode = 57;  
        }  
        else if(returnNumber[1]==2){  
            mode = 51;  
        }  
        else if(returnNumber[1]==1){  
            mode = 53;  
        }  
        else if(returnNumber[0]==7){  
            mode = 53;  
        }  
        else{  
            mode = 55;  
        }  
    }  
    else{  
        if(mode == 55){  
            mode = 53;  
        }  
    }  
  
    switch(mode){  
        case 1:  
            strcat(param, "wu");  
            break;
```

```
        case 5:
            strcat(param, "wr");
            break;
        case 7:
            strcat(param, "wd");
            break;

        case 51:
            strcat(param, "pu3su");
            break;

        case 53:
            strcat(param, "pl3sl");
            break;

        case 55:
            strcat(param, "pr3sr");
            break;

        case 57:
            strcat(param, "pd3sd");
            break;

        default:
            strcat(param, "wr");
    }
    .
    .
    .
```

① GetReadyMove の周囲情報が 9 個の時

```
if(count == 9){
```

② 各移動プログラム

行動プログラム	解説
if (returnNumber[5]==5) { mode = 51; }	右が化石なら pu3su (上にアイテムを置き右に行く)
else if(returnNumber[3]==3){ mode = 57; }	左がアイテムなら pd3sd (下にアイテムを置き左に行く)
else if(returnNumber[1]==2){ mode = 51; }	上が土ならば pu3su (上にアイテムを置き右へ行く)、ターゲットなら pl3sl (左にアイテムを置き上へ行く)
else if(returnNumber[1]==1){ mode = 53; }	
else if(returnNumber[0]==7){ mode = 53; }	左上が化石なら pl3sl (左にアイテムを置き上に行く)
else{ mode = 55; }	その他は pr3sr (右にアイテムを置き下へ行く)
else{ if(mode == 55){ mode = 53; }	周囲情報が9個以外の時

③ put3&search の追加

```

case 51:
    strcat(param, "pu3su");
    break;

case 53:
    strcat(param, "pl3sl");
    break;

case 55:
    strcat(param, "pr3sr");
    break;

case 57:
    strcat(param, "pd3sd");
    break;

```

50番台に put3&search を追加しました

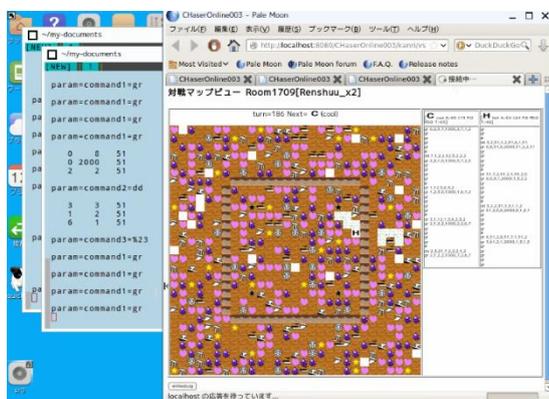

```

183 while(count>0){
184     if(i%3==0){
185         printf("%4d",returnNumber[i]);
186     }
187     else{
188         printf("%64d",returnNumber[i]);
189     }
190     i++;
191     printf("\n");
192 }
193
194 /*-----
195 Actionを發行する
196 -----*/
197 do{
198     strcpy(param,"command2=");
199     mode = 5; //下へ
200 }
201 else if(returnNumber[5]==1){ //もしも右がブロックだったら
202     mode = 5; //下へ
203 }
204 else{
205     mode = 7; //下へ
206 }
207 switch(mode){
208     case 1:
209         strcat(param,"d");
210         break;
211     case 3:
212         strcat(param,"d");
213         break;
214     case 5:
215         strcat(param,"d");
216         break;
217     case 7:
218         strcat(param,"d");
219         break;
220 }
221 }

```

Puppy750_017 サンプル 1 の改造

<https://www.youtube.com/watch?v=R6xq2ysaU6w>



Puppy750_018 サンプル 1 の改造したクライアントで対戦

<https://www.youtube.com/watch?v=8mfzPAbEMpY>

```

184 i = 0;
185 while(count>0){
186     if(i%(3*3)>0){
187         printf("%4d",returnNumber[i]);
188     }
189     else{
190         printf("%64d",returnNumber[i]);
191     }
192     i++;
193     printf("\n");
194 }
195
196 /*-----
197 Actionを發行する
198 -----*/
199 do{
200     strcpy(param,"command2=");
201     mode = 7; //下へ
202 }
203 for(houkou=1;houkou<9;houkou=houkou+2){
204     if(returnNumber[houkou]==2){ //もしも指定方向が土だったら
205         mode = houkou; //指定方向へ
206     }
207 }
208 switch(mode){
209     case 1:
210         strcat(param,"d");
211         break;
212     case 3:
213         strcat(param,"d");
214         break;
215     case 5:
216         strcat(param,"d");
217         break;
218     case 7:
219         strcat(param,"d");
220         break;
221 }
222 }

```

Puppy750_021 サンプル 1 の改造_方向を for で制御

https://www.youtube.com/watch?v=_iBzlTaqjB8

15 疲労について

疲労はパラメータの組み合わせによって表現されています。動作系列では walk 系や put3 系など、系ごとに動作得点が決まっています。

また、アイテム類も「疲労」というイメージにあわせた得点になっています。

具体的な数値は以下の表のようになります。

動作得点	
walk系	-4
put2&walk系	-10
kei系	-3
put0系	-1
put0&walk系	-2
dig系	-6
put3系	-10

アイテム類得点	
場外	-300
穴	-9
ターゲット	50
土	-20
アイテム	10
warp系	0
Client	0
化石系	150
三葉虫	50
ホワイト	-300

16 その他

・グループについて

グループによって、ユーザ毎に利用できるルームとできないルームを分類しています。第一次予戦のルームや予戦練習用ルームは選手登録をしたユーザしかアクセスできませんので注意してください。

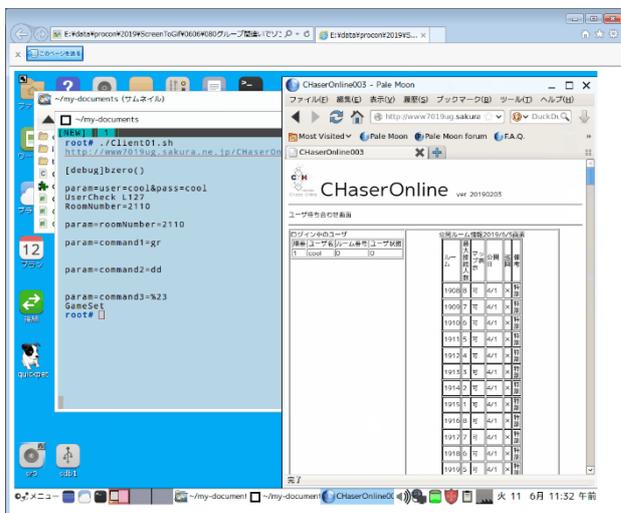
グループは以下のようになっています。

ユーザについて		
一般	昨年度までに登録されたユーザ及び公開ユーザ	使用年限なし
練習(埼玉)	今年度「ユーザ登録シート」で登録されたユーザ	人数制限なし
練習(全国)	今年度「ユーザ登録シート」で登録されたユーザ	人数制限なし
選手(埼玉)	今年度「参加申込シート」で登録されたユーザ	人数制限なし
選手(全国)	今年度「参加申込シート」で登録されたユーザ	人数制限有り
巡回	対戦ルームを監視し、ユーザを見つけると接続するユーザ	巡回区分有り

ルームについて	
一般	誰でも使えるルームで合計得点ランキングに掲載されます。 また、全国プロコン第1次予戦のボーナス得点の対象になりません。
練習(埼玉)	練習ユーザ専用のルームです。
練習(全国)	練習ユーザ専用のルームです。
予戦練習(埼玉)	選手ユーザ専用で予戦に挑む前に練習するためのルームです。
予戦練習(全国)	選手ユーザ専用で予戦に挑む前に練習するためのルームです。
予戦(埼玉)	予戦用ルームで、手動接続は禁止となっています。
予戦(全国)	予戦用ルームで、手動接続は禁止となっています。

グループ区分について		ユーザ					
		一般	練習(埼玉)	練習(全国)	選手(埼玉)	選手(全国)	巡回
ルーム	一般	○	○	○	○	○	○
	練習(埼玉)	×	○	×	×	×	○
	練習(全国)	×	×	○	×	×	○
	予戦練習(埼玉)	×	×	×	○	×	○
	予戦練習(全国)	×	×	×	×	○	○
	予戦(埼玉)	×	×	×	○	×	○
	予戦(全国)	×	×	×	×	○	○

参考動画



080 グループ間違いでゾンビ化

<https://www.youtube.com/watch?v=BmeQNpDGxKs>

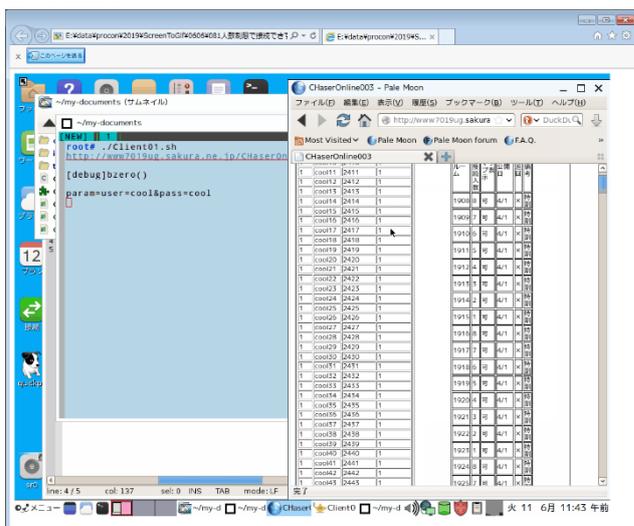
・サーバーについて

今年も第一次予戦の最終日付近ではサーバへのアクセスが集中し大変混雑することが予想されます。以前、混雑によって記録が残らないトラブルが発生しました。トラブルが起きた場合は結果の補正は行いませんのでご了承ください。

・接続人数制限について

負荷を軽減させるため、同時に接続できるユーザ数に制限を設けています。制限人数についてはサーバの負荷状態によって調整します。接続できずに「User Error」で止まってしまう場合は、少し時間を置いてからアクセスをお願いいたします。

参考動画



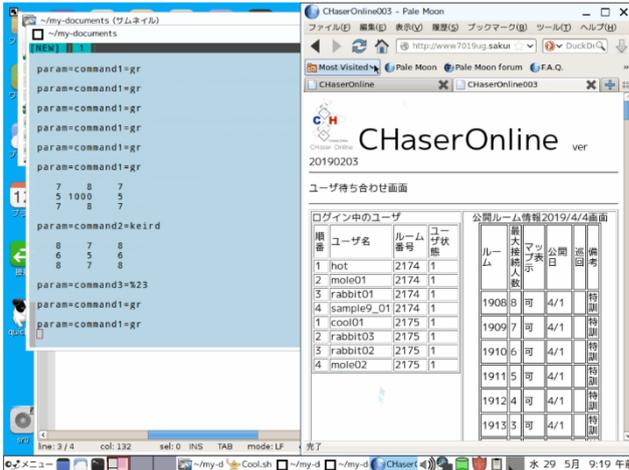
081 人数制限で接続できず

<https://www.youtube.com/watch?v=7ozZ0gLOJjs>

- ・自動巡回ユーザとの対戦方法について

自動巡回ユーザは対象ルームをシェルスクリプトを利用して監視しています。クライアントが接続したの確認すると後攻で接続します。一般用ルームや練習用ルームを使って試しておきましょう。

参考動画（巡回ユーザと対戦する例）



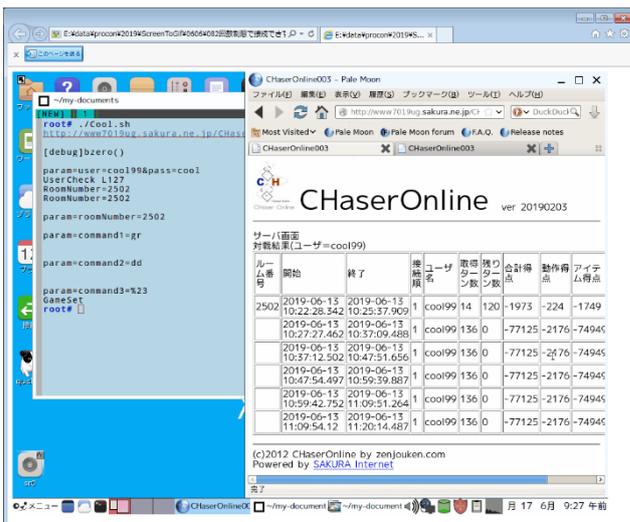
Puppy750_064 巡回ユーザとの対戦_ルーム 2175

<https://www.youtube.com/watch?v=rbX2M9ij9ZA>

- ・ルーム接続回数制限について

各ルームには、それぞれ接続回数制限があります。制限回数を超えたルームには入ることができなくなります。制限回数は個々に異なりますが非公開となっています。

参考動画



082 回数制限で接続できずゾンビ化

https://www.youtube.com/watch?v=ECOIsg_ShWE

17 対戦案内

第一次予戦が次の要領で行われます。

日 時：2019年8月16日（金） 10：00ごろ から

8月30日（金） 19：00まで

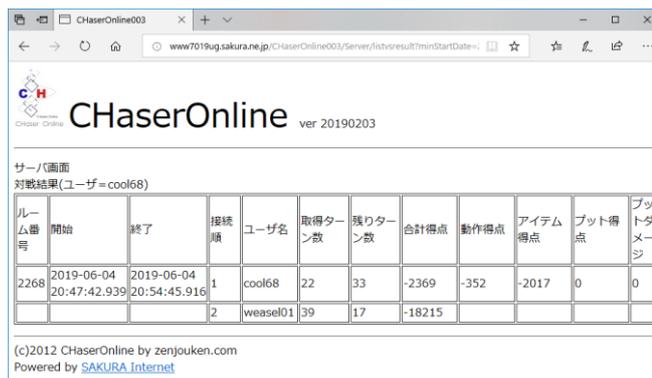
※埼玉プロコンは 8月16日（金）～9月20日（金）19：00まで

ルーム：以下のように一人用と対戦用のルームが準備されます。

全国プロコン
ルーム番号：2140～2151 一人用
2152～2159 対戦用
埼玉プロコン（競技部門）
ルーム番号：2120～2131 一人用
2132～2139 対戦用

対戦用ルームでは必ず自動巡回ユーザを後攻として対戦してください。
自動巡回ユーザとの対戦方法は予戦練習ルームなどを利用して確実にクライアントが先行として接続できるようにしておいてください。
自動巡回ユーザが後攻になったのを確認するには「サーバ」「結果表示」から「対戦結果」で表示された一覧の「接続順」を確認してください。先攻が1、後攻が2となります。

参考画像



The screenshot shows the CHaserOnline web interface. At the top, it says "CHaserOnline ver 20190203". Below that, there is a section titled "サーブ(画面)" and "対戦結果(ユーザ=cool68)". A table displays the battle results for two users: cool68 and weasel01. The table has columns for room number, start/end times, connection order, user name, turns taken, turns remaining, total score, action score, item score, put score, and damage.

ルーム番号	開始	終了	接続順	ユーザ名	取得ターン数	残りターン数	合計得点	動作得点	アイテム得点	プット得点	ブットダメージ
2268	2019-06-04 20:47:42.939	2019-06-04 20:54:45.916	1	cool68	22	33	-2369	-352	-2017	0	0
			2	weasel01	39	17	-18215				

(c)2012 CHaserOnline by zenjouken.com
Powered by [SAKURA Internet](#)

集 計：各クライアントが指定のルームで初めて取り組んだときの得点を集計します。

ユーザ ID：練習用 ID では第一次予戦ルームに入ることはできません。

選手用 ID を使って接続してください。

※事前に選手用 ID にて予戦練習ルームで動作確認をしておいてください。

注 意：ゲーム途中でクライアントが停止するなどのトラブルがあった場合は残りターン数に応じて減点されます。また、ルームへの接続回数も減点対象となることがあります。

インターネット回線のトラブルの場合はゾンビ化解消後すぐに再度接続してください。どちらを採用するかは委員会にて検討させていただきます。

クライアントは第1次予戦、第2次予戦、本戦とすべての対戦において、ゲーム開始から終了まで自動実行するようにしてください。対戦途中の手動操作はルール違反となります。また、自動実行であっても不安定・未完了なクライアントの場合などは集計に反映できない場合があります。

自分の作ったクライアントが集計に入るかどうかは全情研ホームページの「クライアントランキング」を参照してください。

CHaserOnline のステップアップヒントは以上で終了となります。皆さんの参加をお待ちしています。