

## **CG Tabs 4.0 Program**

**By: Chris George**

### **A. Introduction:**

This program is based in Microsoft Excel and is ideal for any debating tournament using a win-loss method of ranking teams, and works for any format where debates involve two teams in a room. It will of course not be useful for BP style, or if you do co-ordinate debating. While it is ideally used for bracketing, whether it is at the high school or university level, it can also be efficiently used for tournaments where no bracketing occurs. **A strong working knowledge of Microsoft Excel and bracketing is assumed.** This program is designed for maximum flexibility and can have up to seven rounds. You should note this spreadsheet does **NOT incorporate individual speaker scores** and a separate spreadsheet must be used. This is done to ensure maximum speed of bracketing and ease of use, especially with regards to data entry.

CG\_Tabs started out as a simple Excel sheet to rank teams, that has evolved over nearly 10 years with various macros used to automate many features. It also incorporates the program DrawMaker ( also written by Chris George, 1994 ) which was used exclusively for non-bracketed events and randomly assigned debate match-ups. CG\_Tabs should be the only program you ever need for any debate tournament that meets the above criteria, and is customized to allow for certain elements often featured in high school events, that tabs programs written for university events are not designed for.

**Note:** Microsoft Excel's Security feature may need to be changed to allow "macros" to be executed. If macros are disabled, it will not work!

### **B. Program Set-up:**

#### **1. Entering Team Names"**

The sheet called "TabSheet" is where school information is entered, and results will entered as well. In Column B, you enter the names of teams using school names and letters such as "Lower Canada College A", "Lower Canada College B", etc. It is crucial that schools with more than one team only differ by the last letter so that school protection can work properly. In Column C, manually enter the last names of debaters, if desired. Never delete Column C, but it can be hidden. The names are simply there to provide you with info to avoid looking it up. This data will never used by the program. In Column D, you can, if desired, enter a code name that will be used. Some tournaments exclusively use team codes to make schools, etc. anonymous. If using codes, it is this information that is pulled into all round pairings. This is the only place where you need to enter codes, team names or debater's names. The information is copied into other parts of the program as needed

You can add as many rows as you want ( up to a maximum of 138 teams ) by selecting the last row and using the **fill down function**. In general, rows should be selected at the left side of the sheet and filled down. Do not copy individual cells. As well, any rows, teams can be deleted by **selecting the entire row.**

**Note: In general, CG\_Tabs is programmed to look for data in specific places. Do not delete individual cells, change the location of information, etc. If you ever get any kind of macro run-time error, some piece of information is missing or inaccurate and should be checked carefully.**

When naming teams, you MAY NOT use a forward slash “/” in any team or code name as this will cause problems with other aspects of the program. It is suggested that you use only alphanumeric characters.

**Note:** There are different ways of using the “codes” column. Even if you do not wish make teams totally anonymous, team codes might be useful for the purposes of tracking schools or provinces debating each other. For example, for high school nationals, you can simply code teams as Ontario A, Ontario B, etc. But if you wish to use codes such as “Alberta – HK” or “Quebec – CG” you must enter this in the **Codes** column, while in the Team Name column you enter “Alberta-A”, “Alberta-B”, etc. Only the codes column is used, but the other data is actually necessary to track provinces debating provinces ( which is often avoided particularly in prepared and random rounds ).

Similarly, for a university tournaments, teams often pick their own name while identifying their institution. For example, a team may code themselves as “McGill Smoked Meat”. This information could be entered straight up in the teams column, but the program will in no way be able to reconcile whether or not it is debating another McGill team. The recommended method would be to enter “McGill A” in the teams column, the coded name in the codes column, and run the program using codes.

## **2. Entering Judges:**

Go to sheet called “Judges”. Here you will enter the names of all possible judges, starting in row 11. It is best that you enter a judge’s full name. You must also assign a rank for every judge. The maximum rank can be changed to anything you want, but integers must be used. The highest ranking is the highest number. A few things to keep in mind:

- **All judges with the same ranking are sorted randomly.**
- It is best to spread out the ranks of judges to some extent. For example, it might be easiest to give judges a rank of either 10, 15 or 20, and then you have some leeway to place people in between these different levels, or give certain people at the top higher numbers, etc.
- Judges ranks can easily be changed round to round.
- You must also specify if a judge is “available” and if so, enter “1” in the “Available” column, otherwise enter a “0”. Judges will come and go over the course of a tournament, and this can be changed round to round.
- It is easiest to enter all possible judges once at the beginning of the tournament, even if they are unlikely to judge. It is much easier to add someone’s name, mark them as unavailable and change if needed later.

You must also specify in the correct cell the last row of the sheet that contains a judge ( simply pick a very high number ) and then the number of available judges is calculated automatically. If you click on the “Sort Available Judges” button you will see the list of judges recreated beside the current list with only the available judges listed. This will help

you determine who is available for that round, and you see their ranking. If you re-run this macro, judges with the same rank will be sorted differently.

**Note:** If, after sorting judges, you see that now wish to make changes to a judge's rank or their availability, do it **ONLY with the list on the left**. You can re-sort the judges after you make changes to availability and/or ranks to see if you are satisfied with the changes. Sorting the judges does NOT place judges into the rounds.

Next, you should specify judge's conflicts ( which means that you do not want a particular judge to judge certain teams at any point ). Be sure you have first entered the names of all teams and all possible judges, then click the button "Create Judging Matrix". In the sheet called "JudgeConflicts" you will see the names of all teams ( using school team regardless of whether codes are being used ) appear down the left, and the names of all judges appear across the top. If a judge is in conflict with a team, simply enter an "x" in that particular spot in the matrix. A judge can be listed as in conflict with any number of teams. This list can be modified at any time during the tournament.

Once you create the Judges' Conflicts matrix and enter data, you can still add new judges to the list, and then add them to the conflict matrix. You will see that there is an option to clear the matrix when the macro is run. If it says "1" and you run the Judge's Conflicts macro, the entire sheet will be erased! Make you this is what you want. If it says "0", and you run the macro, only NEW judges will be added at the far right of the Judge's Conflict Sheet. By default, this is changed to "0" after it is created once to avoid accidental erasure.

### **3. Parameters in Master Sheet:**

As a general rule, you should never change the location of any cells in this sheet. The program looks for certain values in very specific locations. This is where you will tell the program how many teams there are, how many random rounds and whether codes are used. As well, parameters for using the Powerpoint presentation are defined here. You will, if desired, need to enter the path and file name of the Powerpoint template. More info on this is given later.

**The "Govs" Column is a very important.** The data in this column keeps track of how many times a team has been Government ( or Proposition ). Its importance here is based on the fact that certain columns follow this in a particular location. The default of 58 represents Column "BF", which is the 58th column. If you delete or add any columns in "TabSheet", you must change this number. This is why it is suggested you hide rather than delete any unused columns. Otherwise, do not change this number.

### **4. Other Things to Setup:**

The program is already set up with seven rounds, and five judges per room. If you have less than five judges, it is suggested that you hide the extra columns. If you have less than seven rounds, you should hide the extra rounds in "TabSheet" and "Summary".

All information ( team names, code names, and scores ) are also copied into the "Summary" sheet. This sheet is used only to print final results. You will never enter data directly in this sheet, as all data is a direct copy from "TabSheet". Hide any extra round columns if desired. You should change the header or footer using Page Setup, as this is the sheet you would print,

or PDF, and distribute. You may need to use the fill down function to ensure that all teams get copied in properly

Finally, you need to place all room numbers in the sheet “Rooms”. Enter the data starting at row 6 as required. Data will be selected from here and copied into the round sheets.

### **5. Motions:**

The sheet called “Motions” lists all of the debating motions for every round. This info is copied into the Powerpoint presentations automatically when the presentation is created. It is advisable to do this at the beginning of the tournament to save time later. If you are not using Powerpoint presentations, there is no need to put the motions in CG\_Tabs.

### **C. Creating Random Rounds:**

You can have up to four random rounds. Sometimes events will have no random rounds beyond the first. Sometimes they will have two, presumably to debate both sides of a prepared motion ( for example at high school nationals ). Other tournaments that may not even use bracketing, may have three or four random rounds ( in which case all rounds can be created with CG\_Tabs, and the bracketing options ignored ).

You will need to enter the correct number of random rounds in the “Master Sheet”. If there are two random rounds, then all teams will debate once on each side, will not appear in the same room in both rounds, and will debate two different teams. School ( or provincial ) protection is guaranteed in all cases, and this is why it is important to ensure the team names are entered accurately.

**Note:** School protection is ALWAYS guaranteed even if only one round is random.

For the “Debate Same School in Round 2” option, enter a “1” if it would be OK for a team to debate say two different teams from “Lower Canada College” in the two random rounds. If you enter a “0” each team will debate two different schools, provinces or universities as the case may be. You may wish to choose option “1” if there are a small number of teams and/or a small number of different schools or universities, or a large number of teams from certain schools or provinces. If you have three random rounds, all teams will be randomly placed in the third round, and could land on either Government or Opposition. School protection is assured and a team won’t debate a team for a second time, but they could debate a particular school ( but a different team ) again. If four rounds are used, all teams switch sides in round four as compared to round three, and the same criteria apply.

Use the “Create Random Rounds” button and all random rounds and pairings are created and placed in sheets called “Round1”, “Round2”, etc. as they apply. It is important that you do not add, delete or change the location of any columns in the round sheets. Either the team name or code is copied into the round sheets, depending of which you have decided to use.

Finally, there is a sheet called “Random” which has the data from all of the random rounds in it, with everything nicely formatted. One would use this sheet to print out and distribute at a

tournament that uses all random rounds, or perhaps if you were starting with two random rounds, you may wish to distribute both to start. This sheet does not contain any judge's info. You will notice that once you run the "Create Random Rounds" macro, there is a cell ( D14 to be precise ), which will say "Locked". This will prevent you from accidentally clicking on the random rounds macro again, which will delete all data! This is a bad thing, and this lock is provided to prevent you from accidentally doing this. Should you wish to re-do all random rounds, however, simply erase the cell with "Locked" and you will be able to re-create the random rounds from scratch.

**Note #1:** You must create all random rounds at the beginning. You cannot create one random round, and then later decide you want another and run the macro again! If you have two random rounds, you must specify this from the beginning and allow CG\_Tabs to create both from the very beginning.

**Note #2:** The program determines possible draws by "guessing" until it finds one that is accurate and meets all of the criteria. You will of course see nothing but the final result, even though the program may be trying thousands of times. It is possible, that after about 10 seconds, if no draw was found, the program will stop "guessing" and give you a message that has tried the maximum number of iterations and an incorrect draw will be saved. This incorrect draw will likely be of no use to you. It is possible that trying a couple of times will produce an accurate draw eventually. However, it is possible that the draw you requested is actually impossible based on the number of random rounds you want. If you have a lot of teams from certain schools, it may make things harder. If there are not that many teams and lots of random rounds, it will be harder. One simple thing you can do, is to tell the program that it is ok to debate the same school in round two ( as described earlier ). This will open up some combinations that were previously rejected. However, if it still doesn't work, you may simply have too many random rounds.

### **D. Assigning Judges and Judging Analysis for Random Rounds:**

**Whenever any round is created, bracketed or random, the judges are NOT automatically added.** There are several different options to add judges for rounds. If you are doing a bracketed tournament, or any tournament that uses win-loss standings, you will want to ensure an odd number of judges in every room. If you are running a tournament with all random rounds, you may wish to do something different. Here are the different judging types useful for random rounds:

#### **Judging Type "2":**

First, make sure make sure the correct round is selected, and judging type, then click "Add Judges". This option will slot judges and place an EQUAL NUMBER of judges per room, which may or may not be what you want since it could place two judges in every room. Your highest ranked judges will be slotted as chairs in random rounds, in the first column.

#### **Judging Type "4":**

This judging type puts an ODD number of judges per room, which is what you will want for tournaments using win-loss results. Each room will either have one or three judges per room,

although if there is one left over it will be slotted such that the last room will have two judges and you can decide how to manage this.

The program has the option to assign judges to ALL random rounds at once, but you should realize that the judges assignments for each random round will **NOT BE THE SAME**.

Once the judges are assigned, use the macro “Add Judging Analysis”. When this is run, any judge that is in conflict with a team will turn red, and in addition the team ( ‘Gov’ or ‘Opp’ ) in conflict is indicated. When the judges are initially added, any conflicts will be automatically resolved. At this point, you have the option to **MOVE ANY JUDGE ANYWHERE YOU WANT**. You will likely want to make some changes, and in practice you will eye ball panels and make changes. To change a judge from one room to another, **USE COPY AND PASTE TO MOVE JUDGES AROUND**. This is highly suggested as opposed to re-typing names. Once you move some judges, you will need to re-run the judging analysis ( Control – K is the short cut ) to make sure you have something that is correct.

If you are working with a bracketed round, or perhaps round 2 being a random round, you may also want to perhaps ensure a judge has not judged a team before. The program tracks this in the judging analysis in addition to checking conflicts. **If a judge has seen a particular team before, this is identified in blue**. It will tell you which of “Gov” and/or “Opp” the judge has seen and how many times. This is not likely to occur in the second random rounds, but it could. It will certainly happen often in the later bracketed rounds. The program cannot automatically minimize these instances. It is up to you to decide whether to leave things as is, or change them around. For early rounds, resolving this is highly suggested. For later rounds, it is less important, and if you resolved the repeat judging assignments earlier, you will have less issues for later and more important rounds. For smaller tournaments, it is a difficult task to get ideal judging assignments especially if there are many conflicts. Move judges around as desired. Using cut and paste is important as it ensures text is copied exactly. ( and it doesn’t matter if the conflicts are listed in red or blue with the conflict data – just copy and past as is ). Once you have made changes, run “Judges Analysis” again ( control- K ) to see if you are satisfied. It is possible you will create a conflict, or make things worse! This is normal, you change things back, or make a different change. It does take some playing around to make sure the judging assignments are ideal. When you are done with changes, you should run the “Remove Judging Analysis” and everything returns to black and all extra characters are removed.

Sometimes after judge’s assignments are announced, someone switches rooms with someone else, end up in the wrong room, etc. It happens. It is important to go back and correct the judges’ assignments in the appropriate round sheet so that there is an accurate history of who actually judged each round. This is important to ensure that judging analysis of future rounds is correct.

#### **A Note on how CG Tabs Resolves Judging Conflicts:**

When a conflict is resolved automatically, the list of available judges will be searched to find a judge, **WITH AS CLOSE OF A RANK AS POSSIBLE**, to swap with. ( who also doesn’t have a conflict ). Sometimes, you might find a judge was slotted lower than you have them

ranked, but this because the program is looking for an assignment in which there are no conflicts.

### **Tournaments with more than one Random Round:**

If a tournament has more than one random round, there is no ideal way to produce a set of pairings that easily and properly displays all the rounds judge's assignments also, but here are some thoughts on ways to manipulate the program and various printouts to help keep the tournament running smoothly. What you need to keep in mind first of all is that the program will produce judges' pairings for all random rounds that are not necessarily the same, as it is moving people in each round to avoid conflicts, etc. Here are some ways to deal with the different scenarios:

1. Consider a tournament that maybe has just two random rounds, perhaps followed by bracketing. The random rounds in the sheet "Random" can easily be printed and distributed so that students have all of their random round assignments at once. That solves the issues for the students. What I have typically done in tournaments is give judges their assignments for round one only, typically using a projector. Then I will tell all judges to come back to the judge's room, and get what may be a different room assignment for round two. This requires some travel on the part of judges, but minimizes confusion. One could also display the round 1 sheet, with the judges' assignments, then flip to round 2, and have judges copy down their round 2 assignment, which may be different. I have tried this before, and it was a complete screw-up ( on the part of the judges! ). It is best to give judges their room assignment round by round if they are potentially switching, although another option would be to print each of the two rounds spreadsheets, copy it and have it handed out to everyone. This should minimize confusion, but is potentially quite time consuming.

2. Yet another way of handling this scenario would be to do physically copy and paste the judge's assignment for round 1 into the Random Sheet. Then, you will manually manipulate the data, in an effort to ensure all judges stay in the same room for all of the random rounds. You would have to do this manually, as the program would have no way to check any type of conflict, and then you might have to tidy up the formatting if you want to print and distribute. The advantage is that you give out assignments once, and judges don't change rooms. The disadvantage is that you don't have the ability to check for conflicts, and the assignments may not be perfect, and it may take a while. Having said, this is probably what you do anyways for a tournament of this format The advantage this program is giving you is the ability to generate random rounds, well formatted, and enter data easily, and have it sorted, tabulated and formatted properly.

While CG\_Tabs isn't ideal for doing judges assignments and displaying it cleanly for tournaments that would perhaps have even more random rounds, like 3 or 4, there are several workarounds, and it is probably better than what you might be doing anyways!

### **E: Hiding or Showing Team Codes:**

If codes are used, it may be desirable to have the school names temporarily appear in the round sheet to allow you to slot judges more easily. Use the "Show Team Codes" macro in the "Master Sheet" to have school names appear in the round sheet. Each entry will now show

as “Code name / team name”. This allows you to resolve conflicts more easily. You may run Judging Analysis regardless of whether the team names are hidden or not. Once you have resolved any conflicts, use the “Hide Team Names” BEFORE creating the Powerpoint presentation or displaying with a projector.

### **F. Displaying Pairings with Powerpoint:**

This section is optional, but a very convenient way to present pairings, ideally for bracketed rounds ( or round one ). The Powerpoint template is preprogrammed to show special effects by presenting room assignments in Powerpoint. If you want the option where team matchups are with special effects, you must decide whether you want one room or two rooms to show on one slide and enter this number in the “Master Sheet”. Two template files are included for either one or two rooms, but it is crucial to **use the correct template** based on the number you selected.

Decide if you want judges assignments to show in the pairings presentation. If not, enter 0 in the appropriate cell in “Master Sheet”, otherwise enter the maximum number of judges that appear in any room. This has no affect on judge’s assignments. In practice, this feature is generally not used. It is more useful to have the chief adjudicator work on this after pairings are announced and before the round starts, and display to the judges their assignments as an Excel sheet only using a projector.

You must be very careful about changes you make in the Powerpoint template. You may change anything in the first slide, but do NOT delete it. You may also change the title in the third slide, but otherwise do make any changes to the content. Items such as “R”, “Gov” and “Opp” are crucial. You may also change the backgrounds of the slides, or the timing of slide transitions. Do not make any changes in the second slide as the round number is automatically copied in. Finally, you may not delete or add any slides to the Powerpoint templates. When you run “Create Presentation” all pairings are copied in, one or two rooms per slide, along with the pre-programmed animations. The round number and motion are copied in automatically as well. In addition, the last slide(s) shows all pairings repeated with up to 16 rooms per slide.

After the Powerpoint file is created, it needs to be saved under a different name preserving the original template. At this point, any changes may be made manually to the newly named file.

You may decide that this animation of pairings with one or two rooms per slide is unnecessary. There is a relatively new macro added called “Create SHORT Powerpoint Presentation” which ONLY adds the summary of pairings ( with up to 16 per slide ), and does not use the slides that animate and present one or two rooms at a time. You may wish to use this to save time – in fact this is much preferred for any large tournament. You can manually enter how many rooms show per slide and how many times the set of slides is copied. For example, if you had 40 rooms of debate, you might wish to do something like set a maximum of 15 slides per room, and have it copied two times. This means that the pairings will run through, and then the exact same slides will run through again to allow people to confirm their assignment, or if they missed it the first time they will hopefully find it the second time! This

macro will automatically copy all data with slide transitions, and the motion, into the Powerpoint presentation.

Should you not wish to use Powerpoint at all to present pairings, the round sheet may be projected or printed, but be sure to hide the room rank and win-loss of the teams!

### **G. Entering Results:**

You enter results on the “TabSheet”. Starting of course with round one, you need to enter a ‘1’ in the “W/L” column if the team won the round. **Do not enter any other number.** If a team lost, leave it as a zero. Also enter the judge’s scores under J1, J2, etc. **You enter the total team score and not individual scores.** All formulas are already set up to calculate round averages as well as total number of wins and overall average scores. You might wish to add a cell to calculate the sum of the W/L column to ensure that it equals the number of rooms. This helps with error checking. There is nothing worse than having for example a 40 team tournament, and the number of wins adds up to 19 instead of 20!

**Note #1:** Do not enter any data in the Summary Sheet as it is copied automatically.

**Note #2:** Needless to say, it is very important to make sure the data is entered correctly.

Having said that if you discover a mistake later in the tournament, it can be fixed. If during round 4, you find that a team’s score in round 1 was entered incorrectly, you can simply change the number and the corrected data will be what is used to carry forward. Now the bracketing assignments might have been off for rounds already completed, but fixing an error once it is discovered and moving forward is the best you can do.

### **H. Creating Bracketed Rounds:**

After you have entered all team results, you need to sort the data using the macro in “Master Sheet”. Make sure you sort up to the correct round by entering the correct number in “Master Sheet”. Teams are sorted first in descending order by wins, then by speaker points. There may be an odd case where teams are tied and you should notice this when pairing, although it may not matter at all. If you scroll all the way to right in “TabSheet”, you will find ( after the “Govs” column ) the results sorted up to and including each round, and then a column called “Pairings” where you enter the information about who debates who. This is the information you need to bracket each round. Even if you run two random rounds, do NOT delete the columns that rank teams up to round one or else all formulas will be wrong.

You can manually pair all match-ups yourself by examining the sorted data and explicitly deciding who will debate who. The top ranked room is coded by placing the lower case letter “a” beside the two teams who are to debate each other. The “a” goes in the “Pairings” column. Follow this with the letter “b”, “c” in order from highest ranked teams to lowest. There is no error checking, so if you make a mistake it will crash when you run the macro. Once you reach room 26, restart using capital letters “A”, “B”, etc. Once you reach 52 teams, start over with numbers 0,1, 2, etc. This is followed by seven characters ( which are defined and listed in Master Sheet ), making for a maximum of 69 rooms or 138 teams . The user makes all decisions on pairing, and will need to determine pull-ups potentially as well. You have full

flexibility to make any pairings you want. You just need to be sure to enter letters in the correct order, and make sure you don't miss any letters, have exactly two of each letter, etc.

**Note:** If you have just completed round 2, and need to bracket the round 3 matchups, you must enter the data in "Up to Round 2". Be careful about this. If you accidentally put the data in the wrong column, you will get a macro run-time error.

Alternatively, you can have all of the pairings done with "Quick Tab". This is a relatively new feature of CG\_Tabs, and it can now do pairings with pull-ups in any combination. Instead of manually assigning pairings, run Quick Tab and it is done automatically! You have the option for whether to use the middle or bottom team to pull up, and whether or not reseed or not. It is up to you; all four combinations work. The Quick Tab simply assigns the letters in the pairing column, and you still have full flexibility to change whatever you want, as at this time you have not yet actually created the bracketed round.

The program determines Government or Opposition based on which team has had fewer Governments, and in the case of a tie, flips a coin. No consideration is given as to how long it has been since a team has been Government. The "Govs" provides this information handily so you can pair accordingly. When pairing, you will know that whichever team had had fewer Government assignments will automatically take that position in the next round.

**Note:** It is sometimes handy to try and even out the number of times teams have been Gov or Opp early in a tournament. For example, let's say you have a large number of teams, with many teams with the same number of points after round one which was the only random round. You might want to change the pairings that are assigned to maximize the number of teams that end up with an even number of Govs and Opps after round 2. You will know that if you are pairing a team with 0 Govs vs a team with 1, that it is assured that the team with 0 Govs will be Gov to even them out.

After you have set the pairings, either manually or with QuickTab, you may ( it is optional ) wish to run the macro "Check Match-ups Before Creating Round". You should at this point have the upcoming round paired in "TabSheet", but you must not have created the round yet. This macro will create a matrix in the sheet called "OpponentMatrix" which lists every team ( and code if applicable ) and then lists the opponents from each previous rounds, as well as the match-ups that have been paired for the upcoming round. You will be looking for any team who will be debating a team from their own school, or any repeat match-up. Any repeat match-ups appear bolded in red, while any school on school match-ups appear bolded in blue. A combination of the two appears in green ( although this would be a rare case ). You can then decide whether or not to change these pairings, and then you can rerun this macro until you are satisfied with the pairings.

**Note:** Different tournaments will have different philosophies on what to change, or not. Most would say that if teams are tied on points, especially early in a tournament, then such a change doesn't affect anything and should be made as the teams are in effect tied anyways. In practice, very small differences in speaker scores may or may not reward or penalize teams if you make changes. Certainly, for later rounds in tournaments, when brackets are small, such instances are generally left as is to ensure the integrity of the bracketing. But again, this is up

to the discretion of whoever is running the tournament. This macro simply highlights issues of school vs. school matchups, or repeat matchups, which one may or may not want to change.

As mentioned, the above macro is totally optional. In any case, once you have determined your final match-ups for the next bracketed round, then run the “Create Round” macro. Ensure that the correct round number appears in the “Master Sheet”. You enter the number of the round you wish to **create**. Note that the number will be automatically updated after having created the previous rounds so this number SHOULD already be accurate. If not, correct it.

The pairings will now appear in the appropriate round sheet. The Government and Opposition is determined, and all rooms are shuffled randomly. The room numbers appear in the order that they are listed in the Rooms Sheet ( which ideally should be in alphabetical order ). In addition, the room rank ( as determined by you ) appears, and in addition the numbers of wins each team has appears. It will be clear which rooms involve pull-ups also. This information helps you assign judges.

It is advisable not to make any changes to this data, although technically you could. You could for example physically move one room assignment, both Gov and Opp, to another room. ( this may sound trivial, but it has happened before where someone for example had a broken leg and could only debate in a room where one does not have to climb stairs! ). In this case simply copy columns B to E and swap it with another room ( it is best not to change the room number, even though it is easier, since everyone will know you made a particular switch since it won't be in order ).

You could also technically make a change as to who is Government or Opposition in a particular room, although in practice this should not be necessary ( nor is it necessarily fair since the computer has flipped a coin ). But if you do, be aware the changes will not immediately be reflected in the ‘Govs’ column, but will be recalculated when the next round is run. ( although you could manually modify the Govs column yourself )

If after you create the bracketed round, you see you have made a mistake, you can change pairings and re-run the “Create Round” macro without problems. However, you will need to re-assign judges from scratch if you had already done so as the entire round sheet is erased when you do this.

## **I. Adding Judges to Bracketed Rounds:**

At this point, you can add judges in the same way you did for random rounds, although you will not use any form of random assignments. You have a couple of different options. Some types will put equal number of judges in a room, whereas some will ensure an odd number.

### **Judging Type “0”:**

This will first of all put an EQUAL number of judges per room, which may be an even number. Type “0” will “stack” the judges in the top rooms. So, it will put your best judges in the top room, the next best judges in the second room, etc. as you have them ranked. This type of philosophy of allocating judges is better suited for later rounds in a tournament.

**Judging Type “1”:**

Again, this puts an EQUAL number of judges per room, which may or may not be desirable. This is better utilized in earlier bracketed rounds, and it will “spread out” the judges. The top judges will be spread out as chairs across all rooms. So, if you have 12 rooms the top 12 ranked judges will chair all rooms in the rank order of the rooms. Remember that any judges with the same ranking are randomly sorted. Next, in this example, the next best 12 judges will be spread out in the second judge’s slot all the way through the tab, and this carries on until all judges are assigned.

**Judging Type “4”:**

**Note: THIS IS THE MOST USEFUL TYPE FOR MOST BRACKETED TOURNAMENTS.**

This type unlike the above Type “0” and “1”, will ensure an ODD number of judges per room, which is what you will generally want for tournaments based on win-loss standings. It effectively uses the method of “spreading out the judges” to assign everyone. If there is an extra judge that cannot complete an odd number, this judge will be slotted as a second judge in the last room, and you can decide how to deal with it.

This is the same judging type you may have used for random rounds. However, in order to make it work properly for bracketed rounds, **the round must first be sorted in rank order!** There is a macro on the Judges Sheet called “Sort/Scramble Round”. It will tell you whether the round is sorted or scrambled. ( Do not edit this cell! ). You should first click the button to scramble the round, and then the round sheet will display rooms in order. At this point, and only after you reach this point, you can add the judges, and the best judges will be placed as chairs in the rooms in order. Having the tab in order also of course makes it much easier for slotting judges. Once judges are slotted, changed, and conflicts resolved, click the macro again, and the tab will be put back into the SAME ORDER it was in before you sorted it.

**Judging Type “3”, or “Vinay’s Method”:**

This is a very unique and advanced technique, which was designed in large part by Vinay Mysore. It is intended for very experienced users, and if this is your first time using this program, it might be best to skip this section. In practice, it is used rarely, but when used, ideally for the last round or two of an important, and larger, bracketed tournament, it is very powerful as it allows for very sophisticated methods of slotting judges.

For bracketed rounds, what this is effectively doing is combining the two previously designed methods for bracketed rounds. When judges are “stacked”, you are putting the top three judges in the top room for example, the next best 3 in room 2, etc. When you “spread” them out, it will take your top 12 judges for example and make them the chairs in all 12 rooms in order, then fill up with the next best judges as the second chair, etc. “Vinay’s Method” allows you to specify a very specific number of rooms, presumably which you deem to be most important, and put your top judges in those rooms. It effectively breaks the tab in two, and then allows you to put different numbers of judges per room, with different methods, while always keeping an odd number. You must, however, know exactly how many judges you

have, how many rooms are in the “top” and the “bottom”, which method to use, etc. From a practical point of view, you would probably spread out judges in the top rooms, but could use either method for the bottom rooms. The following is an illustration of the data that must be entered:

CUSID Style with Odd Judges: ( Vinay's Method )					
<b>Verify Accuracy of Entries</b>					
	Method	Start	Finish	Max. Jud.	Total Jud.
Top Rooms:	1	1	6	5	30
Bottom Rooms:	2	7	11	5	25
Automatically resolve conflicts?			1		
Method 1 is spread out, Method 2 is stacked.					

What will happen in this example is that the top 30 judges will be put 5 per room in the top 6 rooms. The best 6 judges will chair, the next best six will second chair, etc. In the bottom 5 rooms, the next 25 judges will be slotted with the best 5 of that lot “stacked” into the number 7 room, on down to the bottom 5 judges in the bottom room. In practice you may want to spread out judges for the bottom rooms also. Always remember that judges with the same rank are sorted randomly.

Also, there need not be an equal number of judges per room in the top half, or bottom half. If I entered 5 as the max judges, but then only 28 judges, then the top 5 rooms would have 5 judges, but the number 6 room would have 3 judges. If I entered 27 or 29 judges, this will not work as an odd number per room is not guaranteed. You could enter as low as 17 judges. This will put 5 in the number 1 room, but only 3 in the other rooms. If you had only 15 judges for the top, then you would need to change max judges to 3. The program checks number very carefully before proceeding, and the “Verify Accuracy” will tell you if data is right before proceeding. If not, your specific error(s) are identified. You may ONLY put max judges at 1, 3 or 5. Whichever you pick, all rooms in that block will either have max judges, or that number minus two to ensure an odd number. For the bottom rooms, you once again must check the number of rooms and the max judges and then ensure total judges make sense. However, there need not be a number of judges that ensures an odd number - if there is an ( one ) extra judge he/she will be slotted and one room will have an even number.

Many cool things can be done with this. You could for example slot your top 12 judges in a hidden quarter-final, and then you might have only 1 judge per room after that. You can use the “spread” method for both the top and bottom. You could for example specify 5 judges per room for the top rooms, and then 3 per room for the second block ( and depending on numbers if may fill out bottom rooms with singles. ) If you want to manually resolve conflicts you can disable that feature by putting “0” in the box. However, while this algorithm is quite good, you can never be 100% where the conflicted judge will move to.

As you can see it is totally flexible, although perhaps sound somewhat complicated until you actually use it.

### **Summary of Judging Types:**

As mentioned, in practice, I will typically only use Judging Type “4”, and sometimes will use Judging Type “3” when I need to do something creative, which would most likely be for the final bracketed round. The types 0, 1 and 2 were algorithms that existed before types 3 and 4 were created. Types 0,1, and 2 are not quite obsolete, but are only really useful for rounds where, for whatever reason, you want an equal number of judges as opposed to an odd number.

### **Judging Analysis:**

This becomes more crucial as the event goes on. Once again, conflicts show in red and these will need to be corrected if they appear. As the tournament goes on, the bigger issues becomes judges seeing teams multiple times. Recall this will show up in blue, and you will be told which judges have seen certain teams and how many times. If it says something like “Gov-1,Opp-2”, it means that judge has seen Gov once already, and Opp twice. You may want to fix this, although it is up to decide what is acceptable. A general rule might be to minimize judges seeing the same teams in early rounds, and then when you reach later rounds, you will want your best judges to be judging the best teams, and it is best if they have seen the team no more than once already. Use cut and paste to move judges around, and then re-run the judges analysis (control-K). When you are satisfied, run the “Remove Judging Analysis”.

### **Hide / Show Judge’s Ranks:**

This button will temporarily hide the judges ranks ( by making cell contents white ) to avoid having private and sensitive displayed accidentally. The sorted judges column is also deleted. Showing ranks makes cells black once again and resorts. There is a line that indicates whether judges ranks are displayed or hidden - do NOT delete this line!

### **Display Judges Rooms:**

Often one will use the actual Excel round spreadsheet to display with a projector what rooms judges are in. By running this macro first, found in the Judges Sheet, it will ensure that no sensitive data is displayed publicly and accidently. School names are hidden ( when codes are applicable ), it is ensured the round is scrambled (if it had been sorted), judging analysis is hidden, and the columns that show room rank and team records are also hidden automatically. Judges ranks are also hidden if they were not already, just in case.

### **J – At the end of all bracketed rounds:**

When all preliminary rounds are complete, sort again using the sorting macro to determine break order and/or final standings. The ranks will automatically be copied into Column A. Be careful to notice any ties which are not differentiated, and you may want to decide how to break the tie. The ranks assigned will be copied into “Summary Sheet”. Use the “Summary Sheet” to print any final results. The win/loss and average score of each round, as well as the total number of wins and score, are printed. This page should be easily scaled to fit on one piece of paper, or at least won’t be wider than one page.

### **Hidden Break Rounds:**

The program allows you to run for example a hidden quarter-final in the last round. To do this, you CANNOT use Quick Tab. What you would do is manually pair the quarters with 1 vs. 8 as “a”, 2 vs 7 as “b”, etc. Then you manually pair the remaining part of the tab after the quarters are set.

In order to ensure that the final tab is sorted properly, i.e. the hidden-quarter finalist are in fact ranked 1 to 8, there is a new feature where you can specify a team’s break position. There is a column called “Break” at the end of the “TabSheet”. Initially all teams have an “n” in this column, i.e. the assumption is “no”, this team did not break. If teams make quarters, then enter “q” in the cells for the 8 Quarter-Finalists teams. If you sort the tab all teams with this “q” will rank as the top 8 teams, and then the rank will be “1-Quarters”, etc. regardless of whether they won the hidden quarter.

Then, if desired, and perhaps at a later time after open rounds have taken place, for any teams who make semis, put an “s”. A finalist is denoted with “f” and tournament champions as “c”. This will make sorting easy! It also has the option for “o” for octo-finals. So, in the sort tab, “c” will always be the top rank, followed by “f”, “s”, “q” and “o” in that order. For all others leave it as “n” for did not break. This is totally optional but does solve a problem that people encountered wanting to lock certain teams in the hidden break at the top of the standings. It can also be used to properly distribute final standings after all open rounds and the tournament is complete.

### **Standard Deviation:**

Some tournaments will take teams that are tied in both wins and speaker scores, and then use standard deviation to break the tie. This is a statistical measure of how far spread out the data is. A team with the same scores roughly in all rounds will have a lower standard deviation. Teams whose scores range wildly from high to low, will have a higher standard deviation. While the significance of this statistical measure is itself debatable, and even whether higher or lower is better is subject to some discussion, it is become a standard that teams which are otherwise tied will be sorted such that the team with a lower standard deviation will be ranked ahead of any team with a higher standard deviation ( of course when team tied on wins and speaker scores ). This spreadsheet DOES have a standard deviation column which is calculated and appears in both the “TabSheet” and “Summary” sheet, **but it is not automatically sorted for**, and any ties are not distinguished in the ranking. If there is a tie, the teams may or may not be sorted as you wish them to be. In this case, you may decide to change the ranks using the data in standard deviation, or use a different criteria to break the tie. In this case, you may have to manually cut and paste the rows in the “TabSheet” to put teams in the correct order. Again, remember to cut and paste or insert rows by selecting the entire row. You will then manually adjust the rank column. Of course, if you re-sort the round using the Sort Round macro, it will be put back to where it was! So, you must carefully check for any ties and decide how to deal with it.

**Tournaments using speaker scores only:**

Sometimes tournaments are run without win-loss rankings, and results solely are based on speaker scores. CG\_Tabs can still be used for these tournaments. It can be used to create the random rounds of course, but when it comes to entering results, simply leave the “W/L” columns untouched. This will mean every team will have “0” wins, and the net effect will be the tab will be sorted on speaker scores alone. If you wish, you could even hide all columns that relate to wins.

**K – Other Information:****1 – Short Cuts:**

There are several hot keys which allow you to jump easily from sheet to sheet:

- Control – M jumps to the “Master Sheet”
- Control – T jumps to the “TabSheet”
- Control – J jumps to the “Judges” Sheet
- Control – R jumps to the current round ( using the number in “Master Sheet” )
- Control – K runs judging Analysis ( ideal while assigning judges )

**2 - Other Tidbits:**

After each round is created, the various places where round numbers appear are incremented automatically. This is done to help you have the right numbers for the next round. However, should you decide after creating a bracketing round, you need to redo it, you will find that you will need to manually change the round to be bracketed. If you forget to fix this, you have a macro run-time error. Do not be alarmed, it means that some info needs to be corrected, and most often this is what it is.

**3 - Version History:****Version 4.0 ( August, 2013)**

- fixed a bug whereby it wouldn't work if the rooms and judge numbers were equal.
- added in code to copy all random rounds into a separate sheet so you could more easily print out all random rounds for distribution.
- added ( finally! ) code to quick tab ALL scenarios of pull-ups. The user has the option for middle or bottom pull-up, with or without reseeding after pulling up.
- can alphabetically sort list of room numbers.

**Version 3.9.6 ( January, 2012 )**

- added code to allow for up to 138 teams instead of 104.
- can now add to judging matrix without wiping out previous data.
- random rounds “LOCKED” to prevent accidental erasure of random rounds.

**Version 3.9 ( January, 2011)**

- added the feature to do FOUR random rounds, borrowing code from the original DrawMaker. This effectively discontinues the DrawMaker program and this sheet can be conveniently used for non-bracketing tournaments that may use as much as four random rounds.

- judging type 4 which ensures an ODD number of judges ( used for W-L events ). Can be used for bracketed rounds if tab is sorted first.

### **Version 3.8 ( March, 2008)**

- added the feature "Display Judges Rooms" which is used when you use a projector to show judges assignments and ensures all confidential information is properly hidden and scrambled.

### **Version 3.7 ( January, 2008)**

- added option to hide judges ranks.  
- round can be sorted to do judges assignments then re-scrambled.  
- added sorting feature to protect teams that make the break and it correctly do final ranks.  
- added judging type "3" that allows one to "split" the tab, and use different methods and numbers of judging assignments in each "half". This makes things very versatile but is more for advanced users and is somewhat technical to use. ( Vinay's Method )

### **Version 3.0 ( August, 2007)**

- added analysis of previous opponents to find repeat matches or school protection.  
- corrected a bug which did not properly ensure teams switched rooms in two random rounds.  
- added an option to have two rooms per slide on a Powerpoint presentation.  
- added slides at the end of the Powerpoint presentation which copied in the tab again with 15 rooms per slide.  
- added feature to ONLY show the summary of pairing with 15 rooms per slide.

### **Versions 2.2 to 2.4 ( Summer - Fall, 2006 )**

- added macros and fixed bugs along the way to include creating two random rounds, quick-tab macro for no pull-ups, added judging analysis, judges conflict matrix, and show or hide team names when using codes.

### **Version 2.0 ( January, 2006 )**

- used for the first time at the 2006 High School National Championships in Montreal.  
- added macros to do manually pair and randomly distribute bracketed rounds, assign judges, and create Powerpoint presentations.

### **Version 1.0 ( September, 2001 )**

- a simple spreadsheet with no macros, used for the inaugural OSDU Provincial Championships, which was sorted manually to determine pairings, upon which all macros were added to.

### **Credits:**

- some sections of code based on the original DrawMaker, 1994 ( Chris George ).  
- some ideas, pieces of code, and inspiration from Aaron Cragie's "CragieTab".  
- thanks to Kozo Ota and Michael Stepner for their testing and feedback over the years.