```
#include <iostream>
#include <array>
#include <vector>
#include <algorithm>

using namespace std;


vector<string> evalHand(vector<int> hand)
{
	sort(hand.begin(), hand.end());

	vector<string> found;

	int straightCount = 0;
	int pairCount = 0;

	int lastCard = hand[0];

	bool foundSomething = false;

	for (int i=1; i<hand.size(); i++)
	{
		if (hand[i] == lastCard)
		{
			pairCount++;
		}
		else
		{
			if (pairCount > 0)
			{
				found.push_back("FOUND " + to_string(pairCount+1) + " " +
to_string(lastCard) + "s");
				foundSomething = true;
				pairCount = 0;
			}
		}

		if (hand[i] == (lastCard + 1))
		{
			straightCount++;
		}
		else
		{
			straightCount = 0;
		}

		lastCard = hand[i];
	}
```

```
	if (pairCount > 0)
	{
		//cout << "FOUND " << (pairCount+1) << " " << lastCard << "s\n";
		found.push_back("FOUND " + to_string(pairCount+1) + " " + to_string(lastCard) +
"s");
		foundSomething = true;
		pairCount = 0;
	}

	if (straightCount == hand.size()-1)
	{
		// cout << "FOUND A STRAIGHT! FROM " << hand[0] << " TO " <<
hand[hand.size()-1] << "\n";
		found.push_back( "FOUND A STRAIGHT! FROM " + to_string(hand[0]) + " TO " +
to_string(hand[hand.size()-1]));
		foundSomething = true;
	}



	if (!foundSomething)
	{
		// cout << "NONE\n";
		found.push_back("NONE");
	}

	return found;
}


void printVec(vector<int> v)
{
	for (int i=0; i<v.size(); i++)
	{
		cout << v[i] << "\n";
	}
}

int main()
{

	// various hands for testing
	vector<int> hand = {1, 2, 3, 2, 5};
	vector<int> hand2 = {1,2,3,4,5,6,7};
```

```
	vector<string> results = evalHand(hand2);

	for (int i=0; i<results.size(); i++)
	{
		cout << results[i] << endl;
	}

	return 0;
}
```