



BID FOR PROJECT 20

Project 20 × GRP Team 10

Animation of Sorting Algorithms
and their Correctness

[20] Animation of Sorting Algorithms and their Correctness

Project Title	Animation of Sorting Algorithms and their Correctness
Team Members	
Name	Email Address
Ruizi Han	biyrh2@nottingham.edu.cn
Shiliang Chen	scysc1@nottingham.edu.cn
Yuting Jiang	scyyj2@nottingham.edu.cn
Yijie Lu	scyy19@nottingham.edu.cn
Yiming Tang	scyyt1@nottingham.edu.cn
Yani Huang	scyyh7@nottingham.edu.cn

Contents

Part 1: Introduction.....	2
1. Abstract	2
2. Team understanding towards the project	2
● Description.....	2
● Basic requirement.....	3
● Innovation.....	4
3. Competitive products analysis.....	4
4. Introduction of the team.....	6
Part 2: Working methods and teamwork skills	7
1. Technologies and methods:.....	7
2. Team Management:.....	8
Part 3: Project estimate	10
1. Simple prototype	10
2. Timeline	11

Part 1: Introduction

1. Abstract

The project, developing an open-sourced software, which helps students visualize and understand sorting algorithms and their correctness, is promising. It has a potential and broad market which is focusing on helping people understand and be interested with the sorting algorithm. Nowadays computer science has become a popular subject to society, so this software may be an entry level for the newcomers.

The team has carefully examined and researched the project and holds a part of knowledge towards the project up to now. After two weeks of running-in with team members, the team is working gradually as an efficient team. Through many discussions towards what the project is and how to design the software, although there is still lots of knowledge to learn and tons of difficulties to conquer, the team is firmly ready to take over this project.

2. Team's Understanding towards the Project

● Description

Inspired by some examined children's programming software, for example **Scratch**, the team is considering designing the project software like a programming game, to help users understand principles of sorting algorithms. Firstly, the principle of the sorting algorithm is based on the loop algorithm such as "for" loop and "while" loop. However, because considering users may not learn about the sorting algorithm, the program language will be complex. There are some ways to achieve the same meaning as the language, which can be similar to Pseudo code. For example, "While" can be described as until the condition of the restriction, the same thing should be repeat for n (a providing number) rounds.

Therefore, one possible game might be provided word blocks which can be formed as a completed command like a while loop. The Preliminary design of game will be like: choosing one sorting algorithm module (selected one from several types of sorting algorithm)-> the system firstly display how the disorder elements will be sorted (animation) -> users drag blocks to finish a command -> attempt to run the command -> if the elements all switch to the right place, some interesting animation will be displayed to encourage users. Otherwise, everything goes back to the beginning. This game trick is based on the Scratch.

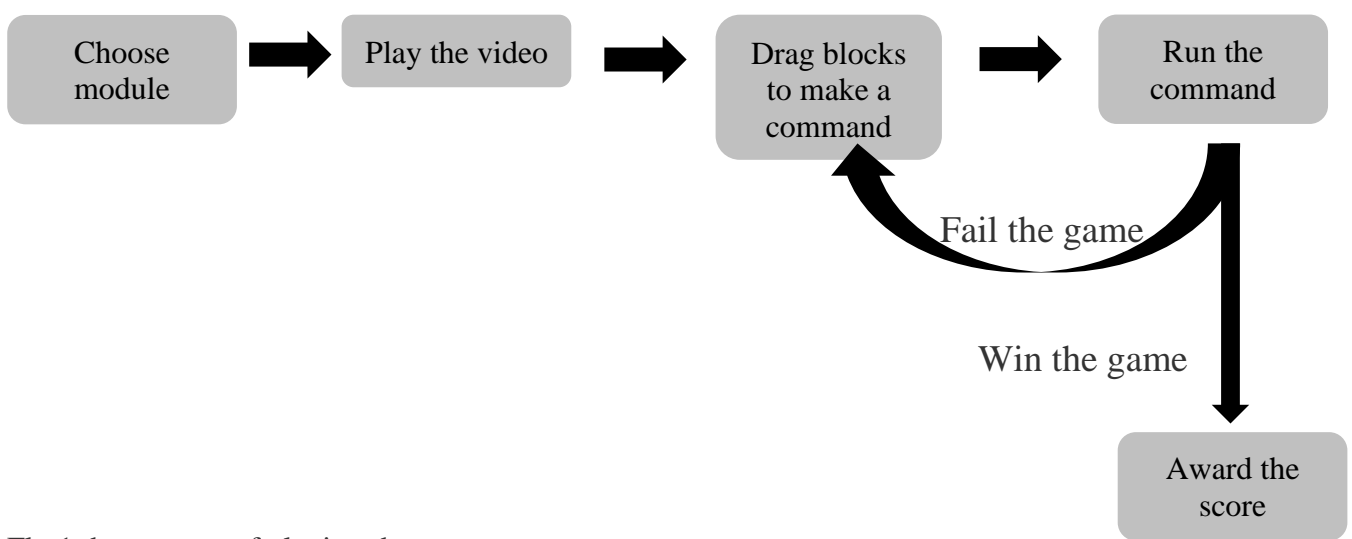


Fig.1 the process of playing the game

● Basic requirement

Non-functional

1. Each user has single account for saving his or her own model for the next use.
2. Two ways to set initial state: elements are created randomly by the system or provided by the users. Both can be sorted by order after commanding.
3. Users can freely see the animation of the process of sorting algorithm step by step. They can choose to backward, play and pause the process at any time.
4. Users can drag the blocks and have chance to make it to a command.
5. If users have problem to deal with it, they have three times to see tips which can guide them.

6. Feedback system. Users can give the feedback during using, programmers can receive that for better promoting.

Functional

Several ways to make the sorting algorithm visualizable: there are several model containing different ways to display how the algorithm sorting the disorder elements. for children: cartoon animal character with different size such as cat, tiger, elephant. For adult: the rectangles with different height, the circles with different size and color, the scatted dots, the bars with different pitch (more ornamental).

● Innovation

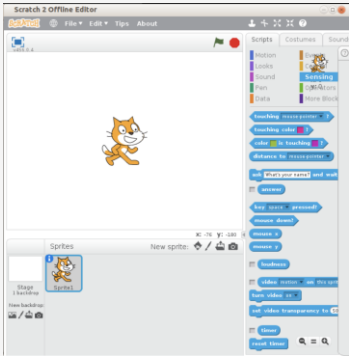

1. Differentiated with other existing products, the animation will be displayed more smoothly, instead of, barely switching disorder elements.
2. The software has various types of visualization. Users can choose the one they prefer.
3. The software can be used world widely. Users can set the language according to their preference.
4. The software does not use boring theory to teach users but use games to inspire them and guide them to understand and prove the sorting algorithm by themselves.

3. Competitive products analysis

In this part, the team will examine the most 2 typical competitive products, that is, Scratch and <https://visualgo.net/en/sorting> (Hereinafter referred to as "Visualgo"). Scratch is actually a block-based visual programming language, particularly targeted at children, to help them study programming. Although it is not really specialises in sorting algorithm, The team still considers it to be a competitive product is because it is focusing on teaching programming algorithms and ideas, which is closely similar to what the project aims to achieve.

5 Visualgo is a website project, doing exactly the same thing with this project. It is web-based, teaching and allowing the user to explore sorting algorithms by themselves.

The table below is comparing them with their advantages and disadvantages.

Product	Scratch	Visualgo
Image		
Advantages	1. Users who do not have program experience can still understand the rule. 2. Users can not only learn the basic knowledge of programming but gain fun from it.	1. Clearly explains the algorithms. 2. With high degree of freedom, users can learn more knowledge through autonomous learning.

Disadvantages	<ol style="list-style-type: none"> 1. The animation might be a little simple. The expression and fluency of it could be improved. 2. Sometimes it may only give one solution of the algorithm, the space of execution is small. 	<ol style="list-style-type: none"> 1. Web-based tool is not actually software. 2. UI is too ugly to be in line with modern aesthetics.
---------------	---	--


Fig.2 the analysis of similar products

After analyzing those products, the team considers the project as an executable software, rather than a simple tool. Researched competitive products are mostly websites, which in the teams' opinion they are not open-source software at all. In contrast, the team decided to build an independent software. The team chooses to extract the essence and discard the dross of them, and develop a modern, learning-friendly and simple-to-use software.

4. Introduction of the Team

1. Team members' brief introduction and CVs

- Yuting Jiang is a real team worker and has experience in every phase of software engineering. She is skilled in many aspects, such as producing UML and plan writing and she achieved a high score in FSE.
- Shiliang Chen and Ruizi Han used to work together in the same team on a machine learning topic. They are also good learners and leaders.
- Yijie Lu has good experience in data visualization.

- 
- Yiming Tang is a designer, has 2 years of designing experience and produced advertisement for Clinic and The Hub of UNNC. He is good at making animations.
 - CVs: see Attachment 1: Team members' CVs.

2. Programming language skills

Members can all apply C, Java, Python, Matlab, HTML, MySQL adroitly.

3. Teamwork experience

Most of members have participated in the summer research, and some members have extra projects outside the school. Additionally, in the previous project, everyone has got good feedback from team partners or sponsors.

Part 2: Working methods and teamwork skills


1. Technologies and methods

In this part, it will discuss what technologies the team will use to finish the project.

The team have decided to develop this software mostly by high-level language, i.e. Java, Python, and supported by scripting languages such as JavaScript.

After feasibility study of various languages and development processes, the team will mainly use high level language such as Java or Python.

For high-level languages, the team's research and experience show that although they run much slower than underlying programming languages such as C or C++, they are more effective, as their grammars are much simpler, the mechanisms are also more abundant, which can save developers' time. Another considerable advantage is that they have rich graphic libraries to learn and use. Python as example, there are Pyglet,



Pygame and much more libraries to use, to the animation part. They also have strong community, meaning that we can deal with problems through the Internet fastly and conveniently.

At the same time, the team is considering finishing the animation part by the web front-end technologies, including HTML5, CSS and JavaScript. These technologies are still the simplest front-end skills in the world, producing gorgeous animations with a few lines of code compared to other traditional skills.

In conclusion, since members have learned Java in PGP and most of modules this semester would use Java, the team may consider Java more in this project. The team shall discuss this more, consider more situations, and make a final decision on which programming language to use for this project.

2. Team management

1. The whole project will be carried out by six people. Since it is a small development task and we hope to have great customer involvement, Agile project management approach will be used to embrace changes to requirements, delivers and frequent releases.

Specifically, we will use Scrum method utilizing backlogs as a formal “to do list” which contains a set of tasks to trace work. MSFT Planner may help members maintain backlogs.

During the process of development, Sprints will be planned based on the backlogs made during the meeting. They are two-week development iterations helping to complete subtasks weekly.

Except formal meetings with the sponsor, informal meetings stand-ups would be held weekly to make sure the efficiency of our team.

In order to ensure the code quality, systematic ways will be taken. Initial thing would be confirmed such as comment, naming, indentation, and change log. Git will be used as a version control tool. What is more, software would be developed using a test-driven development approach while appropriate and pair programming will be deployed to mostly avoid mistakes. As the process going on, team would manage by several useful GitHub features. The plan is to use issues with labels to raise questions, distribute tasks, alert bugs, show what to-do is, Doing and Done. Along with Boards, a very clear visible feature for managing tasks while clearly showing the whole process, Milestones are also used for making stage-based objectives with due time to control the productivity. A brief early-stage management plan is as follows,

Stage one:

- Analyze requirements and do background search
- Write user stories and personas according to the requirements
- Design prototypes with collaboration to customer
- Outline planning and architectural design

Stage two:

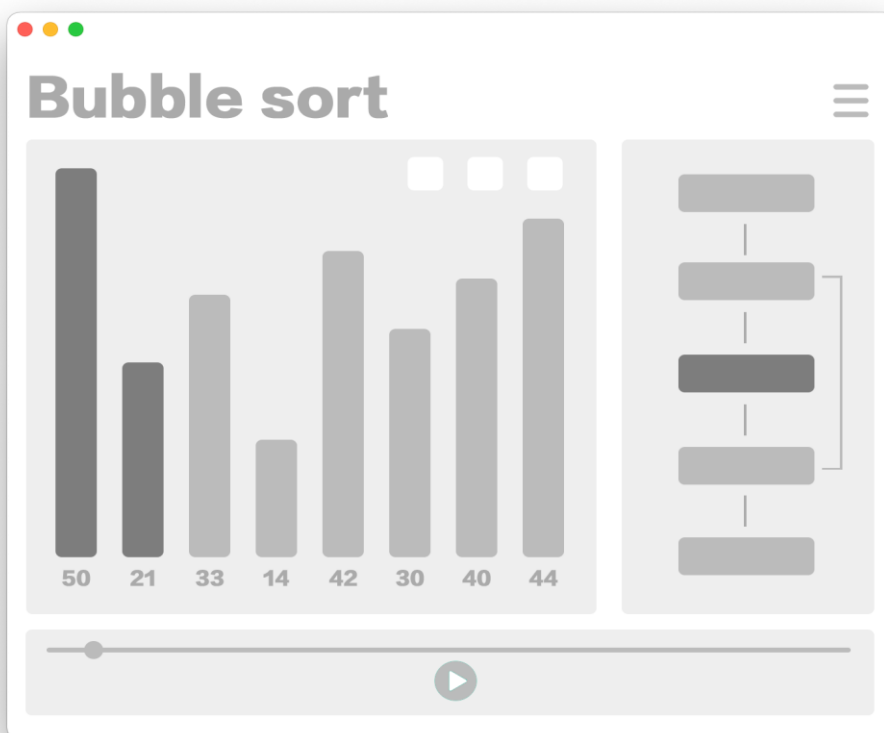
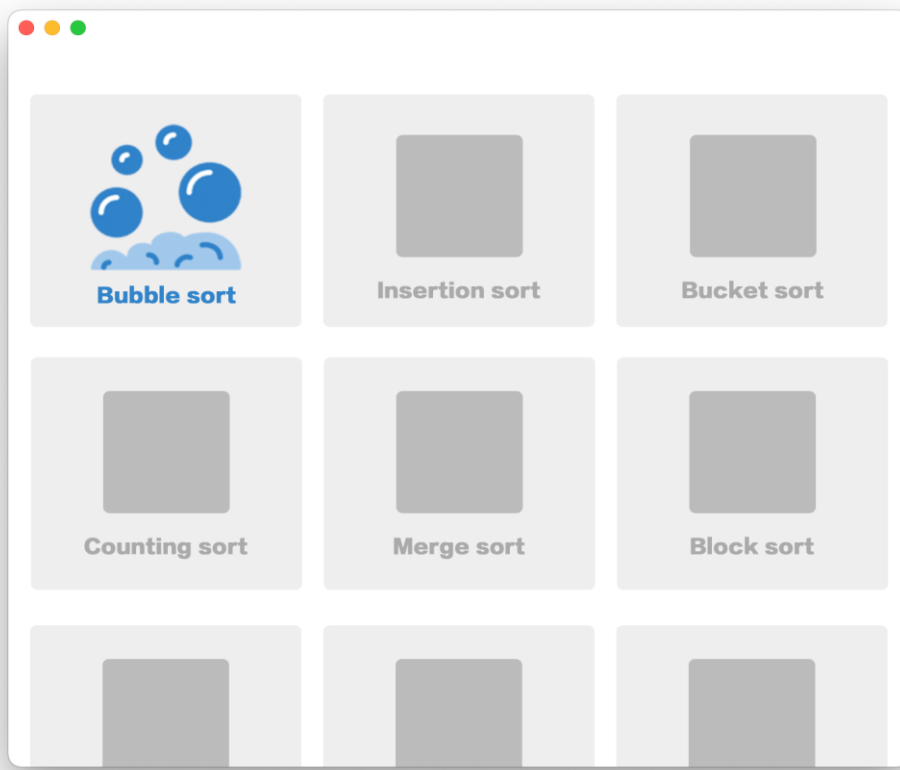
- Divide members in pairs and distribute works to pairs.
- In each sprint, develop with testing and customer
 - a. Access b. Select c. Develop d. Review
- Daily informal meeting and weekly formal meeting with supervisor

Stage three:

- Project closure and evolution

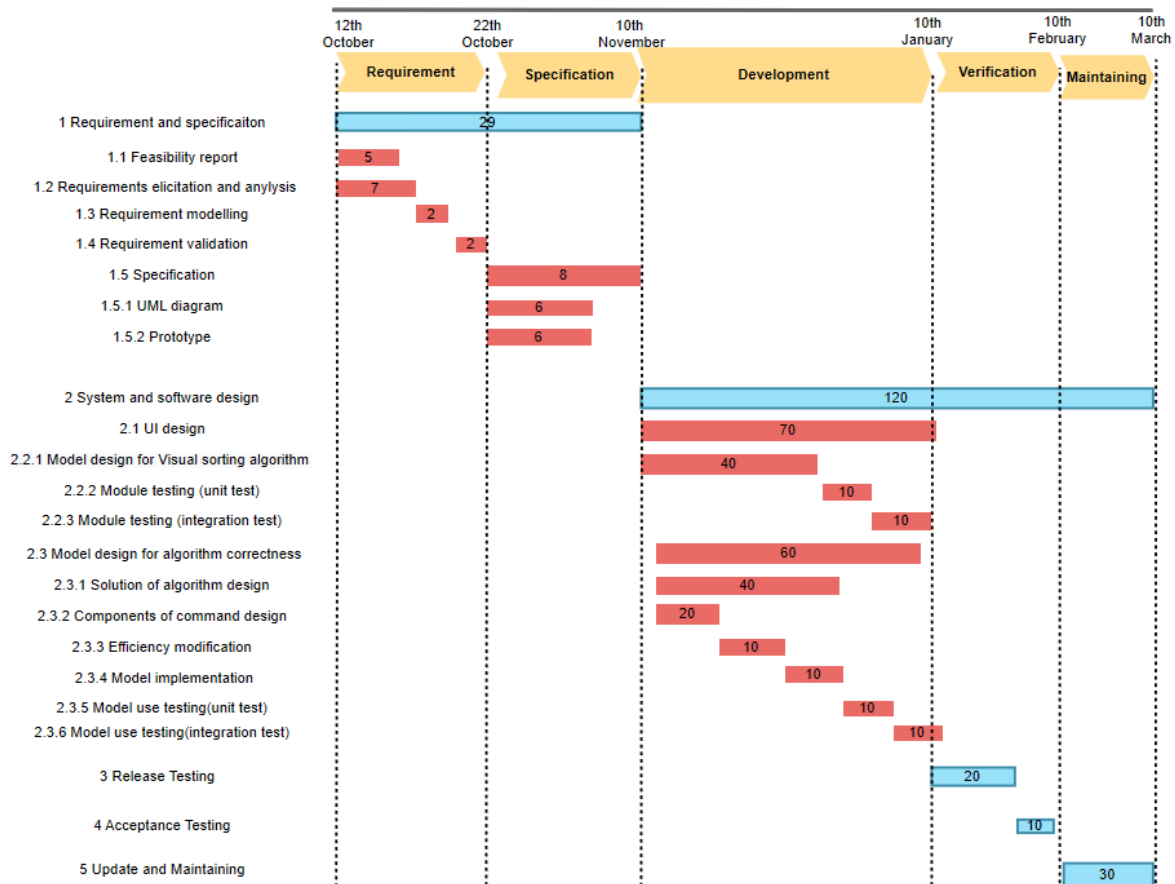
Part 3: Project estimate

1. Simple Prototype



Flg.3 Prototype

2. Timeline



Flg.4 Timeline

Description:

- **Before Thursday, 29 October 2020 (requirement and specification):**

The first process will have the two agendas, most parts focus on the requirement and specification. Firstly, the more detail of requirement should be checked, **a complete requirement with functional and non-functional command** should be made by collecting information from surveys, interviews, and other available methods. This draft should be confirmed in the team members' last meeting. After that, the requirement will be discussed with the sponsor, the expected result is that the requirement fits the users' need. What the next is the specification process, the plan is that the specification can have **the low-facility prototype, suitable diagrams**. While a completed team website will be made to hold and track the rate of the process.

- **Before Thursday, 10 December 2020 (interim reports)**

- **Before Thursday, 9 April 2021 (software and final reports)**

During this period, developing the software will be the first thing to do.

Preliminary plan is to use Agile techniques which might have several circles, and the goal of every circle will depend on the components of the system. Each circle will contain developing, reviewing, assessment, selecting. To make it clear, developing is the process to programming the system. Reviewing is contributed by the frequent meetings, and they might be held on Thursday every week. Additionally, when it comes to assessment, the unit testing for each circle will be executed and integration testing will be done for every subsystem. At the end of each circle, a simple report could help to collect a recent potentially shippable product.

When the whole system is finished, system will be modified by the release testing, and acceptance testing after some volunteers using. The report should be made as the same as the previous state.

- **On Tuesday, 13 April 2021 (software demonstration, team presentation, team promotional digital artefact)**
- **On Wednesday, 14 April 2021 (open day, presentation day, team live Q&A)**
- **Before Monday, 19 April 2021 (individual final reports)**