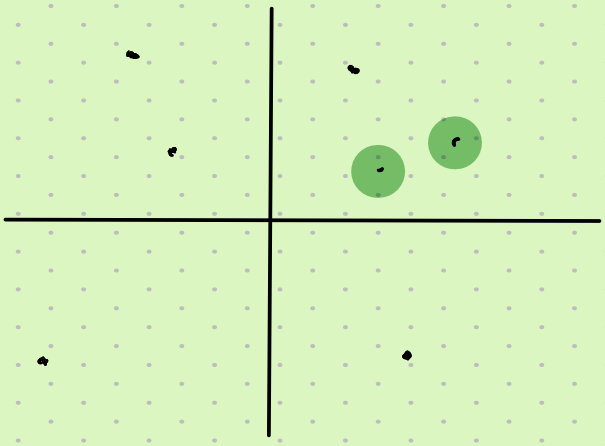
Reminder: here are the tools you have:

- Variables (and datatypes, and assignment)
- Conditionals, if/else... (and boolean expressions...)
  &&, ||, !
  <, >, <=, >=, !=, ==, ...
- loops (while, for, break, continue, booleans)
- functions (value vs reference parameters, return statements)
- vectors + strings (expandable containers of variables)

More practice problems.

Given a collection of points in the plane, find the two distinct points that are closest (Euclidean distance)



How to represent the points?

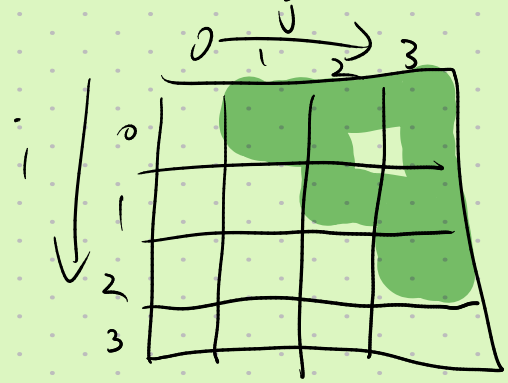
Could use two vectors:

```
vector<int> X, Y;
// Pi = (X[i], Y[i])
```

	X	Y
P_0	3	7
P_1	-1	2
P_2	14	3
⋮		

Given the above representation, details could be as follows:

(say # points = n)



// say d(i,j) gives
// dist. (or dist^2) between
// P_i, P_j

How to go through all pairs:

```
// candidate pair of points?
size_t a=0, b=1;

for (i=0; i<n-1; i++) {
    for (j=i+1; j<n; j++) {
        if (d(i,j) < d(a,b)) {
            a = i;
            b = j;
        }
    }
}
// print answer
cout << ...
```

```
int d(size_t i, size_t j)
{
  int dx = (X[i] - X[j]);
  int dy = (Y[i] - Y[j]);
  return dx*dx + dy*dy;
}
```

Other questions from 2 semesters ago:

Compute difference between sum of all odd integers & even integers.

E.g. if input was 1 9 2 7 4 5,

output should be (16) = (1+9+5+7) − (2+4)

```
int x;
int s = 0;
while(cin >> x) {
    if(x%2) s += x;
    else    s -= x;
}
cout << s;
```

Say all $x \geq 0$. Then this works too:

$$s += \left((x\%2) - (1 - x\%2)\right) * x;$$

$x\%2 == 1 \rightarrow +1$

$x\%2 == 0 \rightarrow -1$

$x\%2 \equiv 1$ if odd, 0 if even

$1 - x\%2 \equiv 1$ if even, 0 if odd