**C++ Programming Final Project Proposal**

Charlie Comeau 龚成荫 2019403371

1. **Problem description:**

The world is a magnificently large, diverse, and complicated place. As an exchange student from America at Tsinghua University, this semester I have had the pleasure (and the burden) of experiencing what exactly that really means. However, I have also had the pleasure of discovering that there are more things that unite us than divide us. While food comes at a close second – in my opinion, money and what it can buy commands the most care and attention of everyone I have met both here and at home.

On a macro level, money is an unspoken yet easily understood idea that transcends all languages, political borders, ethnicities, and backgrounds. It is one of the most powerful motivations known to man, and its importance to our society is like that of water to agriculture. Just like how plants need water to live, humans need money to purchase and fulfill their own needs. In short, the importance of money simply cannot be overstated in today’s world.

However, many people still struggle with money. For a country that proudly defines itself as the richest nation in the world, a surprising 49% of Americans currently live paycheck-to-paycheck with no money left over after spending for savings. In other words, about half of America is just one medical emergency away from serious debt. Among the young adult population, those figures are even worse and hover around 70%. Additionally on a personal level, many of my friends are unfortunately the human representation of these money-tight statistics.

On the other hand, China has had a remarkable and an unprecedented economic growth within only a few generations, managing to increase their GDP by about 44,500% in the past 60 years. Clearly, money is and has been on both country’s minds. While the causes of both these figures are quite fascinating and heavily debatable, I believe they reinforce that the importance of managing money also cannot be overstated and there exists a massive and statistically significant need for this skill. This is why I would like to design an online budget through C++ for my final project.

As aforementioned, young college students are part of the statistically worst group about money. As a result, there exists a wealth of resources online designed to help students budget their money. Even though my program will most likely pale in comparison to theirs, it is my hope that I can help someone (such as myself) be smarter about how they choose to spend their money.

1. **Fundamental Design**

This program will attempt to break spending decisions into three different categories: fixed, variable, and savings. Savings will be defined as the money left over after fixed and variable spendings.

As an additional option program, the program will offer to see how well the user’s spendings follow the 50/30/20 rule: 50% on necessities (mortgage, rent, insurance, etc), 30% on wants (fun money), and 20% on savings, or choose a different percentage for each category that adds up to 100 based on user preference.

For privacy, the user will first log in using a username and password, then enter their monthly after-tax income into the system. From there, the user will be able to choose a month, input all of their spendings accordingly, and to input how much money or percent they would like to save for that month.

For each spending input, the user can add a date, a title, a note, an amount spent. Additionally, the user will be able to delete and modify each detail of each spending input after being inputted. With every new input, the totals and percentages of the 3 spending categories and the total left-over, non-committed money will be updated accordingly.

If the user chooses to, they will also be able to see how much of their money should be allocated to necessities, wants, and savings based on the 50/30/20 rule or their own chosen percentages, and how much money they have that is left over for each category after each new spending input.

If the user uses all of their after tax income, then they can still add spending data but however they would be incurring more debt with every additional amount spent. As a result, the system will prompt a warning to the user when they have maxed their budget.

1. Login verification

The user chooses their identity to enter the system, and the system will need to perform a user and password verification in order to allow them in. Each user will have a different username, each user will only have one distinct password, and each user will only have access to their own budget data.

1. Budget information

* After-tax income
* User-determined percentages for fixed, variable, and savings
* Amount of money dedicated to each spending category (after-tax income \* percentage)
* Amount of money left over in each spending category (after-tax income \* percentage – sum of all category input costs)
* Total leftover money (debt if negative)
* Month

1. Spending item input information

* Date
* Spending category
* Title
* Amount
* Note

1. Design of interface

After opening the program, the user will have these options:

* Existing user sign in
* New user registration

After the user signs in, the user will have these options:

* Input after-tax income
* Choose target percentages for each spending category (default 50/30/20)

After the user determines that, the user will have these options when entering spending data into the system:

* Enter new data
* Modify previous data
* Delete previously inputted data
* View category amount totals
* View conditional calculations for target percentages
* Add more income to the existing after-tax income

Even after the user has budgeted all the allotted money, the user will still be able to add spending data into the system. This will represent the amount of money the user is in debt for that month. As a result, when the user has used all their allotted money, the system will give the user a warning and will ask whether they would like to continue adding spending data or not.

On a personal level, this will be a more advanced version of what I have done for myself in Excel this past summer while traveling in Europe. After a few very costly mistakes in the very beginning, I learned the hard way that Europe is much more expensive than America, and thus enormously more expensive than China. Out of self-anger and frustration, I chose to implement a basic budget for myself in order to keep a better track of my finances and be smarter with my money. However as a student, I did not have an after-tax income, therefore the goal of my Excel spreadsheets were not to budget using an externally fixed amount of funds, but to help me set reasonable expectations for various spending categories and to maximize my money in an expensive environment. And I am happy to report that after implementing my budget, there were no more costly mistakes made!