California State University, Sacramento
College of Engineering and Computer Science

Computer Science 35: Introduction to Computer Architecture

Fall 2019 – Project – *Gold Rush!*

Overview

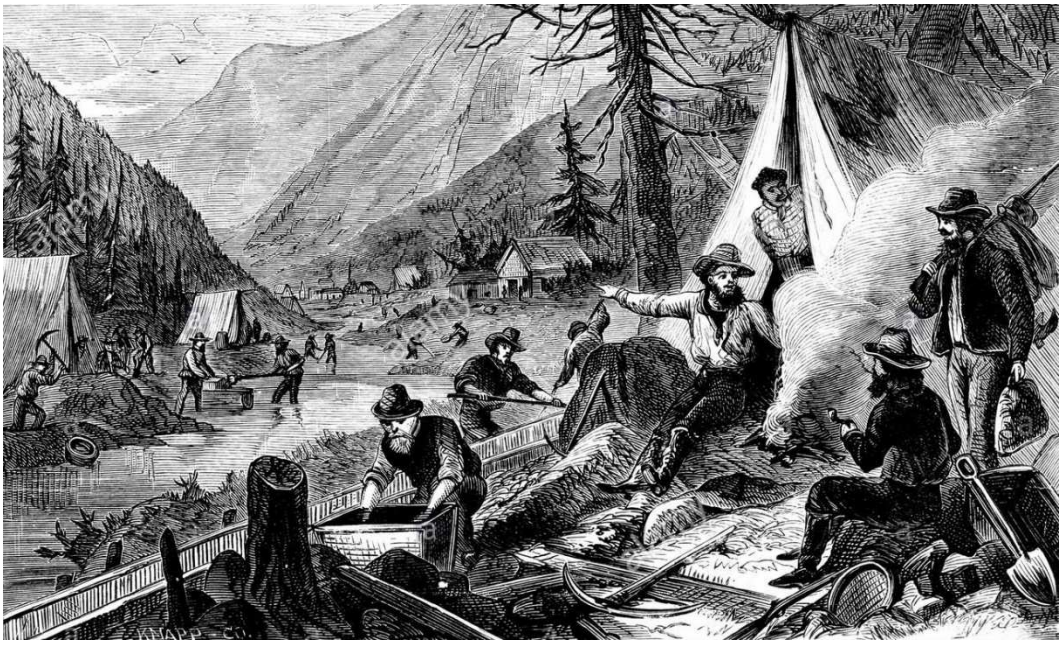
Back in 1848, California was a peaceful untamed wilderness. There is a smattering of people – who had recently fought a war for independence. A war named after the proud bears that flourish in the unspoiled lands.

The only attempt, to build a civilization, is in a small town called Sacramento. Under the management of John Sutter, the small town is growing at an impressive rate. So much so, that cut timber is needed badly. To solve this dilemma, John Marshall is hired to build a saw mill on the American River in nearby Coloma. … The saw mill uses use the power of the river to cut the nearby timber and then float it downriver to eager population.

Then, one day, something odd was seen in the river – GOLD!

This ignited the blaze that became the Gold Rush! A flood of "Forty-Niners" from all over the United States (and the World), arrived to seek their fortunes in the untapped streams and rivers of the Sierra Nevada.

Over the next 10 years California was transformed into a bustling metropolis.



Some Facts About the Gold Rush

1. Before the Gold Rush, California only had about 157,000 people. 150,000 were native-Americans and 6,500 "Californios" (Mexican citizens living in California). The rest were settlers from the United States. California was mostly uninhabited.

2. Gold was discovered at Sutter's Mill by James Marshall in 1848. Wood was needed to construct buildings in Sacramento and the saw mill was built on the American River (yes, the one right next to Sac State). The Mill is located in Coloma – a short distance from here.

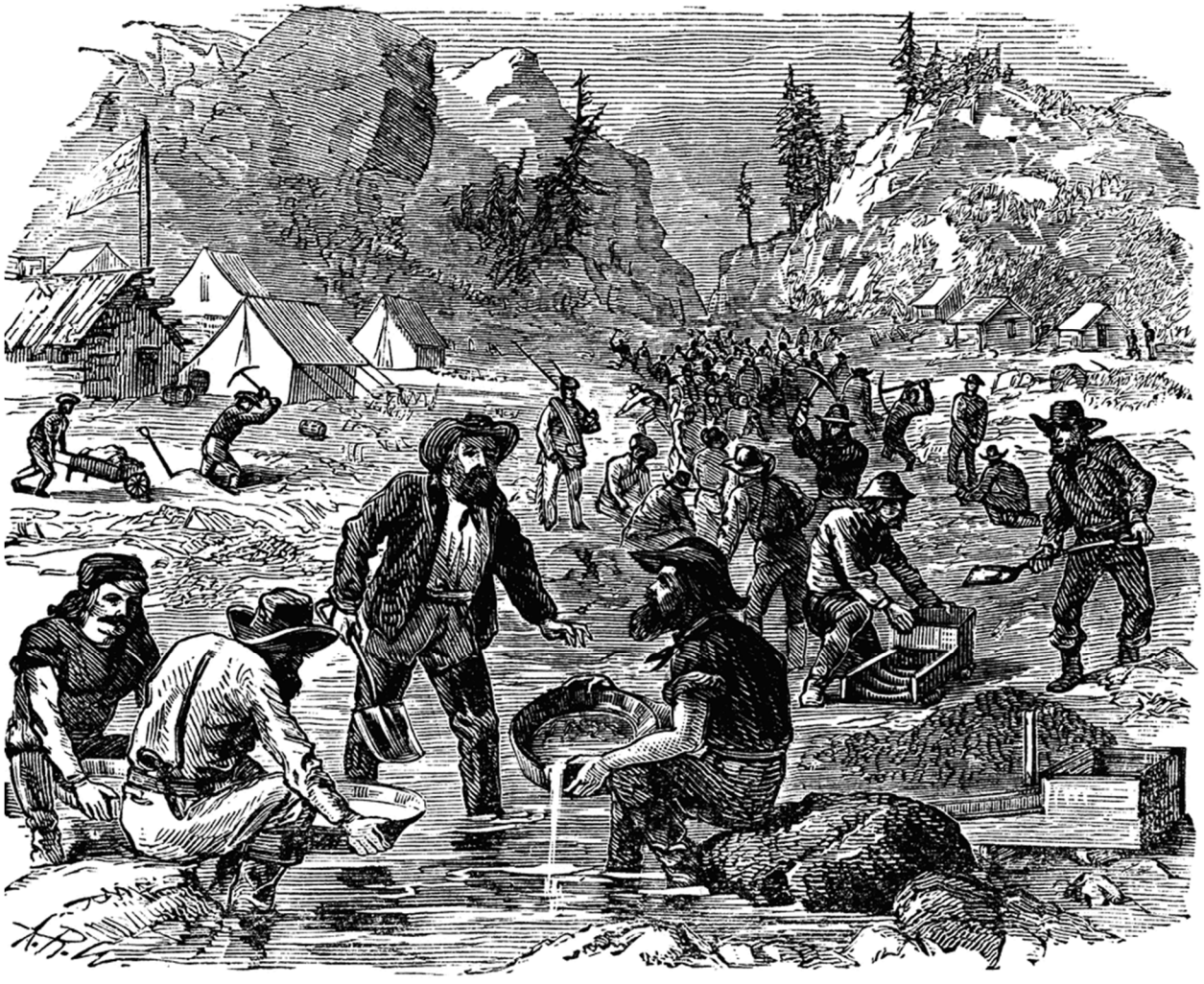
3. People began to arrive in Northern California in 1849. This is the origin of the term "Forty-Niner".

4. There were two routes that Forty-Niner's to follow to get to California. The first was the overland California Trail which traveled through the Northern Nevada desert. Modern-day Highway 80 follows this path. The second route was by ship. The destination was the Golden Gate (the entrance into San Francisco).

5. By the mid 1850's, more than 300,000 people has settled.

6. The Forty-Niners were very diverse. 25% of them came from outside the United States. The rest included religious and ethnic minorities from around U.S. 92% Forty-Niners were men.

7. An ounce of gold was worth over a thousand dollars. The largest single gold nugget was 160 pounds making it worth over 3 million dollars.



Gold Rush Technology

Pan: : This is the Forty-Niner's most basic, reliable, and recognizable tool. Usually made of tin or iron, the Forty-Niner would put water and soil into a pan and swirl it in a circular motion. The goal was to have the water carry the light particles out of the pan – leaving only heavy particles (rocks and gold).

Cradle : Also called a "rocker", the Forty-Niner would place water and soil into the top of the device and rock it left and right. As the water flowed downwards, horizontal ridges (called riffles) would capture the heavy particles.

Sluice: : A sluice uses the same basic technology, as a cradle, but is far more advanced. It is essentially a wooden channel (similar to a rain gutter) with one end elevated. The sluice was connected to a continuous flow of water – usually upstream. This liberated the Forty-Niner from manually adding water (as well as the laborious rocking) and, thus, allowed them to work on several at the same time.

Forty-niners with pans and a sluice.

The Game

Overview

Like the classic Oregon Trail (created in 1971), your game is going to create a simple simulation based on resource management. In this style of game, you attempt to maximize your score by using your resources and obtaining assets. In your case, you will be creating a (very simple) game based on the life of Forty-Niner.

The game takes place over a 20-week period. You will start with $100 (not bad for 1849). Your goal is to make as much money as possible. Each week, you gain and lose a random number amount of money (resource). You can spend your money (resource) to protect your assests. These, in turn, can help you acquire more money.

Each week can be broken into four logical sections:

Part 1: Sunday

Even though people from all over the World to California, the Forty-Niners, as a whole, took Sunday off. Some Forty-Niners rested, some did repairs, and others spent the day up to no good. So, each Sunday, you get to decide:

- Do nothing. It doesn't cost anything but doesn't help either.
- Repair the sluice – which brings it back up to 100%. It will cost you $100 to do so.
- Go into town (Auburn or Placerville) to enjoy yourself. Your endurance will go up from 5% to 50% (it might have been a great day!). However, it will cost you between $50 and $200 to do so.

Part 2: Profits

- You can earn between $0 and $100 automatically by panning for gold.
- Your sluice can earn you between $0 and $1000. However, if its durability is at 0%, then it is broke and doesn't produce any gold.

Part 3: Costs

- Food costs between $30 and $50.

Part 4: Wear and Tear

- Your endurance (the willpower to keep going) wears down from 10% to 25% each week.
- The durability of your sluice, which handles a heavy workload, wears down from 20% to 50% each week.
- Your gold pan doesn't wear down (it's made of tin and quite durable).

Sample Output

Your solution doesn't have to look exactly like this. However, this should show you the basic gameplay. For readability, the user's input is displayed in **red** and randomly generated values are in **blue**. You don't have to use color (unless you are going for extra credit). *As always, please feel free to change the wording of the text.*

```
====================
CALIFORNIA GOLD RUSH
====================

Rules:
1. 20 weeks (5 months)
2. Your endurance drops 10-25% each week. If it reaches 0%, the game ends.
3. Panning for gold yields 0-100 dollars.
4. A sluice yields 0-1000 dollars (durability drops 20-50% each week)
5. Food costs 30-50 dollars.

WEEK 1

You have $100
Your endurance is at 100%
Sluice is at 100%
It' Sunday! Do you want to 1. Do nothing, 2. Repair sluice (-$100), 3. Go to town.
1

Panning for gold yielded $30
The sluice yielded $50
You ate $32 in food

WEEK 2

You have $148
Your endurance is at 82%
Sluice is at 79%
It' Sunday! Do you want to 1. Do nothing, 2. Repair sluice (-$100), 3. Go to town.
1

Panning for gold yielded $46
The sluice yielded $862
You ate $44 in food

WEEK 3

You have $1012
Your endurance is at 72%
Sluice is at 42%
It' Sunday! Do you want to 1. Do nothing, 2. Repair sluice (-$100), 3. Go to town.
1

Panning for gold yielded $10
The sluice yielded $211
You ate $50 in food
```

```
WEEK 4

You have $1183
Your endurance is at 59%
Sluice is at 8%
It' Sunday! Do you want to 1. Do nothing, 2. Repair sluice (-$100), 3. Go to town.
2

You repaired the sluice to 100%.

Panning for gold yielded $67
The sluice yielded $277
You ate $44 in food
```

I removed a bunch of weeks

```
WEEK 20

You have $5367
Your endurance is at 4%
Sluice is at 33%
It' Sunday! Do you want to 1. Do nothing, 2. Repair sluice (-$100), 3. Go to town.
3

Going to town cost you $103
You regained 21% endurance.

Panning for gold yielded $67
The sluice yielded $337
You ate $34 in food

====== GAME OVER ======

You ended the game with $5634
Your endurance was 13%
```

Have Fun

Use your imagination. Your game doesn't have to be the Gold Rush. You can base your game on any fun theme that you want. But, only if you keep the same gameplay.

For example, here are some possible scenarios:

- Kittens
- Cartoon: Spongebob Squarepants, Rick and Morty, Archer, etc….
- Politics
- Movies: comedy, sci-fi, horror, etc…
- Video games
- Television programs
- Characters from a book
- etc…

Due Date

The assignment is due the **end of Dead Week**. I strongly suggest that you get to work on this assignment as early as possible. If you did well on your labs, it shouldn't take more than a few hours.

Mural in Century Theatres, Folsom

Tips

1. *Write you program in parts*

DON'T attempt to write the entire program at one time. If you do, you won't be able to debug it. Experienced programmers use incremental design. Make a basic program and, very slowly, add the features you need.

So, first get the main loop working… then, bit by bit, add the rest of the functionality.

If you get stuck in an infinite loop, you can press Control+C to exit any UNIX program.

2. *Random Numbers*

The library has a built-in subroutine called "Random" that you must use to make your project work. Please read the documentation on how to use it.

Requirements

YOU MUST DO YOUR OWN WORK. DO NOT ASK OTHER STUDENTS FOR HELP.

If you ask for help, both you and the student who helped you will receive a 0. Based on the severity, I might have to go to the University. Any project that permits multiple sluices will receive a zero.

1. **Print the title of your program.** **(5 points)**
2. **Print the game rules. Let the player know how the game works!** **(5 points)**
3. **Loop for 20 weeks.** **(10 points)**

   If you change the project theme, please feel free to change this value (e.g. 12 for months)

4. **Part 1: Decision** **(15 points)**

   Input the player's choice. There needs to be at least 3 choices. The program must do different things based on the input.

5. **Part 2: Profits** **(15 points)**

   Your program must calculate how much they earned automatically (panning) and how much they earned for each asset (sluices).

6. **Part 3: Costs** **(15 points)**
7. **Part 4: Damage** **(15 points)**

   Decrease both the endurance and the sluice.

8. **Comment your code!** **(10 points)**
9. **Proper formatting:** **(10 points)**

   Labels are never indented. Instructions are always indented the same number of spaces. Add blank lines for readability.

Extra Credit

1. ***Color – 5 points***

   Make use of color to enhance your game. The color must be meaningful – don't just set the color at the beginning of the program.

2. ***ASCII Art – 5 points***

   Use ASCII-art to make your program exciting. The ASCII-art must be meaningful and not something overly simple like:

   ```
   ==========:)
   ```

   It's a happy worm!

3. ***At least 2 more decisions – 10 points***

   There is more to do than go to town and repair sluices. Give the program more decisions that can help or hinder the game.

4. ***Another resource – 10 points***

   Right now, there is only three resources – money, endurance, and the sluice. What else can be considered? Food? Happiness? Whiskey?

5. ***Random events – 5 each for a max of 15.***

   What other types of events, good and bad, can occur. The more you add, the more you can capture the feeling of the Gold Rush.

Submitting Your Project

Run Alpine by typing the following and, then, enter your username and password.

```
alpine
```

To submit your lab, send the **source** file (not a.out or the object file) to:

```
dcook@csus.edu
```

How to Connect from Home

Step 1 – Windows

The three servers that we use to do our labs cannot be accessed from off campus – at least directly. To connect these computers, first connect to Athena using Putty.

```
athena.csus.edu
```

Step 1 – Macintosh

Open the Terminal program. This is the same UNIX prompt that you get when you connect to Athena. Mac-OS X is a version of UNIX. Neat! Once at the prompt, type the following where ***username*** is your ECS username.

```
ssh username@athena.csus.edu
```

Step 2 – Secure Shell to SP1, SP2, or SP3

Once you are connected, you need to Secure Shell (SSH) to the SP computers. Basically, you will connect to Athena and it will connect you to the SP computer. Type the following at the UNIX prompt (this example uses SP2).

```
ssh sp2@ecs.csus.edu
```

You will have to enter your password again and (maybe) have to manually type "yes".

UNIX Commands

Editing

Action	Command	Notes
Edit File	**nano** *filename*	"Nano" is an easy to use text editor.
E-Mail	**alpine**	"Alpine" is text-based e-mail application. You will e-mail your assignments it.
Assemble File	**as -o** *objectfile asmfile*	Don't mix up the *objectfile* and *asmfile* fields. It will destroy your program!
Link File	**ld -o** *exefile objectfiles*	Link and create an executable file from one (or more) object files

Folder Navigation

Action	Command	Description
Change current folder	**cd** *foldername*	"Changes Directory"
Go to parent folder	**cd ..**	Think of it as the "back button".
Show current folder	**pwd**	Gives a file path
List files	**ls**	Lists the files in current directory.

File Organization

Action	Command	Description
Create folder	**`mkdir`** *foldername*	Folders are called directories in UNIX.
Copy file	**`cp`** *oldfile newfile*	Make a copy of an existing file
Move file	**`mv`** *filename foldername*	Moves a file to a destination folder
Rename file	**`mv`** *oldname newname*	Note: same command as "move".
Delete file	**`rm`** *filename*	Remove (delete) a file. There is **no** undo.