**CS 542 Homework: Relations and functions as sets of ordered pairs**

For this homework, we are working with ordered pairs of unsigned's:

pair<unsigned, unsigned>

and with sets of these pairs:

set<pair<unsigned, unsigned>>

I find these long type names somewhat unwieldy, so I'll use abbreviations:

typedef pair<unsigned, unsigned> OP;

typedef set<OP> SOP;

but you don't have to use these abbreviations.

Here are some little functions to get you used to working with these types:

void show(const OP & op);

show's job is to output the 2 unsigned's in op, with parens and a comma.

For instance, if your main said

OP x(7, 3);

show(x);

or the shorter

show(OP(7,3));

then your show function would output

(7, 3)

OP makeOP(unsigned first, unsigned second);

makeOP's job is to create and return an OP with values equal to the args.

(Not really useful, since main can make its own OP as easily as it can call this function, but this is just for practice.)

OP add(const OP & a, const OP & b);

add's job is to create and return an order pair gotten by adding the corresponding parts of the to args. For example, if your main said

show(add(OP(1,2), OP(5,7)));

then your show function would output

(6, 9)

void show(const SOP & sop);

show's job is to output all the ordered pairs in sop by passing each one to your previous show function.

bool elementOf(const OP & op, const SOP & sop);

elementOf's job is to return whether op is an element of sop.

For example,

elementOf(OP(1,2), SOP{OP(1,1), OP(2,1), OP(2,2))

should return false.

bool firstFound(unsigned first, const SOP & sop);

firstFound's job is to return whether the number in first appears as the first member of any OP in sop.

bool secondFound(unsigned second, const SOP & sop);

secondFound's job is to return whether the number in second appears as the second member of any OP in sop.

bool eitherFound(unsigned n, const SOP & sop);

eitherFound's job is to return whether n appears in either position of any OP in sop.

(It's always fine your functions to call your other functions if you want to.)

SOP Union(const SOP & sop0, const SOP & sop1);

Union's job is to create and return (by value) a set that is the union of the two sets passed to it.