Appraisal

In this section I am going to talk about my interactions with my end user during my project, user feedback from students in my Further Maths class including how well I think my program has turned out and I will perform an overall project and testing evaluation. This will be done using my objectives that me and my end user set the program to meet and after seeing which of these I have met I will then be able to judge how well I have met these objectives by using my end user and user feedback that I have obtained. I will also use my own opinions on what I have done to do this because I have completed one of the user feedback questionnaires; so that I can use what I think about my program for this section.

Interaction with end user and other users of the program

* 19.08.11: Me and my end user met to have an initial discussion on what teaching program he wanted and a little bit about how he wanted me to implement it.
* 15.09.11: Interview with end user conducted where he decided on what he wanted the program to do and be like.
* 23.09.11: My questionnaires were given out to my Further Maths class to fill in; where I collected useful opinions and information to base my program around. My end user was there to oversee proceedings.
* 04.10.11: My objectives for the proposed system were accepted and validated with a signature by my end user.
* 12.02.12: We both decided that the problem with selecting two or more arcs that have the same arc weight was not a big problem, so we decided to leave it as it is. Also we decided that a space can indeed be used when a user enters a forename, so that they can enter their surname or middle name.
* 29.02.12: He said that the application of Kruskal’s algorithm on a network on paper that I did for the testing section was done in the correct way and he was satisfied with the way the program performed Kruskal’s algorithm.
* 30.03.12: He checked and had a go with my program; he enjoyed using it. Also the students in my Further Maths class filled in a questionnaire each to give me feedback.
* 20.04.12: My end user gave me the completely filled in end user feedback form, so that I could see what he likes and dislikes about my program.

User feedback

The user feedback has been collected from my fellow Further Maths students and me because I am a Further Maths student also; therefore it is appropriate to offer my thoughts on my program from a user’s perspective. Below is a summary of the feedback that I obtained from the students in my A2 Further Maths class along with the analysis of the responses for each question:

1. What did you like about my program and why?

The common answers to this question were that my program was a useful way of learning Kruskal’s algorithm, easy to understand, simple, easy to use and it looked professional. Others remarked that it was informative and that the timed runs were good as they added competition where I said that because of this it encouraged you to perform the algorithm faster; this happened and made us feel confident about performing Kruskal’s algorithm again seeing that most of us hadn’t done this topic for over a year.

Analysis of responses from question 1:

The responses from question 1 are very pleasing as it showed that the majority of users of my program will find it a useful way of learning Kruskal’s algorithm; which was one of my main objectives for the program. Most also found it easy to use and understand, simple and most thought it looked professional. This means that it will encourage people to use it and seeing that some thought it was informative means that two of my objectives have been met.

1. What do you think needs to be improved about my program and why?

The common answer to this question was that the Runs needed improving where I said that when doing it once it is useful and can help someone better understand the example in the revision guide but after this you don’t really need to go into the Run form again. Others suggested that to do this the arc weightings should be randomized like they are in timed runs and also different networks would help too. Another also said about the Run form that instead of saying after the run has been completed that in the exam they will have to draw the minimum connector separately from the network the program should display text that says “Now try again but instead draw the minimum connector on paper using Kruskal’s algorithm and then check your answer using the Run feature”. They believed that this would allow students to practise performing Kruskal’s algorithm for real before moving on to timed runs. Another improvement that was common amongst us four was that when there are two or more arcs that have the same arc weight that are in the minimum connector then in timed runs you have to choose a specific one of these arcs. However Kruskal’s algorithm states that if this case exists then it doesn’t matter which of these arcs are chosen and in what order; some remarked that this was frustrating because it stopped them from completing the timed run in a quick time when they were performing the algorithm correctly. Another said that I should add a glossary so that if students don’t understand a word such as “Cycle” then they can look up what it means if they are not sure. They also suggested explaining what a minimum connector is and why it is useful to the student; which I believe could be implemented in the glossary suggested. A user noticed a small issue with the program this being that users only need to enter a space for their name for it to be accepted, meaning that when the record is displayed on screen to users in the Times\_Achieved form and the Update\_Kruskal form then it would look as though the user had not entered a name. Also another student said that when the message box displaying the text “Go to next stage” the message box in the Run form covers up the EC arc. Also in the Run form a student said that once you had pressed the “Begin Run” button there was no way to exit it and therefore you had to wait until the Run had finished to do something else. The same person said that they believed that the stopwatch displayed on screen during the timed runs should be displayed as 00:00:00 and not as 0:0:0 because it would be easier for students to understand.

Analysis of responses from question 2:

The responses show that there are still a lot of improvements to be made to my program especially to the Run form. However this does mean that I can make my program as good as possible thanks to the feedback; so that even more students are happy with it. Therefore I want to put most of the suggestions made by the students questioned as improvements to be made to my project. However I didn’t think that it was important to make changes to my project so that the user can’t only enter spaces because not many users will do this and the rest will enter their name properly. Also it allows those who don’t want to input their name into the system to save their time achieved. I will be making the following improvements based on the user feedback for this question:

* I believe that a glossary should be added because one of the students in my Further Maths class thought that it would be a really good idea; so that users can look up words that they don’t understand. I also believe that it will be easy to implement because all I would have to do is to add another form and name it “Glossary” and then add a button for it in the main menu or an option for it in the Kruskal’s algorithm combo box. After that I would just have to add labels to the form to add a new word to the glossary. The glossary would state all of the words used in the program and in network questions that are difficult to understand; words such as “Cycle” and “Minimum connector”. For the minimum connector part of this glossary I will explain what a minimum connector is and what it is used for in industry.
* I will also make the arc weightings on the network for runs randomized which will be the same as it is for timed runs. The timed runs and runs will also feature different networks which again could be easy to implement because I could use a random number generator to decide which form is loaded so that I could have five forms assigned to runs and timed runs and then each one of these corresponds to a number and if the random number generator produces the form’s corresponding number then it will show that form. I will also just have to copy the code I currently have for the Run and Timed\_Run forms and then paste it into these new forms so that then the only major change would be the networks on each of the forms.
* Instead of saying “However for the exam you will have to draw the minimum connector separately from the network given.” when someone completes a run of an algorithm the program will now display this “Now try again but instead draw out the minimum connector on paper using Kruskal’s algorithm and then check your answer using the Run feature”. This is so that students get more out of using the Run form.
* The program could be changed so that if two or more arcs have the same arc weight then the user would still get the choice correct no matter which one they chose in any order. However this would be quite complicated to implement.
* For runs instead of having message boxes appear after every stage of the algorithm the user would just have to press the “Enter” key to move on to the next stage and press the “Backspace” key to go back a stage. Therefore I could then display text to help the user understand what the algorithm is doing better. This will also mean that the message boxes won’t get in the way of the network and the user will be able to exit a run whenever they want to.
* I will also implement a stopwatch which displays the time in this way 00:00:00 instead of 0:0:0 because it is more conventional and it is easier for students to understand. This will be done so that when millisec = 0-9 or sec = 0-9 or min = 0-9 then they will always have a “0” in front of this number when displayed on screen.

These will be mentioned and in explained in more detail in my improvements section of this document.

Due to the problem concerning two or more arcs that are in the minimum connector that have the same arc weight my program does not fully apply Kruskal’s algorithm to the network because the fact that it doesn’t matter which is chosen in what order is part of the algorithm.

1. Did you come across any errors in my program, explain?

The most common answer to this question was that a cycle was produced in the timed runs, meaning that an incorrect minimum connector was produced. The two who said this were me where I came across a few during construction and the student only found that one cycle was produced during the 45 minutes that she was testing it for. The others said that there were no other errors apart from the improvements mentioned in Q2 of this questionnaire; which they said weren’t really errors. I also noticed that in the Update\_Kruskal and Times\_Achieved forms a blank record was output to the screen meaning that it displayed a time of 0:0:0 but no rank or forename. However because only I noticed this then it is clear that it is not that big a deal. I also noticed during testing that when the user enters a long forename the time achieved by them is not displayed in line with the other user’s times achieved.

Analysis of responses from question 3:

I was surprised when I first found a cycle in a small minority of my timed runs but most of the time my program works as it should. I have also checked the code for any issues and I can’t find anything therefore I think that it must be either a small issue with my coding or there is a bug/problem with Visual Basic. However in any case I have told users what to do if the program produces an incorrect minimum connector in the user manual. Therefore there is no need for an improvement to be made to my program to stop this. I will make the following improvements to my program based on the responses to this question:

* Even though it may be problematic I will try to stop the program from creating a blank record each time the Update\_Kruskal and the Times\_Achieved forms are opened; as I believe it makes my program look less professional.
* I will also reduce the number of characters that a person can enter for their forename to 25 characters so that users with long names won’t have their time achieved moved out of line with the others as much. However I have noticed that to prevent this problem fully then the layout of both the Update\_Kruskal form and the Times\_Achieved form must be changed meaning that the list boxes must be made larger and the form so that the times achieved can be moved further over in the list box.

I am pleased that the users I have questioned have spotted errors in my program that I didn’t see, so that I can now correct them at another date. It also shows that I have built a fairly robust system because most of the students I questioned couldn’t even find one error in my program.

1. Did you find my user manual easy to use with the program and understand?

Yes 4

No 0

Don’t Know 0

Analysis of responses from question 4:

This shows that most of the people who will use my program will find my user manual easy to use and understand with my program; meaning that users will use it more often because it will be a lot less confusing to them. This will then mean that users will not be put off by using my program and instead want to use it as they know what to do, thereby encouraging them to want to use it.

1. Why?

Most suggested that my user manual was easy to understand that that the images were useful and clear, it was informative as it explained what to do if a cycle appears, what each button and form does and I said that the installation instructions were easy to understand because I had a go at installing Visual Basic 2010 express myself to see whether they were accurate. However some of the people questioned seemed to be undecided on whether my program was concise, for example one said that the text was straight and to the point and therefore making it easy to follow but another said that it was clear but not particularly concise. However this was the only person out of those questioned that didn’t think it was concise.

Analysis of responses from question 5:

The answers to this question proved that my user manual was concise, informative, and easy to use and understand. It is also pleasing to see that they thought it was very useful in learning how to use my program and learning what everything does in my program. Also when I installed Visual Basic 2010 express I found that my installation instructions were very accurate and helpful in installing the software. Therefore due to the answers to both question 4 and 5 I have decided that my user manual is as good as it can be and therefore doesn’t need improving, however if my other improvements are implemented then the user manual will have to be changed to accommodate these changes.

1. Rank these resources by their usefulness? (1=most useful and 6=least useful. 1-3 count as low ranks and 4-6 count as high ranks.)

Resources Ranks given

The D1 textbook 2,2,4,5

The D1 revision guide 6,4,4,5

The D1 Moodle Resources 5,6,3,3

The MEI Integral site 1,2,6,6

My D1 teaching program 3,3,5,4

The teacher 2,1,1,1

Analysis of responses from question 6:

This shows that most of the students questioned thought that my D1 teaching program was as good as a lot of the other resources in terms of usefulness and was clearly seen as being more useful that the D1 Moodle resources and the D1 revision guide. Also if my program was ranked really highly by these students then they wouldn’t want to use it for revision but if it was ranked really lowly it would cause them to perhaps use it all the time but it is ranked somewhere in the middle of these two extremes. This suggests that students would want to use my program with the other resources sometimes and without the other resources sometimes too; so that when my program is used with another resource then the other resource would be made more useful and my D1 teaching program would be made more useful too. This is what my program is designed for; this being that it is able to be used with the other resources and without so that users can get the most out of it. However if I wanted to improve its rank among the other resources I would have to make it more teaching based like going through examples; which is what I have tried to do using runs but they need to be improved with more detail and explanations.

1. Why have you given each resource their rank?

The D1 textbook:

Those who gave this resource a low rank said that it was useful because it gave students lots of different practise questions.

Those who gave the resource a high rank said that the examples and questions were difficult to understand and difficult to follow. Also I said that it couldn’t be used in bite sized chunks and the other person said that there are no detailed solutions to the questions unless you log on to the MEI Integral site.

The D1 revision guide:

All of the students questioned gave the D1 revision guide a high rank because most of them didn’t use it and therefore didn’t know what it was like to use for study. However one student said that the revision guide is useful for other modules, however they didn’t really use the D1 revision guide. I said that the good point about the D1 revision guide is that it can be used in bite sized chunks and that it is basically the textbook but it is more concise.

The D1 Moodle resources:

Those who gave this resource a low rank said that the D1 Moodle resources were useful because they explained thoroughly how to apply the algorithm.

Those who gave this resource a high rank said that there were some good resources on Moodle but there were very few. The PowerPoint’s on Moodle which show how to perform algorithms, do linear programming and perform simulations were very useful as were the exam solutions done by my end user, which are easier to follow that the exam boards; however they do contain mistakes.

The MEI Integral site:

Those who gave this resource a low rank said that the MEI Integral site has many worked examples, questions, worked solutions to questions in the textbook, study plans which suggest which questions are better to do in the textbook and additional exercises for practise. All of these resources plus more are very useful.

Those who gave this resource a high rank said that they didn’t use the MEI Integral site very much for revision of D1 and therefore didn’t know how useful it was for their studies of D1.

My D1 teaching program:

Those who gave my resource a low rank said that my program was very easy to use; it is good for giving users a good understanding of Kruskal’s algorithm in a short space of time. I also thought that it works very well with the other resources for teaching Kruskal’s algorithm, thereby making it a very powerful revision resource. However both of us said that the resource only covers one section of D1 this being Kruskal’s algorithm whereas other resources cover all of the D1 syllabus, therefore suggesting that another resource is better.

Those who gave my resource a high rank said the same thing as those who gave low ranks this being that this program only covers a small part of the D1 syllabus and therefore can be less useful than the other resources at teaching D1. However they did say that it is a very useful revision tool for Kruskal’s algorithm and also one remarked that if the question asked specifically about ranking the resources in order of their usefulness at teaching Kruskal’s algorithm then they would have given it a lower rank perhaps a 2/3.

The teacher:

All of the students questioned gave this resource a low rank, the reasons for this were because it is very helpful to talk to someone who knows the subject and can show you where you have gone wrong in solutions to questions. Also I said that all of the students get taught the D1 syllabus from the teacher, meaning it is a resource that everyone will know whether it is useful or not. He can also go through worked examples on the board and encourage students to work independently; this is done by giving them questions to answer. They also tell students that the other resources exist so that they can use them for revision purposes so that they can gain as much understanding and knowledge of the D1 syllabus as possible before the exam. However one person said that this resource is limited to lesson time and therefore can be frustrating if students have a burning question.

Analysis of responses from question 7:

The answers to this question showed why students use some of the resources for studying and why they don’t use some of the resources for studying. This means that I can now suggest improvements to my program that it will encourage more students to use it for revision purposes. The common reason for why students didn’t like a resource was that they didn’t use it because they either didn’t realise that the resource existed when they were doing D1 or they didn’t own it; the D1 revision guide was the only resource that some users didn’t have access to or didn’t own. However with my program this will hopefully be prevented because my end user has uploaded the program and user manual to Moodle so that students will see it when they are enrolled in the course. My end user has also said that he will tell students that they can use my program as a way of learning Kruskal’s algorithm; this will alert them to the presence of my program on Moodle and if they know that a resource exists then they are more likely to use it. Also my program is free, meaning that those students with less money can use the resource; this thereby encourages more students to use it for studying because you have to pay for the MEI Integral site and the revision guide. Also another reason for why a student may not want to use a resource is because the resource is too complicated and not concise enough. This means that it is difficult for these resources to be used in bite sized chunks; which isn’t good for revision. These aspects put students off using these resources due to them being difficult to read, meaning that the information is not taken in as well. My program however can be used in bite sized chunks, it is simple, it is easy to use and understand and therefore the information is more easily taken in by students; my program has met these objectives which was proved by responses to other questions in this questionnaire and my end user feedback. The main reason for the students questioned not choosing my program to study D1 was that the program only covers one part of the network section of the D1 syllabus. Therefore for them to use my program over other resources it would have to cover more of the D1 syllabus.

The common reason for why students did like a resource was that they give you lots of different resources the priority of which are worked examples of questions, practise exam questions and fully worked solutions to questions. This means that my program will have to be improved so that it contains more of the workings that students have to do to complete a Kruskal’s algorithm question in the exam; this is so that they can see how to answer a Kruskal’s algorithm question correctly in the exam. These improvements will definitely have to be made in the Run form so that the worked example is clearer, detailed and helps the student through the question better. These improvements may also be made to the Timed\_Run form because the solution to the question shown on screen may not be as good as it can be as it doesn’t contain all of the workings needed to complete a question on paper in the exam. I was very pleased that the students questioned remarked on how my program is easy to use and understand and that it can be used very well with the other resources; these are two of the main objectives that I want my program to meet.

Due to the responses of this question I am going to be making the following improvements:

* At the end of a run the program displays the minimum connector in word form as well as on the network and its weight; which will be displayed at the bottom of the screen once finished. This will look something like this:

Minimum connector = CD, AC, GF, CB, DE, EG Weight = 23

* However at the end of a timed run the program asks the user to input the minimum connector and its weight. This is so that the student is tested on their knowledge of Kruskal’s algorithm more like they are in the exam. Text boxes and labels will be used so that the user can enter what the minimum connector is and its weight. This will look something like this:

Minimum connector = , , , , , Weight =

The user will have to type each of the six arcs in the minimum connector into each of the six text boxes; in the form that they will have to write it in the exam.

These improvements to the Run and Timed\_Run forms could be implemented very easily, for example in the runs the minimum connector and its weight would be the same every time meaning that I could just output these two answers to screen. However if the arc weights were randomized due to the improvement to the Run form mentioned earlier then the way to implement this would be the same as in the Timed\_Run form; except that the user wouldn’t need to enter anything. This would be done by using the textboxes so that it depends on what position1, postion2, position3, position4, position5 and position6 equal. In the Timed\_Run form this code will be written so that it doesn’t matter which arc is put into each textbox but instead it matters whether the correct arcs that are in the minimum connector are put into the textboxes in any order. The program could then mark whether each one is correct or incorrect by displaying picture boxes with a tick or cross in them. If the weight entered and minimum connector entered by the user are correct then the stopwatch will stop and they will be taken to the Forename\_Entry form after clicking on the “OK” button on the message box; which says that they have completed the timed run. However users will only be able to enter the minimum connector and its weight when the user has produced the minimum connector on screen. In the Run form the minimum connector and its weight will just be output to the screen in labels.

* For runs instead of having message boxes appear after every stage of the algorithm the user would just have to press the “Enter” key to move on to the next stage and press the “Backspace” key to go back a stage. Therefore I could then display text to help the user understand what the algorithm is doing better. This would probably be quite difficult to code because I would have to write a lot of code so that it recognised the “Enter” and “Backspace” keys being pressed and then once the “Enter” key had been pressed the code would have to cause the algorithm to move onto the next stage and if the “Backspace” key had been pressed the code would have to cause the algorithm to go back a stage. Therefore I would have to get a lot of help with this and it would take a long time to implement.
* Also I want to improve my program so that it covers more of the D1 syllabus, therefore I am going to try and get my program to perform Prim’s algorithm network form, Prim’s algorithm tabular form and Dijkstra’s algorithm. This will require four forms for each algorithm like with Kruskal’s algorithm at the moment. Prim’s algorithm network form will be the easiest to code because it is the same as Kruskal’s algorithm except that the smallest arc weight to go into the minimum connector would be chosen from the arcs that branch from the starting vertex and then the next arc that is in the minimum connector is the shortest arc from the next vertex and so on until all six arcs are in the minimum connector. The other two algorithms will be more difficult to code but I will have to draw a table for Prim’s algorithm tabular form whereas with Dijkstra’s I could just use the same network/networks as for Kruskal’s algorithm and Prim’s algorithm network form.

1. Which of the following objectives does my program meet?

Its interactive 4

Its informative 3

It can be used in bite sized chunks 3

It enables students to study on their own 4

It’s enjoyable and engaging 4

It teaches Kruskal’s algorithm 1

Analysis of responses from question 8:

This shows that most of the main objectives that I set the program have been satisfied according to students because they believe that it is interactive, it’s informative, it can be used in bite sized chunks, it enables students to study on their own and it’s enjoyable and engaging. This means that all of these objectives have been met and to a very good standard because most of the people questioned believed that these objectives had been met by my program. Also I said that it meets one of my objectives this being that it teaches Kruskal’s algorithm which my program does and to a very good standard.

1. Do you think that this program will help students with their studies of D1?

Yes 3

No 0

Don’t Know 1

Reason for your answer:

The main reason for those who answered “Yes” to this question was that the program allows people to practise performing Kruskal’s algorithm; and therefore teaches it to students. Also a lot can be got out of it in a short space of time and it initiates competition between students which may help people enjoy it more. People also thought that it was easy and simple to use also me and the others thought that it would be very good at being used with the other resources. We thought that this could make those resources better at teaching students Kruskal’s algorithm with my program. For example if Kruskal’s algorithm is initially difficult for a student to understand then my program will allow them to understand it better where they can then build on this understanding by using the other resources. However I thought that the program may not have a large impact on students studying D1 especially if there is no Kruskal’s algorithm question in the exam; although I do believe it will be a massive help to learning Kruskal’s algorithm.

The student who answered “Don’t know” to this question said this because he thought that some students may use resources that could explain more about other algorithms thereby saving time because they wouldn’t have to keep changing resources. However he did think that the program is helpful for learning this area of D1.

Analysis of responses from question 9:

It was pleasing to see that the majority thought that my program will help students with their studies of D1 this was mainly because the program can be used with other resources, initiates competition, allows users to practise using Kruskal’s algorithm, simple, easy to use and understand and it allows students to get a lot out of using it in a short space of time. This will thereby encourage students to use it when learning Kruskal’s algorithm. The only issue that a student questioned had with saying that my program will help students with their studies of D1 was that it only covers a small part of the D1 syllabus and would therefore prefer to use the other resources instead. I hope that this issue will be addressed with the improvements that I proposed earlier because it has been mentioned by students throughout this questionnaire; therefore it is very important for me to improve this.

End User feedback

My end user feedback was collected from my end user this being my Maths and Further Maths teacher, Andy Bradley. He has filled in a questionnaire so that I can see if what I have produced is what he wanted me to do from the start. Below is a summary of what my end user said for each question along with the analysis of what he said for each question:

1. What did you like about my program and why?

He said that my program clearly shows Kruskal’s algorithm in operation, it challenges students to use the algorithm correctly and quickly, it enables students to see the correct use of the algorithm, it is simple to use and easy to understand.

Analysis of response from question 1:

This was pleasing because he highlighted exactly what I wanted the program to be and do; he seemed to be very pleased with what I have produced for him as per the objectives we set the program to meet at the begin. Also what he has said means that my program performs Kruskal’s algorithm correctly; which coming from someone who has a lot of experience at Maths and Kruskal’s algorithm is very reassuring and tells me that users will definitely be learning how to perform the algorithm correctly in preparation for the exam. Because my program has clearly put these thoughts across to him then he can tell other students about my program where I believe they will be impressed by what he says about it and will therefore want to use it for revision.

2. What do you think needs to be improved about my program and why?

He suggested for improvements that instead of saying “However for the exam you will have to draw the minimum connector separately from the network given.” On the Run form he thought it would it would be better to say “For the exam you will have to draw out the minimum connector separately from the network.” On the Run form he thought that the arcs selected should be much thicker as they are selected so that users can see them better; as he said that he couldn’t see the coloured arcs very well. He also said that it would be better if the arc weightings were randomized in runs like they are in the timed runs and it would be even better if the Timed\_Run and Run forms had 5 completely different networks to use for timed runs and runs of Kruskal’s algorithm.

Analysis of response from question 2:

Most of the improvements that my end user has mentioned have been ones that the students questioned have also mentioned. However I will now make the following improvement to my program:

* On the Run form I will make the arcs on the network as large as the ones on the Timed\_Run form, meaning that the two networks will look the same in terms of layout and size.

The other improvements that were suggested by my end user have already been covered by the improvements that I have suggested after analysing the responses from the Further Maths students.

3. Did you come across any errors in my program, explain?

He said that he didn’t come across any errors in my program.

Analysis of response from question 3:

This is very pleasing because it supports the fact that I have produced a very robust solution to the problem that my end user wanted solving.

4. Did you find my user manual easy to use with the program and understand?

Yes / No / Don’t Know

Analysis of response from question 4:

This proves that my user manual is very easy to use and understand with my program because all of the people that have been questioned about it have answered “Yes”.

5. Why?

He said that my user manual was simple and to the point, meaning that he thought it was concise.

Analysis of response from question 5:

This is pleasing because this is exactly what I wanted my user manual to do and be like and therefore it will be a very easy read for students and students will be able to find out what they need if they are stuck; easily.

6. Is it what you wanted me to produce? Does it meet the objectives that we agreed upon during the analysis stage?

Yes / No / Don’t Know

Reason for answer:

He said that the reason for answering “Yes” was because it meets almost all agreed objectives; the exception being that it only does one algorithm (Kruskal’s). However he did say at the time that trying to get the program to perform all four algorithms was very ambitious and he believes that it is better that I have done a good job with one than a poor job with all four.

“The proposed system must run and teach students about at least one of the four algorithms.” This was made a main objective for my program as we knew at the start that the likely outcome would perform only one algorithm due to time constraints.

Analysis of response from question 6:

This is very pleasing indeed to hear my end user say this because it means that I have made the program achieve what we wanted it to achieve and it means that my hard work has paid off. I was also pleased that I did a good job at making the program perform Kruskal’s algorithm, however I would have liked to have implemented more of the four algorithms that we agreed upon at the start. But according to this from my end user it doesn’t matter too much and he is still very pleased with the final results.

7. How well does this program meet the objectives that we agreed upon during the analysis stage?

He said that the program meets the main objectives because he believes that it is interactive, it can be used in bite sized chunks, it enables students to study on their own, it is informative and it was finished by Easter. How well I achieved the first four of these objectives are shown below as said by my end user:

It’s interactive: There is an opportunity for students to compete with other students at timed runs thereby making them go faster.

It’s informative: The information contained in the program is easy to understand and remember.

It can be used in bite-sized chunks: It is simple to use the different modes such as the demonstration (run) and time challenge (timed run). It does not take long to use and to get some understanding of the algorithm.

It enables students to study on their own: It is clear and it is easily followed by all students.

Analysis of response from question 7:

I am again pleased about how well my end user thinks that I have done in meeting these objectives because I worked really hard at trying to make my program meet them. Also I liked that it was finished well before Easter; therefore my program was up and running early enough. Also this response from my end user again proves that I have produced a very good program as I have met a lot of the objectives we set it to meet to a very high standard which will encourage not only students but also teachers to use my program for revision purposes or maybe even for teaching lessons. Teachers could teach lessons using my program by using the Run form as the basis for a worked example on the active whiteboard.

8. Which of the following objectives does my program meet?

Its interactive 1

Its informative 1

It can be used in bite sized chunks 1

It enables students to study on their own 1

It’s enjoyable and engaging 1 Analysis of response from question 8:

This is very pleasing because many of the people questioned indicated that my program met most or all of these objectives which are some of the main objectives me and my end user set this program at the start to meet. This will encourage students to use it because students who use it will mostly only have good things to say about it, meaning that they will then recommend it to others. I am also very pleased that my end user also thinks that it is enjoyable and engaging because I found it difficult to put this across to the user during construction. I am also very glad that my end user thinks that all of these objectives have been achieved because they are an integral part of whether a resource is liked or disliked.

9. Do you think that this program will help students with their studies of D1?

Yes / No / Don’t Know

Reason for your answer:

He said that it gives the topic more of a gaming feel and that will help students to engage and encourage students to use it more.

Analysis of responses from question 9:

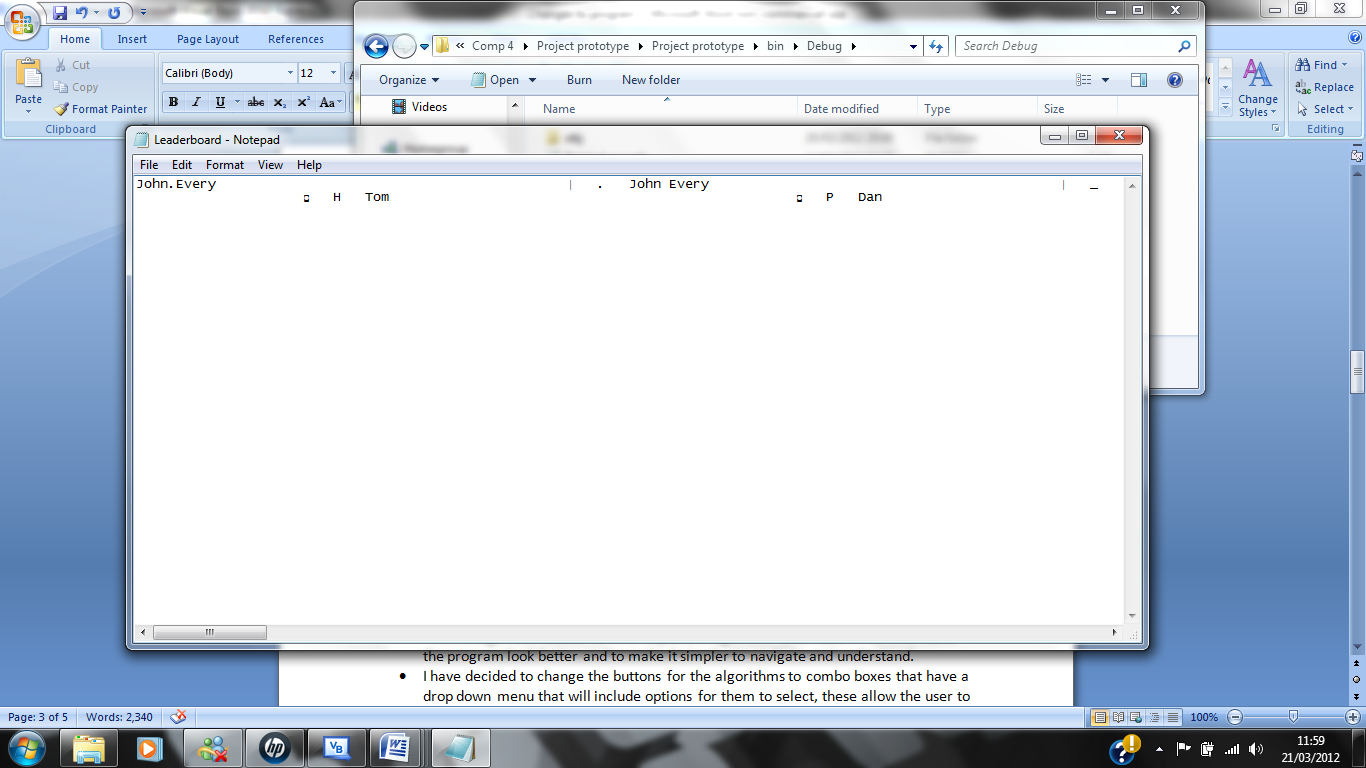
I am pleased that my end user thinks this of my program because this is one of the aspects I wanted it to have because I believed that making it more like a game will attract students to using it and thereby making it easier for the student to take the information in.

Changes section

The changes made before the design are shown in the design section pages 49-50.

Changes made during construction:

* I have decided that the arcs and the weights in table boxes will change colour to red instead of blue, so that those with poor eyesight can still see when an arc or weight in a table box has been included in the minimum connector. This is because someone with poor eyesight who looked at my program said that it was difficult to see the change in colour of the arcs and had difficulty seeing the ring around the weight in the table box. I have decided to change this as red-green colour deficiency is less common than poor eyesight, therefore I will be helping more people. However the vertices will still change to blue for Dijkstra’s algorithm because they are big and can be seen easily.
* I have decided to make it so that during runs and timed runs of Prim’s algorithm network form and Kruskal’s algorithm the vertices will change colour to blue so that users know that a vertex has been connected.
* For runs I have decided to change the way in which users move from each stage by instead of them pressing the “Enter” or “Backspace” key to go onto the next or previous stage, the program will display message boxes after every stage that says “Go to next stage” where they have to press the “OK” button or the “Enter” key to move onto the next stage, therefore they can’t go back to the previous stage. The run will now be started by the user pressing a button that says “Begin run” which will then be hidden for the remainder of the run. I have decided to do this because it is easier and simpler than the previous method because a lot more coding would be needed to do it. Also I would rather spend more time on coding the algorithms for the runs and timed runs instead of concentrating on something that isn’t necessary.
* I have decided that the program will display the following text on screen at the beginning of a run which will then be removed once the run has been completed: “Find the minimum connector for this network using (insert name of algorithm here) algorithm. Draw the minimum connector." And below this "If you have the D1 revision guide then turn to page 23 where the example allows you to follow the use of the algorithm in this run." The following text will be shown once a user has completed a run but hidden at the start and during it: "You have completed a run of (insert name of algorithm here) algorithm and you have found the minimum connector of the network." And below this: "However for the exam you will have to draw the minimum connector separately from the network given." This is because I believe this will help the users more than what I had previously decided to display.
* I have decided to change the cursor property on all the line shapes for timed runs and the help button to “Hand” so that then when the user’s cursor is over the line shape or the help button the cursor will change to the “Hand” icon so users know that they can and have to click on the line shapes and click on the help button.
* Also I have decided to make the arcs larger for timed runs than runs because I found that it was difficult to click on the arcs as they were so small, therefore this will now be made easier. However for runs the arc size can remain the same as it is unimportant.
* I have decided to change the maximum time allowed to do a timed run of an algorithm to 9:59:99 this is because the user needs to be able to complete a timed run in less than 10 minutes. This is because if they take an hour in the exam then they will more likely not have enough time to do the rest of the paper.
* I have decided to not implement Dijkstra’s algorithm, Prim’s algorithm network form and Prim’s algorithm tabular form because I am short on time and I would rather focus all my efforts on implementing and making the Kruskal’s algorithm pages as good as possible instead of doing a lesser job on all four. I can do this because I have spoken to my end user and he is fine with the program only teaching one algorithm; as we had agreed in the objectives that the program must teach students at least one of the four algorithms.
* I have changed the way in which the help button works because instead of a dropdown box being displayed a list box will be displayed instead with the algorithm inside. They can also use the scroll vertical and horizontal bars to see all of the writing; as I don’t want it to take up too much space. Also instead of a button I have coloured an oval shape blue and then placed a label inside which has a white question mark in it. This is so that it is more like the help button used in Microsoft Office meaning it is more familiar to users, meaning they are more likely to know what it does without looking at the user manual. Also when they press the help button again the list box will be hidden by the program so that users can take it off the screen when they don’t need it any more.
* I have decided to allow users to use a space when they enter their forename as I don’t think it is that important to the design of the program and it will make it easier for users with the same name to distinguish themselves on the leader board. Also if the user wants to enter their surname or middle name then they can but if users don’t want to divulge this information then they can just enter their forename. I have been allowed to do this by my end user because I had a meeting with him and he said that he agreed that a space should be allowed when users submit their forename.
* I have made my program save users forenames and times achieved into the leader board file until it reaches a maximum of 100 data items where the program won’t save anyone else’s data until some of the records have been deleted by the end user; this being Andy Bradley or another Maths teacher. The user will know when the maximum amount of records has been reached because once they have completed a timed run and they have been taken to the Forename\_entry form and they try and enter a forename and press the “OK” button a message box will be displayed and it will say this “Data not saved because the maximum of 100 records has been reached; contact Andy Bradley", once they press “OK” on the message box the program will take the user back to the Main Menu page.
* Also because I am now using binary random files instead of text files data is stored differently in them, therefore this is how they are now displayed in the file:



* I have also changed the way in which the program deletes records from the leader board file by instead creating a new file called “Kept records.dat” and saving all of the records I want into the new file and then renaming this file to “Leaderboard.dat”, so that only the files wanted are kept. Before this the original “Leaderboard.dat” file is deleted.
* Also instead of saving the ranks of each person’s time into the “Leaderboard.dat” file I will instead apply the bubble sort to the records each time they are output to the screen this being in the Times\_Achieved form and the Update\_Kruskal form. This is so that the files won’t take up as much room in the file and it will make it easier to code.
* Even though I have decided that that my program will code for less algorithms I still want to keep the form’s titles and names the same so that my program can be adapted easier later on to include more maths and computing modules.
* Also along with a “Save” button there will also be a “Delete” button on the Update\_Kruskal form. This is so that the teacher can select the record in the list box that they want to delete and then press the delete button on the form, where a message box will appear which will ask the user whether they want to delete the record if they press the “No” then it won’t delete the record if they press the “yes” button then it will. However if the “Save” button is not clicked then these changes won’t remain permanent and therefore won’t be saved into the file. When the “Save” button is clicked the user will again be asked if they want to save the changes where they will have to press the “Yes” or “No” button, if they press the “Yes” button then the changes will take effect; they will see this if they press the “Quit to Main Menu” button and then go back into the form. If they press no then the records that were deleted will return when they load the form again. Also there is no limit to how many deleted records can be deleted permanently, however the user will have to delete each record one at a time.
* If the stopwatch for timed runs is left to count up until the maximum time limit of 9:59:99 then instead of the user having to enter their forename they will instead be told that they have failed the timed run and directed back to the main menu because they must learn to complete it in under 10 minutes.
* Instead of the algorithm moving on to the next stage when you click on the correct arc the algorithm has already been run by the program and the user then has to choose the arcs that are in the minimum connector in the correct order as you would do normally. Therefore if each arc chosen at each stage by the user is the same as the arcs chosen at each stage of the algorithm that was performed when the “Start” button was pressed; then the user’s chosen arcs were correct.
* 0:0:0 instead of 00:00:00 because it is easier to code and it is not as important as the coding of the algorithms and inputting the records into the correct file.
* The arc weights will now be stored in a one-dimensional array because I have no need for the vertex letters any more as I am not performing Dijkstra’s algorithm. However there is another one-dimensional array which contains true or false statements at each position. The true or false statements indicate whether an arc which is at the same position in the done() array as its corresponding arc weight in the weightings() array cannot be chosen to be in the minimum connector.
* I have decided to not display the question at the top of the Timed\_Run form because I didn’t think that it was important for timed runs because they have probably just used the run form; therefore they don’t need the question being shown. Also if they are doing D1 then they will already know what the questions are like anyway.

Evaluation

Below are the objectives that me and my end user set the project along with how well they have been met and if they have been met:

|  |  |  |  |
| --- | --- | --- | --- |
| Objective number | Objective | Has it been met by my program?(Yes/No/Don’t know) | How well has it been met? (Use evidence)  If it hasn’t been met, why?  If it hasn’t then why not? |
| Overall objectives below: |  |  |  |
| 1 | The proposed system must be interactive. | Yes | I have met this objective to a very high standard because in both the end user and user feedback questionnaires all of the five people ticked the “It’s interactive” box for their answer to question 8. Therefore they all believe that my program has met this objective; meaning that because it was unanimous then the objective must have been met to a very high standard. |
| 2 | The proposed system must enable people to self study without needing to go to lessons. | Yes | I have met this objective to a very high standard because in the end user and user feedback questionnaires all of the five people ticked the “It enables students to study on their own.” box for their answer to question 8. Therefore they all believe that my program has met this objective; meaning that because it was unanimous then the objective must have been met to a very high standard. |
| 3 | It must be able to be used in bite sized chunks and be enjoyable thereby allowing people with less time to absorb large amounts of information to prepare them for an exam. | Yes | I have met this objective to a very high standard because in the end user and user feedback questionnaires four out of the five people ticked the “It can be used in bite sized chunks.” box for their answer to question 8; the person who didn’t was one of the students. Also all five people ticked the “It’s enjoyable and engaging” box for their answer to question 8; meaning that the students found my program to be very enjoyable. Because most of the people questioned believe that my program has met this objective then the objective must have been met to a very high standard. |
| 4 | It must be able to be used alongside the current system. | Yes | I believe that I have met this objective because some of the students questioned suggested and said that they thought it could be used well with the other resources thereby making my program a powerful resource. These comments were made and suggested in question 6, 7 and 9 of the user feedback questionnaire. |
| 5 | In a question for a run or timed run of Dijkstra’s algorithm the starting vertex must be indicated. | No | I have not met this objective because I had limited time and therefore I didn’t have time to code Dijkstra’s algorithm into my program. |
| 6 | The proposed system must be informative. | Yes | I have met this objective to a very high standard because in the end user and user feedback questionnaires four out of the five people questioned ticked the “It’s informative” box for their answer to question 8. Therefore the majority believe that my program has met this objective; meaning that the objective must have been met to a very high standard. |
| 7 | The proposed system must be able to be accessed through the D1 Moodle resources. | Yes | The system can be accessed through the D1 Moodle resources along with my user manual; where both have now been uploaded and can be opened in the D1 Moodle resources by Maths and Further Maths students. However due to limitations in my equipment I am unable to deploy my program, therefore the code can be seen and altered by users. |
| 8 | The proposed system must be able to be used by students without any specialist software. | No | I have not met this objective because due to limitations in my equipment I am unable to deploy my program, therefore students who want to open and use my program will need Visual Basic 2010 express or similar installed on the computer; so that they can open and use my program. |
| 9 | The proposed system must be able to be saved onto and used from memory sticks and CDs. | Yes | I have met this objective because it can be saved on to memory sticks and CD’s and used from them, however I decided during my project that this is not a necessity and therefore didn’t matter too much if I didn’t meet it. This is because the changes to the leader board on the memory stick and CD files will not be able to be updated to the master copy on Moodle but it can be used for study purposes just as well. |
| 10 | The proposed system must be completed by Easter at the latest. | Yes | I completed the system two weeks before the Easter weekend. Therefore I believe that I have met this objective to a very good standard. |
| 11 | It must encourage students to use it. | Yes | I believe that I have met this objective to a very high standard because as proved by my user and end user feedback all of the people questioned and my end user said that my program met the following objectives in Q8: Its interactive, it enables students to study on their own and it’s enjoyable and engaging. Also most of the five people questioned thought my program met these objectives: It’s informative and it can be used in bite sized chunks. Because these objectives are thought of as being met by the majority then they must have been met to a high standard. These aspects will also come across to other users and therefore this will encourage students to use it because revision is easier for people if it is done little and often. Also if it is enjoyable and engaging then they will be keener to use my program and a growing number of students like interactive learning. Finally the fact that it is informative is very important because the users will know that this correctly performs Kruskal’s algorithm and is teaching them the right stuff. This is proved by my end user in his response to question 1 of the end user questionnaire; where he says that my program correctly performs Kruskal’s algorithm. Also a lot of the students that I questioned said that my program was easy to use and understand, simple and professional. This will encourage people to use it because these aspects make resources better; this is proved by the responses that I obtained from question 6 and 7 of the user feedback questionnaire where the students who gave my program low ranks gave them for the reasons just mentioned. |
| 12 | It must be able to be used by the teacher; by themselves and to teach lessons. | Yes | I believe I have met this objective to a fairly high standard; however I only think that teachers would use the run feature for teaching lessons because it would help them prepare a worked example or use it to perform one to the students. I don’t think that the rest of my program could be used to teach lessons although the timed runs could be used as a question for the students to work through during class. |
| 13 | It must help students to complete past exam papers on the topic. | Yes | I have met this objective because my program performs Kruskal’s algorithm correctly; this was said by my end user in Q1 of his feedback form. Therefore because it performs Kruskal’s algorithm correctly the student can then follow the algorithm working and work out what to do to solve an exam question. Also my program provides information on what users will actually have to do in the exam. |
| 14 | The proposed system must find the minimum connector of a network for runs and timed runs of Kruskal’s algorithm, Prim’s algorithm network form and Prim’s algorithm tabular form. | Yes | I have met this objective, however my program only finds the minimum connector of the network using Kruskal’s algorithm. This is because I was limited by time constraints and therefore I didn’t have time to make my program perform the other algorithms. |
| 15 | The proposed system must find the shortest route between two vertices along with its weight for runs and timed runs of Dijkstra’s algorithm. | No | I have not met this objective because I did not have time to code my program so that it would perform Dijkstra’s algorithm. |
| 16 | The proposed system should be able to be expanded and updated to include other sections of the D1 module. | Yes | I believe that I have met this objective because I have kept all the Kruskal’s algorithm parts on their own as they would be in the program with other algorithms; therefore all someone would have to do to add other sections would be to add their bits as they wouldn’t need to alter mine. |
| 17 | The proposed system should be able to be replicated for other Mathematics and Computing modules. | Don’t know | I am not sure if I have completed this objective because it may be fairly simple to do a similar program for parts of D2 that use different algorithms to solve different problems; this could still be interactive. However it may be difficult for areas of Mechanics such as circular motion or parts of C3 such as differentiation and integration; which may be difficult to be made interactive and they may be difficult to code. |
| 18 | It should also be able to be used without the D1 textbook, the D1 revision guide, the MEI Integral site, The D1 Moodle resources and the teacher. | Yes | I know that I have met this objective to a high standard because the students I questioned for the questionnaire did not use any of these resources when they used the program to fill in my questionnaire. |
| 19 | It should teach people about all four of the algorithms. | No | I have not met this objective because I had limited time and therefore I didn’t have time to code Prim’s algorithm network form, Prim’s algorithm tabular form and Dijkstra’s algorithm into my program. |
| 20 | It should be able to be used alongside PowerPoint’s on the subject. | Yes | I have met this objective because in the analysis of my user feedback I found that some of the students questioned mentioned that they thought my program was great at being used with the other resources. |
| Specific objectives below: |  |  |  |
| 21 | The system must be able to process and display all of the forenames, times and ranks of those students who have completed timed runs; when requested by the user. | Yes | I have met this objective because the program outputs the leader board of times achieved in a list box on both the Times\_Achieved form and the Update\_Kruskal form. On the Times\_Achieved form it gave users the opportunity to see the leader board and where they rank with other users. On the Update\_Kruskal form the teacher can delete records from this leader board. In the user feedback section of this document some students questioned remarked on how professional my program was and how well laid out it was; therefore these forms were well laid out. |
| 22 | The proposed system must run and teach students about at least one of the four algorithms. | Yes | I have met this objective because my program runs Kruskal’s algorithm on both the Run and Timed\_Run forms. It also teaches students Kruskal’s algorithm correctly; which was proved by my end user’s comments to question 1. Where he said that he liked it that my program “enables students to see the correct use of the algorithm”. |
| 23 | The program must display the network and the question being answered when either a run or a timed run of the algorithm is chosen. | Yes | I have met this objective for Kruskal’s algorithm but only for runs because I didn’t think that it was important for timed runs because they have probably just used the run page therefore they don’t need the question being shown. Also if they are doing D1 then they will already know what the questions are like anyway. |
| 24 | The proposed system must display a set network for runs and timed runs of Kruskal’s algorithm, Prim’s algorithm network form and Dijkstra’s algorithm. | Yes | I have met this objective to a decent standard because I have only made my program perform Kruskal’s algorithm and not the other three due to time constraints. For runs and timed runs of Kruskal’s algorithm the set network is displayed each time the Run and Timed\_Run forms are loaded. |
| 25 | The proposed system must display a table for runs and timed runs of Prim’s algorithm tabular form. | No | I have not met this objective because I have not made my program perform Prim’s algorithm tabular form due to time constraints. |
| 26 | The proposed system must be able to store at least 100 students forenames along with their times and ranks for timed runs of the algorithms. | Yes | I have met this objective to a high standard because my program now only saves up to and including 100 records. Therefore the program stores at least 100 records. |
| 27 | The timer must be displayed on the screen when the user does a timed run of an algorithm. | Yes | I have met this objective because the stopwatch which displays the time elapsed since the user pressed the “Start” button is displayed on screen during all timed runs and the time achieved by the user is displayed on the Forename\_Entry form. |
| 28 | The program must produce the set menu that it will be coded for when it is started, when the quit button is pressed during runs and timed runs of algorithms and when displaying the times achieved by students. | Yes | I have met this objective because my user interface works to a very high standard as proved by my testing section (tests 1.1, 1.2, 1.3). |
| 29 | All the times achieved from timed runs of algorithms must be stored in order starting with the quickest time and inputting the rest downwards. | Yes | I have met this objective to a very high standard because the leader board of times achieved is displayed in a list box on both the Times\_Achieved form and the Update\_Kruskal form from quickest to longest time. Also all of the times achieved are displayed in line with one another except when the user enters a long forename. |
| 30 | The proposed system must store all of the times for timed runs of each algorithm in different files therefore there is only the times achieved for Kruskal’s in one file, times for Prim’s algorithm network form in another, times for Prim’s algorithm tabular form in another and times for Dijkstra’s algorithm in another. | Yes | I have only partly met this objective because all the times achieved at timed runs of Kruskal’s algorithm are stored in one file but because my program doesn’t perform the other algorithms then there is only one file that times are permanently saved into. |
| 31 | The proposed system must be completed by Easter for my end user. | Yes | I completed the system two weeks before the Easter weekend. Therefore I believe that I have met this objective to a very good standard. |
| 32 | It must output an error message if the forename entered contains characters that can’t be stored as string, contains spaces or contains 0 characters. | Yes | I have met this objective to a good standard because an error message is displayed if the forename entered by the user contains no characters or contains over 50 characters. However I decided to allow users to enter spaces because if users wanted to enter their surnames or middle names then they can. However this caused an error which was discovered in Q2 of my user feedback by a student. This error allowed a user to enter only a space for their forename; which was then seen on the Times\_Achieved form and the Update\_Kruskal form as being a record but it looked as though the user had entered no forename. Also forenames can have numbers in them; this is so that users can enter internet names if they want to. |
| 33 | It must display the working involved in performing the algorithm for runs and timed runs of each algorithm. However timed runs will do this immediately and runs will be slower so that the student can see what is going on. | No | Unfortunately I couldn’t meet this objective due to time constraints because what I wanted to do on the Run form would have been too difficult and it would have taken too long. Therefore I had to just allow the run of the algorithm to be performed with no working being displayed apart from the colouring of the arcs that are in the minimum connector. This is the same as in timed runs; however I decided during construction that it was not important for the timed runs to show the workings. |
| 34 | It must save the times, forenames and ranks of students into separate files for each timed run of an algorithm. | Yes | I have met this objective; however this is only the case for the times achieved at timed runs of Kruskal’s algorithm because I didn’t have enough time to code my program so that it would perform the other algorithms. |
| 35 | When the user chooses quit on the main menu it must exit the program. | Yes | I have met this objective to a very high standard; this is proved by my testing section (test 1.1). |
| 36 | The data stored in the proposed system should be able to be easily updated and maintained. | Yes | I have met this objective to a high standard because if changes are going to be made to my program it is fairly simple to make them, for example if someone wanted to add Prim’s algorithm network form to my program then they would only have to do what I have done for Kruskal’s but for this algorithm instead. This could be done very easily as they wouldn’t have to change the code that I have written very much. |
| 37 | The proposed system should be able to run Kruskal’s algorithm, Prim’s algorithm network form, Prim’s algorithm tabular form and Dijkstra’s algorithm in both runs and timed runs. | Yes | I have met this objective but only for Kruskal’s algorithm due to time constraints and therefore I did not have enough time to code my program so that it would perform the other algorithms in both runs and timed runs. |
| 38 | The proposed system should turn the arcs or vertices red if they have been chosen by the user and are correct in runs and timed runs of algorithms. | Yes | I have met this objective to a high standard because all of the arcs that are in the minimum connector turn red on the Run form and turn red when the user clicks on the correct arc on the Timed\_Run form. Also the vertices that these arcs are connected to turn blue so that users can see that the vertices have been connected. However in the end user feedback for Q2 my end user said that he had difficulty seeing the red arcs on the Run form; therefore I have suggested an improvement to my program to address this which is explained in the analysis part of Q2 of the end user feedback. |

Testing Evaluation

My testing section proved that my program works as it should in most cases apart from the Times\_Achieved and Update\_Kruskal forms creating a blank record each time that they are opened and that if a long forename is entered then the time achieved by the user will not be in line with the others.

During my testing I tried to use many different types of testing these being top-down testing, bottom-up testing, white box testing, black box testing, system testing, alpha testing and beta testing. This was to make sure that I managed to find as many errors as possible and this worked because during my beta testing one of the users found a cycle in one of the minimum connectors produced by a timed run. I couldn’t find anything wrong in the coding to explain this; however I believe that it could either be a small fault with my code or there is a fault with Visual Basic and a bug has been produced. Therefore in my user manual I have told users that if they encounter anything odd with my program then I have advised them to exit the program and then reload it, which I have found fixes the problem.

Even though I used a lot of different testing methods I have still not been able to test the working of my program when it is loaded on Moodle this is because I haven’t been able to deploy the program and put it on Moodle as I wanted. However I could put the file that contains the program on Moodle but users would be able to see and change the code but I would be able to see any problems that have been encountered by students and then try and fix them. If I can then I will need to adapt my testing section, however it will be more accurate and now relate more to what users see.

Improvements to the program that I have wanted to make during construction but could not do due to limitations in time and the improvements to my program that I would like to implement that my end user and users questioned have suggested.

* The program could send the times of the students to the teacher via e-mail once they have done a timed run of the algorithm. This is so that the teacher knows how students are getting on, however to make it more bearable for the teacher the program should send an email of the leader board after every three records have been saved. This means that only a small number of emails will be sent at once and the user’s time achieved will still be acknowledged by the teacher if there is a problem with saving.
* Also in the timed run of Dijkstra’s algorithm the program asks the user to perform all the working in the boxes like the working values before it moves onto the next stage. (This can only be done once Dijkstra’s algorithm has been implemented into my program.)
* For Prim’s algorithm tabular form the program will draw the network underneath the table for each stage, for a runs and timed runs. Also the student could enter the numbers above each column and cross out the rows that should be crossed out. Or they could click on the vertex’s column that they think should be marked and have its row crossed out. (This can only be done once Prim’s algorithm tabular form has been implemented into my program.)
* A search option could be included in the Times\_Achieved form and the Update\_Kruskal form, so that users can find users and their times achieved quickly and easily. This would be done so that if the teacher enters a name into a text box and presses the “Search” button then the program will highlight the first record achieved by this user in the leader board then if it is pressed again then it finds the next one and so on.
* Instead of having all of the times achieved sections in separate forms they could all be put into one form where the user can click different tabs at the top of the form to display different times achieved. The same could be applied to the ones in the teacher’s section. (This can only be done once other algorithms have been implemented into my program.)
* Make it so that the teacher can delete multiple records when only pressing the “Delete” button once this will aid speed and convenience. This will be done so that the teacher can select multiple records at a time and then press the “Delete” button.
* When the user presses either the “Delete” button or the “Save” button when nothing has been selected to be deleted or nothing has been deleted to be saved on the Update\_Kruskal form; then a message box should appear which says this to the user. This is because at the moment the program displays the message boxes as normal but the user may think that it is going to delete a record or save changes to the leader board which they haven’t made/done. This is just to prevent confusion for users using my program.
* For runs and timed runs of Kruskal’s algorithm or any algorithm implemented the question must be shown at the top of the form before the algorithm commences. This is so that the user knows what they are doing and what’s going to happen once they begin either a run or timed run of an algorithm.
* I believe that a glossary should be added because one of the students in my Further Maths class thought that it would be a really good idea; so that users can look up words that they don’t understand. I also believe that it will be easy to implement because all I would have to do is to add another form and name it “Glossary” and then add a button for it in the main menu or an option for it in the Kruskal’s algorithm combo box. After that I would just have to add labels to the form to add a new word to the glossary. The glossary would state all of the words used in the program and in network questions that are difficult to understand; words such as “Cycle” and “Minimum connector”. For the minimum connector part of this glossary I will explain what a minimum connector is and what it is used for in industry.
* I will also make the arc weightings on the network for runs randomized which will be the same as it is for timed runs. The timed runs and runs will also feature different networks which again could be easy to implement because I could use a random number generator to decide which form is loaded so that I could have five forms assigned to runs and timed runs and then each one of these corresponds to a number and if the random number generator produces the form’s corresponding number then it will show that form. I will also just have to copy the code I currently have for the Run and Timed\_Run forms and then paste it into these new forms so that then the only major change would be the networks on each of the forms.
* Instead of saying “However for the exam you will have to draw the minimum connector separately from the network given.” when someone completes a run of an algorithm the program will now display this “Now try again but instead draw out the minimum connector on paper using Kruskal’s algorithm and then check your answer using the Run feature”. This is so that students get more out of using the Run form.
* The program could be changed so that if two or more arcs have the same arc weight then the user would still get the choice correct no matter which one they chose in any order. However this would be quite complicated to implement.
* I will also implement a stopwatch which displays the time in this way 00:00:00 instead of 0:0:0 because it is more conventional and it is easier for students to understand. This will be done so that when millisec = 0-9 or sec = 0-9 or min = 0-9 then they will always have a “0” in front of this number when displayed on screen.
* Even though it may be problematic I will try to stop the program from creating a blank record each time the Update\_Kruskal and the Times\_Achieved forms are opened; as I believe it makes my program look less professional.
* I will also reduce the number of characters that a person can enter for their forename to 25 characters so that users with long names won’t have their time achieved moved out of line with the others as much. However I have noticed that to prevent this problem fully then the layout of both the Update\_Kruskal form and the Times\_Achieved form must be changed meaning that the list boxes must be made larger and the form so that the times achieved can be moved further over in the list box.
* At the end of a run the program displays the minimum connector in word form as well as on the network and its weight; which will be displayed at the bottom of the screen once finished. This will look something like this:

Minimum connector = CD, AC, GF, CB, DE, EG Weight = 23

* However at the end of a timed run the program asks the user to input the minimum connector and its weight. This is so that the student is tested on their knowledge of Kruskal’s algorithm more like they are in the exam. Text boxes and labels will be used so that the user can enter what the minimum connector is and its weight. This will look something like this:

Minimum connector = , , , , , Weight =

The user will have to type each of the six arcs in the minimum connector into each of the six text boxes; in the form that they will have to write it in the exam.

These improvements to the Run and Timed\_Run forms could be implemented very easily, for example in the runs the minimum connector and its weight would be the same every time meaning that I could just output these two answers to screen. However if the arc weights were randomized due to the improvement to the Run form mentioned earlier then the way to implement this would be the same as in the Timed\_Run form; except that the user wouldn’t need to enter anything. This would be done by using the textboxes so that it depends on what position1, postion2, position3, position4, position5 and position6 equal. In the Timed\_Run form this code will be written so that it doesn’t matter which arc is put into each textbox but instead it matters whether the correct arcs that are in the minimum connector are put into the textboxes in any order. The program could then mark whether each one is correct or incorrect by displaying picture boxes with a tick or cross in them. If the weight entered and minimum connector entered by the user are correct then the stopwatch will stop and they will be taken to the Forename\_Entry form after clicking on the “OK” button on the message box; which says that they have completed the timed run. However users will only be able to enter the minimum connector and its weight when the user has produced the minimum connector on screen. In the Run form the minimum connector and its weight will just be output to the screen in labels.

* For runs instead of having message boxes appear after every stage of the algorithm the user would just have to press the “Enter” key to move on to the next stage and press the “Backspace” key to go back a stage. Therefore I could then display text to help the user understand what the algorithm is doing better. This would probably be quite difficult to code because I would have to write a lot of code so that it recognised the “Enter” and “Backspace” keys being pressed and then once the “Enter” key had been pressed the code would have to cause the algorithm to move onto the next stage and if the “Backspace” key had been pressed the code would have to cause the algorithm to go back a stage. Therefore I would have to get a lot of help with this and it would take a long time to implement.
* Also I want to improve my program so that it covers more of the D1 syllabus, therefore I am going to try and get my program to perform Prim’s algorithm network form, Prim’s algorithm tabular form and Dijkstra’s algorithm. This will require four forms for each algorithm like with Kruskal’s algorithm at the moment. Prim’s algorithm network form will be the easiest to code because it is the same as Kruskal’s algorithm except that the smallest arc weight to go into the minimum connector would be chosen from the arcs that branch from the starting vertex and then the next arc that is in the minimum connector is the shortest arc from the next vertex and so on until all six arcs are in the minimum connector. The other two algorithms will be more difficult to code but I will have to draw a table for Prim’s algorithm tabular form whereas with Dijkstra’s I could just use the same network/networks as for Kruskal’s algorithm and Prim’s algorithm network form.
* On the Run form I will make the arcs on the network as large as the ones on the Timed\_Run form, meaning that the two networks will look the same in terms of layout and size.