Design Document

Introduction

This document describes the design of project that is part of employe selection at Scala Computing. The project is to calculate the fastest time to cross bridges of a group hiking in a jungle.

Problem Description

A group of hikers going through a jungle and has to cross several bridges. Each bridge can only accommodate two people, they need a torch to show the path on the bridge. When two hikers cross a bridge, they can only the speed of the slower guy. One of the hikers that already crossed might have to come back to pick up hikers that haven't crossed yet.

Algorithm

The solution to this problem is to find the fastest hiker in the group. The reason is that almost half of the time crossing the bridge is spent when the lone torch bearer goes back to pick up another hiker. The pseudo code of the solution is as follows:

```
•  Intiliaze a container that is holding hikers with the information
   about the name, speed, and bridge where s/he joind the grup.
•  Initliaze a map (hash table) that map bridge name to its length.
•  Initialize a list of bridge in sequence where they have to be
   crossed

   For each bridge to be crossed:
   -  Group hikers that cross in this bridge to a list.
   -  Find the fastest hiker
   -  For each hiker in the group:
          ▪  Calculate time he takes to cross
          ▪  Calculate time the fastest hiker going back
          ▪  Add the above
          ▪  Add number the previous step to total time
   -  Substract the time the fastest hiker takes to go back (since
      he does not need to go back after crossing with the last
      hiker)
```

Time and Space Complexity Analysis

Time complexity for the algorithm consist of fixed part and variable part. The variable depends on the number of input. Processing data from yaml file definitely will be the same of whatever methodology we use to solve the problem. We loop over the number of bridge to be crossed. Assume the number is M. If we look at the inside the loop, there are two separate loop, each goes over list of hikers. Assume the number of hiker is N. So the total time complexity would be: M x 2N ∞ MN.
Space complexity for this algorithm can be measured by the two lists (person and bridge path), dan a map of bridge. In addition, we have a list that consists of hikers that are to cross a particular bridge. The original data structure is to be same across algorithm since they contain data from yaml file. This algorithm creates new list (one list) of hikers that are to cross a particular bridge. This is in addition to orginal data obtained from the file. So total space complexity is N.

Testing

For testing purposes, I have created a program that wait for yaml file and perform calculation on getting the file name. We can test quite a few things with this setup:

1. Testing invalid file name.
2. Testing invalid yaml specification.
3. Testing border cases:
    - Yaml file is empty
    - Only one hiker and one bridge
    - More bridges than the brdge to be cross (there is a bridge outside the route)
    - Hiker with bery high speed

I also provide debug flag to enable a print out of data structure if needed.

Error condition and handling of exception

The program has considered several error and border case. Those include:

- Bad ymal file name
- Bad ymal format
- Zero speed: we ignore the hiker that has zero speed, and continue with the next hiker.

Code Structure

The code uses yaml open source code to parse yaml file. The code can be obtained from https://github.com/jbeder/yaml-cpp. The build is done using cmake. The following are the file structure of the project:

```
project
   |
   |_ CMakeLists.txt
   |_ main.cpp
   |_ bridge-cross.yaml
   |_ third-party
         |_ yaml-cpp
               |_ (... contents of yaml-cpp git repo ...)
```

I will put in the attachment steps to make a build. The main code that I developed is main.cpp. There is one simple class (Person) that is declared and defined in main.cpp because the class is small and simple enough. The class declaratioan ideally is put it Person.h, while the definition goes to Person.cpp. This is done later on when project and the class become more complex.

Conclusion

The project is to dsign and code problem of group of hikers crossing bridgs with certain limitations. Part of the project to get open source code for ymal file processing. The project is done in allocated time. There are a few things that can be improved if time permitting, including:

- More granular exception handling
- More testing to onsider all possible error in input and logic.

Attachament A
How to build the project
The step-by-step guide can be at the following web-site
https://www.reddit.com/r/cpp_questions/comments/bdxcp0/im_a_beginner_to_c_help_me_install_and_use/.