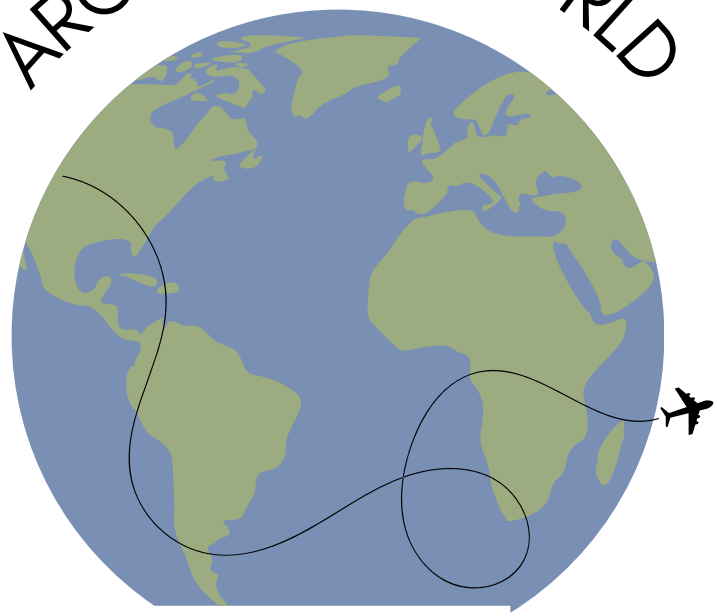
AROUND THE WORLD



[redacted]

BIRD TOUR

Pau, France

The Pyrenees Mountains are home to 90% of Europe's vulture population

Tokyo, Japan

The mejiro forage for nectar from blooming cherry blossoms

Toucans are my favorite bird.

Poconé, Brazil

Despite their appearance, Shoebills are docile to humans. These birds are terrifying and I would love to meet one.

Eastern Province, Zambia

The Wiradjur called these birds "guuguubarra" after the sound they make, now we refer to them as kookaburra

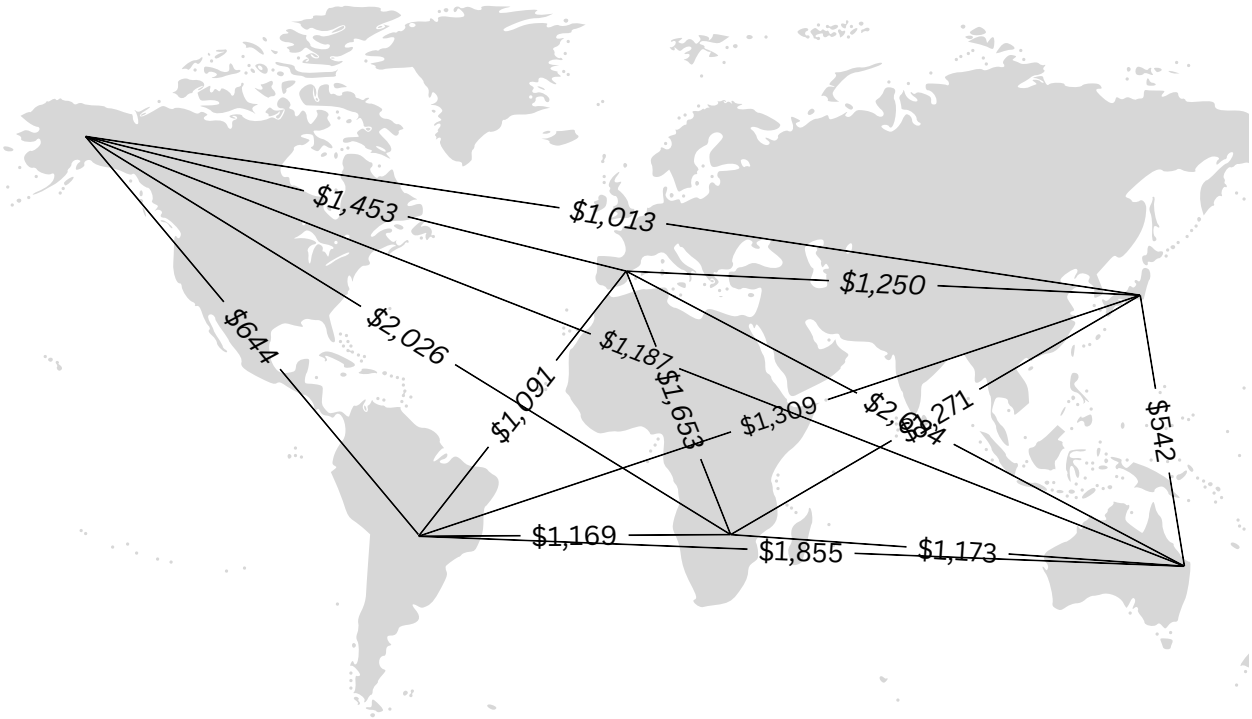
Beerwah, Australia

I really like birds. I know that we did not have to pick a justification for our trip around the world but it made it more fun for me.

COST OF THE TRIP

After we choose our destinations, we need to research the cost of a plane ticket from each destination to the other. It's best to organize this information in a table before plotting it onto a graph. Imposing your data onto a map is not not the most efficient but we will optimize our graph in the next section.

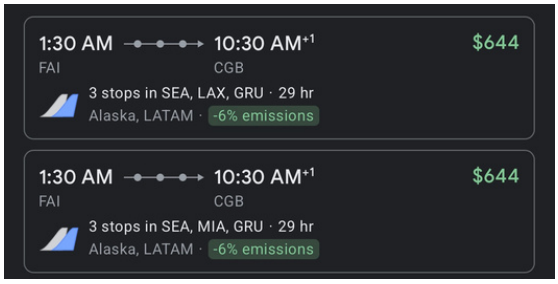
	PAU	TOKYO	ZAMBIA	BEERWAH	POCONÉ
FAIRBANKS	$1,453	$1,013	$2,026	$1,187	$644
PAU		$1,250	$1,653	$2,684	$1,091
TOKYO			$1,271	$542	$1,309
ZAMBIA				$1,173	$1,169
BEERWAH					$1,855

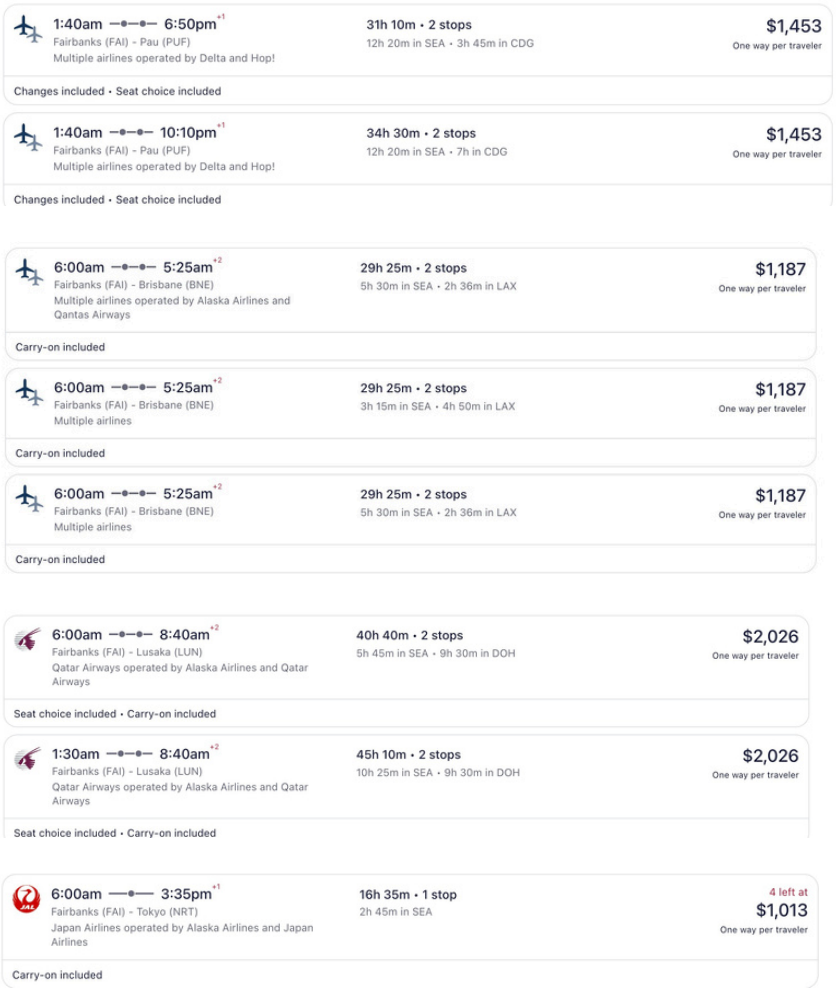


Some places I'd like to go are remote, for consistency I'll use the airports closest to my desired destinations.

Fairbanks (FAI), Pau (PUF), Tokyo (NRT), Zambia (LUN), Beerwah (BNE), Pocone (CGB)

SOURCES





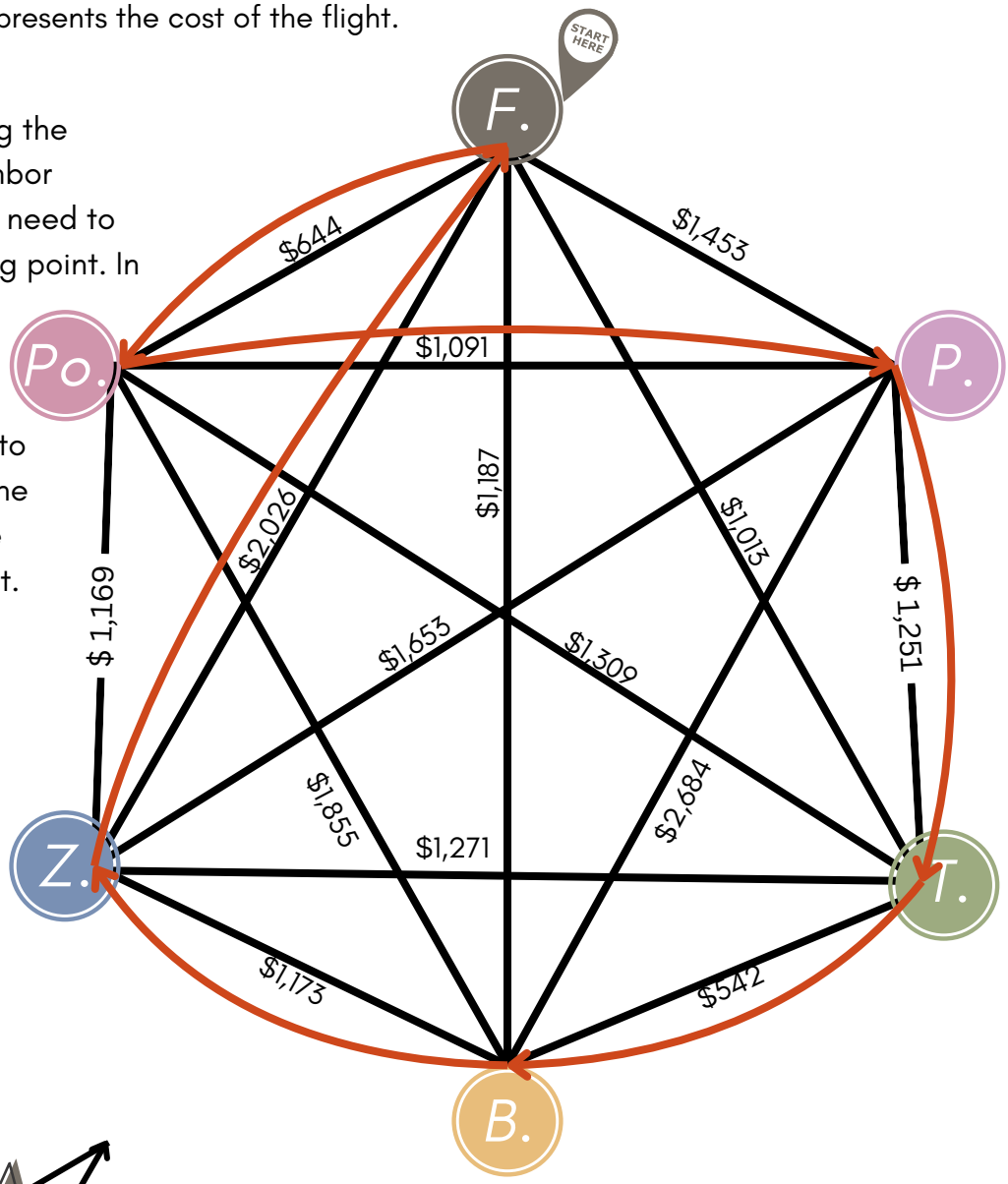
**One of these examples is from Google Flights because Expedia said there were not enough flights coming in and out to give a price.*

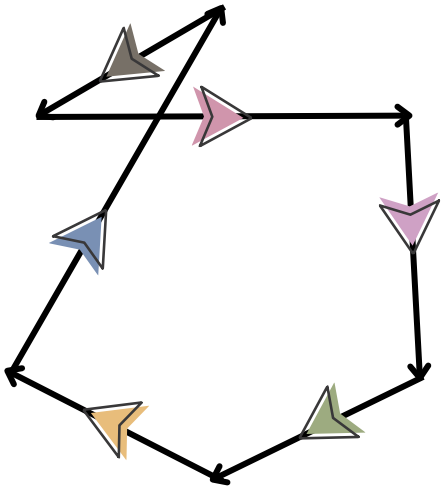
CHEAPEST ROUTES

NEAREST NEIGHBOR ALGORITHM

6 cities means 6 vertices each connected to one another by an edge. The weight provided on the edges represents the cost of the flight.

When applying the Nearest Neighbor algorithm you need to have a starting point. In this case, we start from Fairbanks (F) and proceed to travel along the edge with the lowest weight.



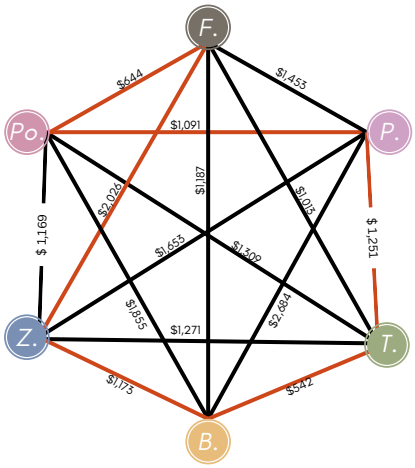


Circuit : F, Po, P, T, B, Z, F

Weight : $ 6,727

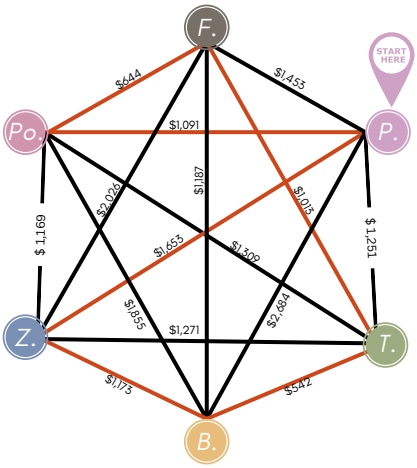
Next we will explore the Repeated Nearest Neighbor algorithm. Repeating the above process across the other 5 vertices in hopes to find a cheaper circuit.

REPEATED NEAREST NEIGHBOR



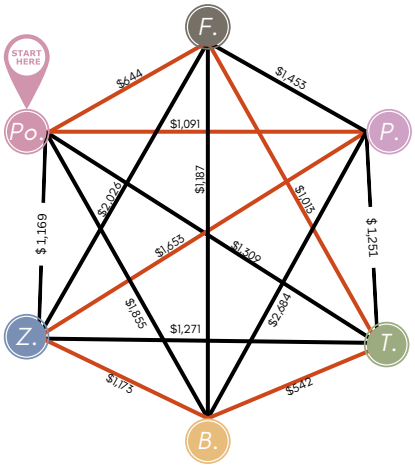
Circuit : F, Po, P, T, B, Z, F

(Our Nearest Neighbor circuit) *Weight :* $ 6,727



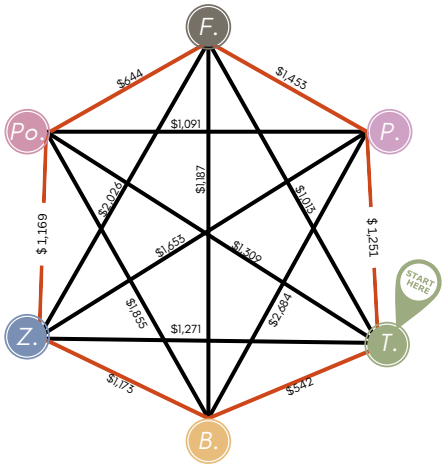
Circuit : P, Po, F, T, B, Z, P

Weight : $6,116



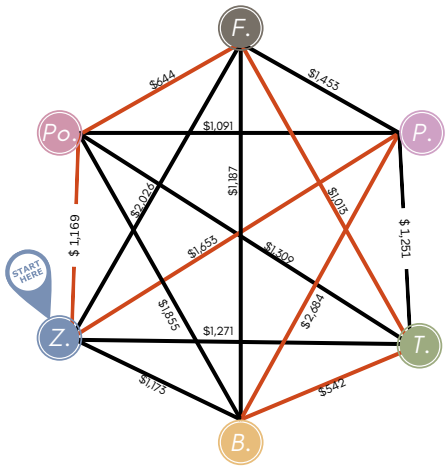
Circuit : Po, F, T, B, Z, P, Po

Weight : $6,116



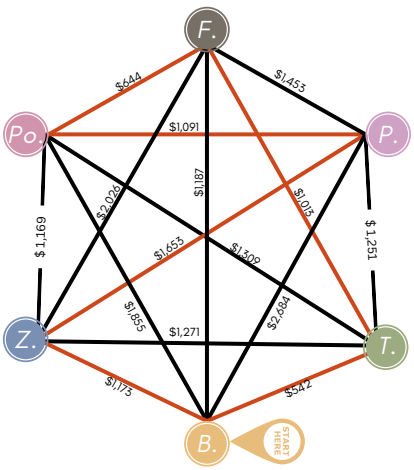
Circuit : T, B, Z, Po, F, P, T

Weight : $6,232



Circuit : Z, Po, F, T, B, P, Z

Weight : $7,705

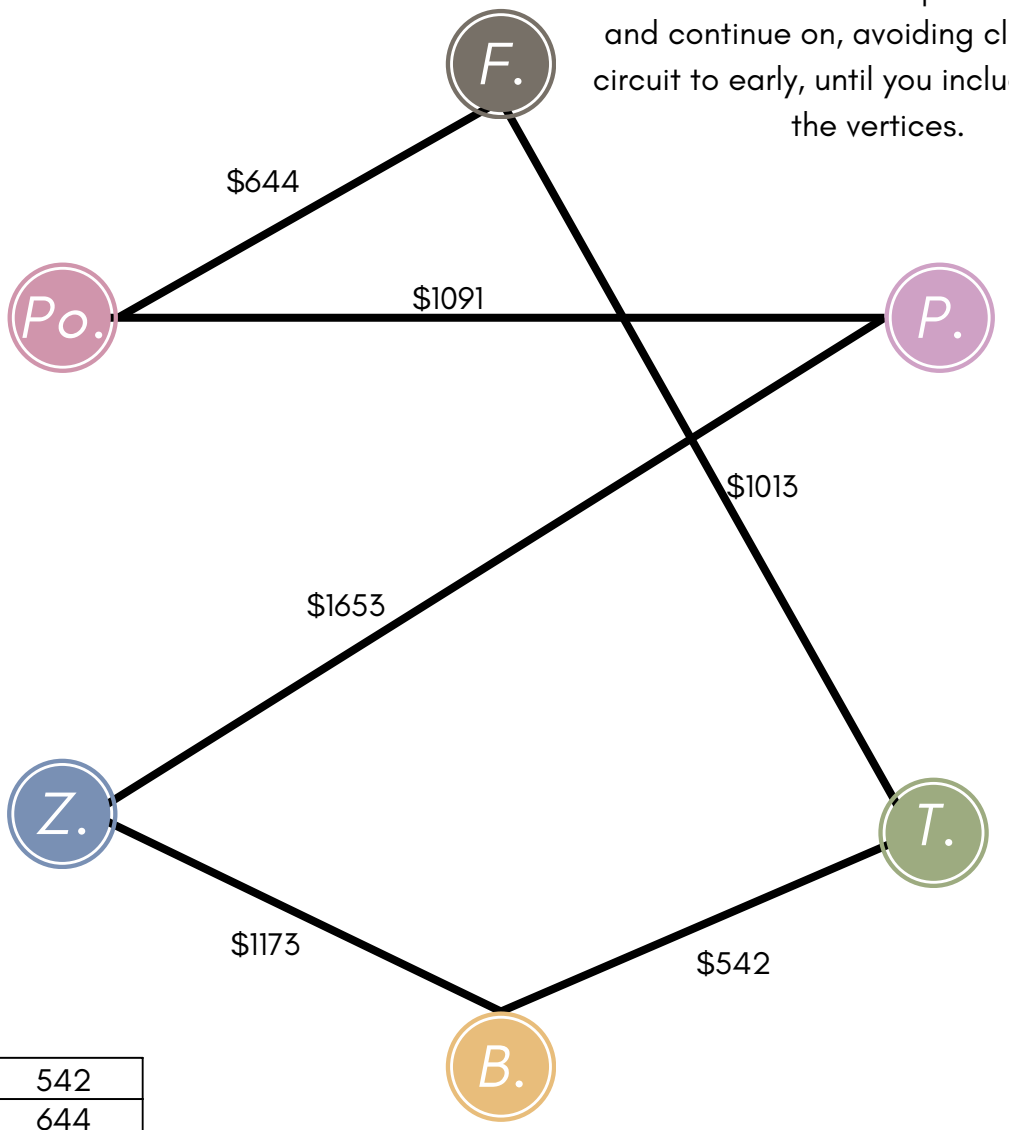


Circuit : B, T, F, Po, P, Z, B

Weight : $6,116

CHEAPEST LINK

The Cheapest Link algorithm involves organizing the edges in order by weight. You will add the cheapest edges first and continue on, avoiding closing your circuit to early, until you include each of the vertices.

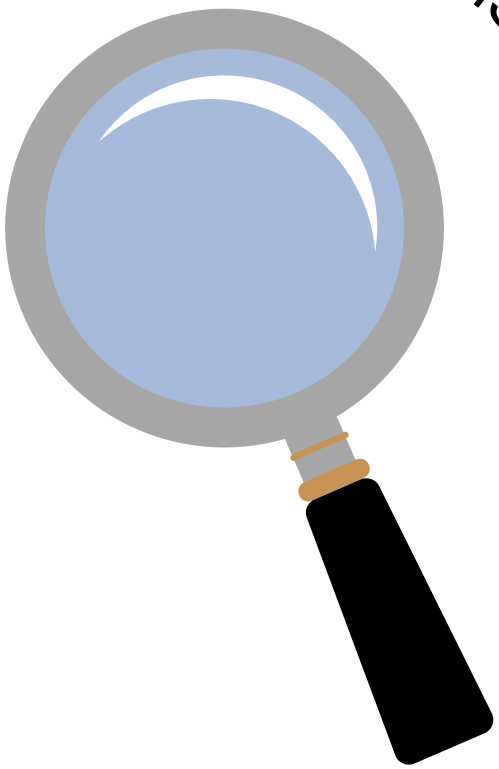


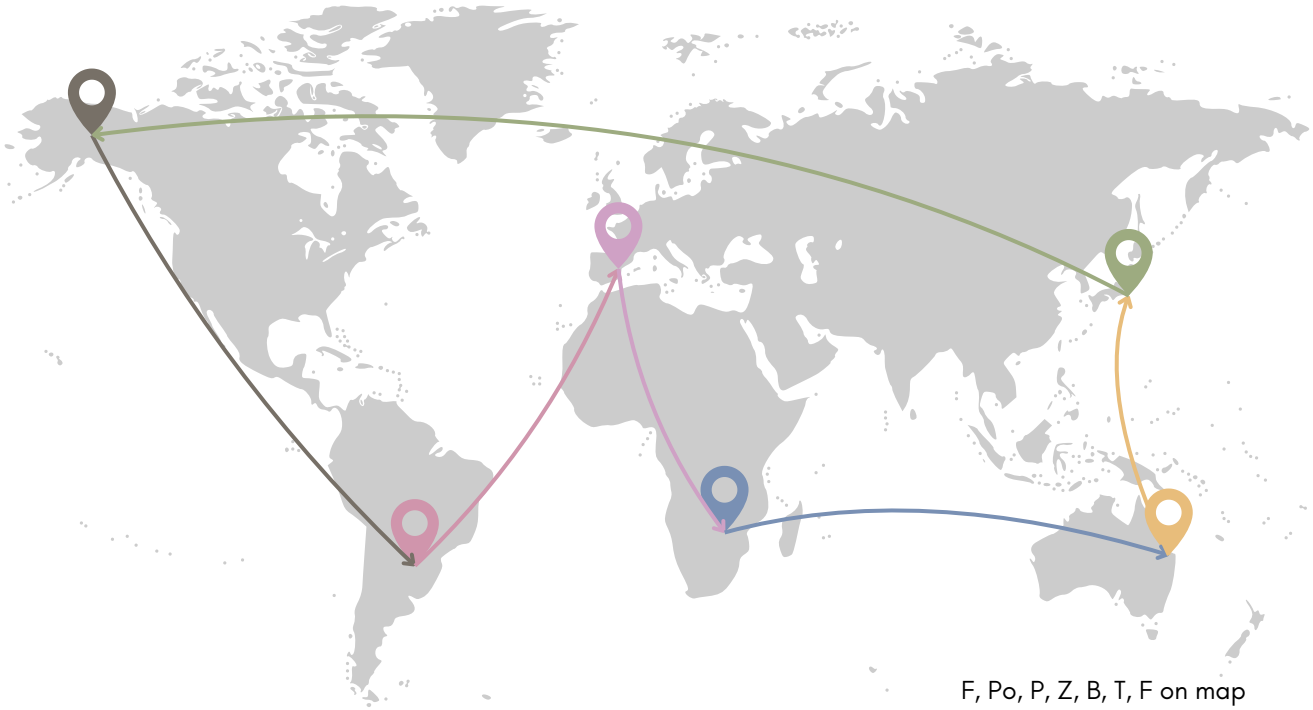
TB	542
FPo	644
FT	1013
PPo	1091
ZPo	1169
ZB	1173
FB	1187
PT	1250
TZ	1271
TPo	1309
FP	1453
PZ	1653
BPo	1855
FZ	2026
PB	2684

Circuit : F, Po, P, Z, B, T, F

Weight : $6,116

FURTHER ANALYSIS





F, Po, P, Z, B, T, F on map

1. *Your turn - Can you find a cheaper Hamiltonian circuit? If not, give a justification for why think you have already found the cheapest circuit.*

I think the cheapest Hamiltonian circuit possible, in this case, is found using the Cheapest Link algorithm. If you add the 15 costs together and divide to find the average cost of a plane ticket it = $1,345.66. Meaning the average cost of this trip would be $8,127.96. Comparing my findings to that, they are all cheaper options. $6,116 is the cost of the cheapest circuit I found and I have been unable to do better on my own. Full disclosure, I would not know how to ask a computer to do this for me and find a cheaper path.

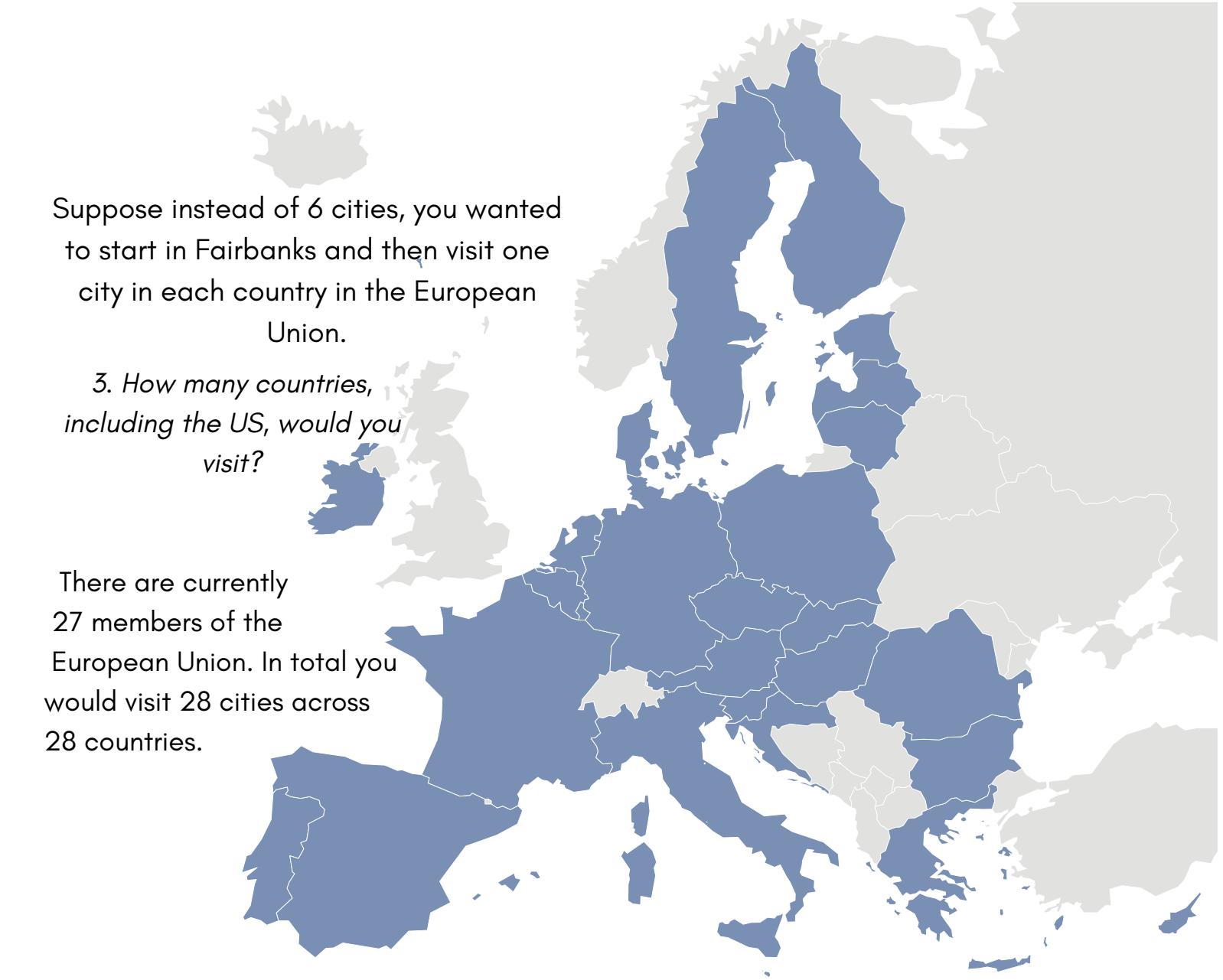
2. Which algorithm gave the cheapest route? Did this surprise you? Write a few sentences explaining your answer. Mention each algorithm in your explanation.

The Cheapest Link algorithm. That itself did not surprise me. With Repeated Nearest Neighbor however, I was surprised to see 3 out of 6 times it gave me the same cost as my Cheapest Link. I feel like I did it correctly and was surprised by the result. It does reaffirm for me that $6,116 is likely the cheapest I could get. Nearest Neighbor is far too restrictive for me, often backing you into a corner where you have to choose a more expensive edge to complete your circuit. NN would cost me $6,727 and I would much rather use my $611 savings on souvenirs.

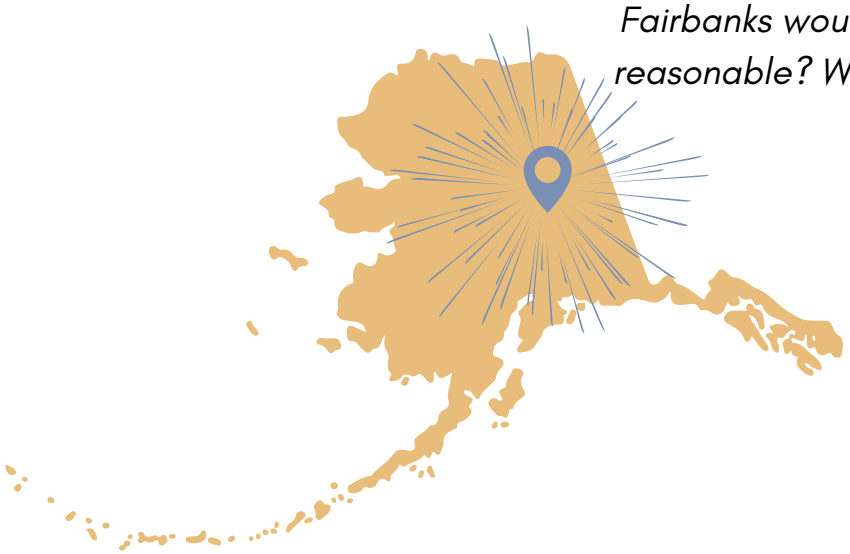
Suppose instead of 6 cities, you wanted to start in Fairbanks and then visit one city in each country in the European Union.

3. How many countries, including the US, would you visit?

There are currently 27 members of the European Union. In total you would visit 28 cities across 28 countries.

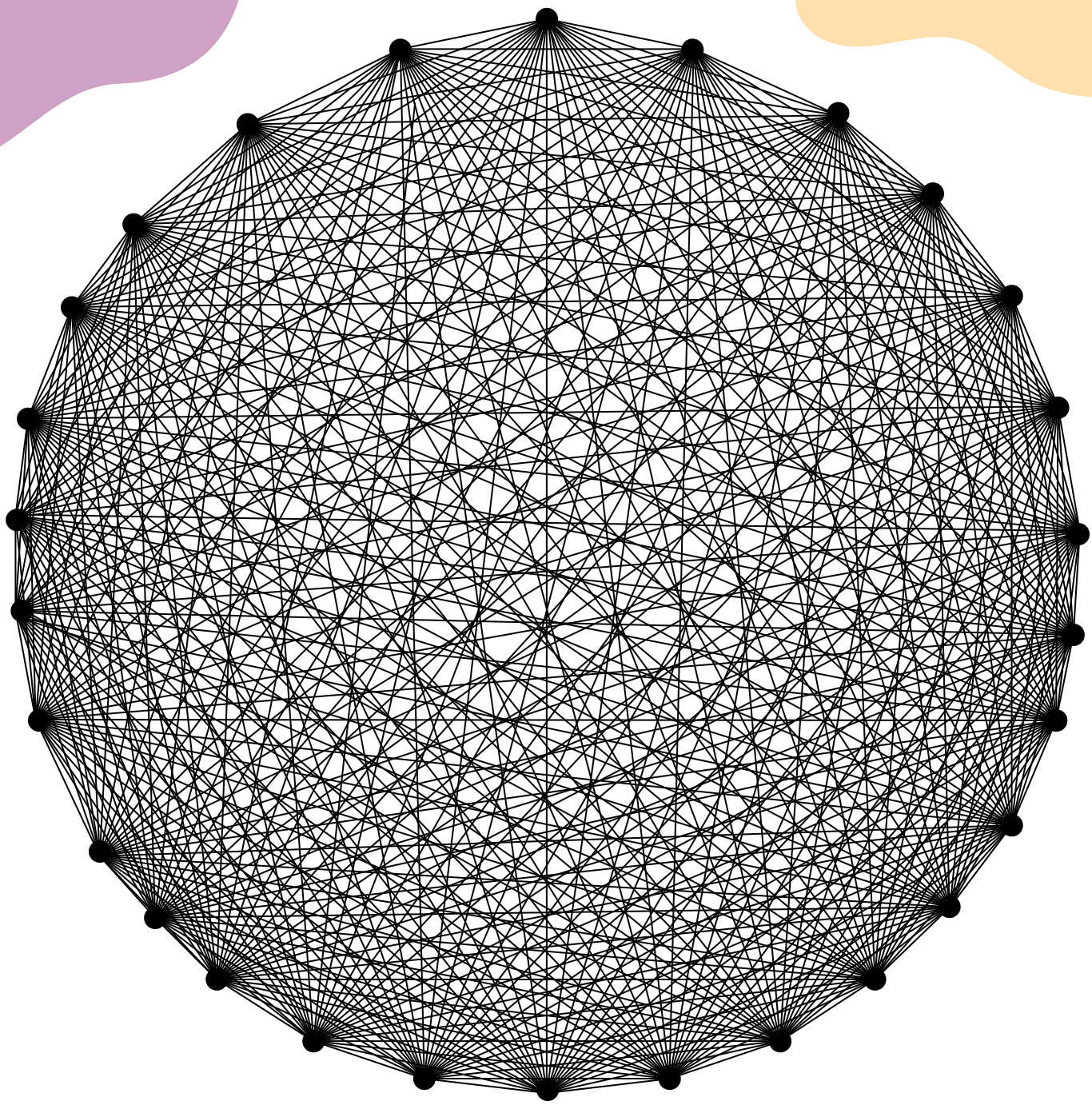


4. If you were to use the Brute Force algorithm to check all the possible Hamiltonian circuits, how many Hamiltonian circuits starting and ending at Fairbanks would you have to check? Would this be reasonable? Write a few sentences explaining your answer.



First, let's think about what that graph may look like. There will be 28 vertices each with 27 edges adjoining them together.
28 x 27 = 756 divided by 2 = 378 edges

But what does that look like?



27! or $1.08888695 \times 10^{28}$

This level of processing power does not exist. To even add weights to this graph and find a path using Cheapest Link would be a very laborious task. It is absolutely unreasonable, I understand wanting to go to Europe but I'm going to guess that if you have the time and money to do that, cutting cost isn't a top priority. My advice would be to look for the cheapest plane into Europe and once you're there simply travel by train.