

## CS274, Winter 2001, Test 1

Time 70 min. 5 Questions, 20 pc each. Use extra paper as needed, but make sure to identify each answer.

YOU MUST RETURN THIS PAGE. NAME \_\_\_\_\_

- 1 Implement function `swap()` to swap two integers in the following context:

```
int main(void)
{  int x=5, y=7;
    swap(x,y);
    ...
}
```

- 2 Write a complete program to read names from the keyboard until the user simulates the end of file. Each line in the file is `firstName` followed by a space and `lastName`. You don't have to store them, but print them to the screen, one person per line, with `lastName` followed by comma followed by a space and `firstName`. You may assume no more than 10 chars per name.

For example, if the user types

```
john doe
adam meas
^D
```

the output would be

```
doe, john
meas, adam
```

- 3 Redo #2 above. Now the data comes from file `people.txt` and there is also middle name, and the output is to be produced in `people2.txt`. You may not assume length is predictable and you should use `string` class. Do not use indirection but process the files. For the input file:

```
john adam doe
susan jill herself
```

the output should be

```
john    a.    doe
susan  j.    herself
```

- 4 Write one function that will output an argument message followed by integer arguments. Arguments are a C-string and nonnegative integers. Up to 5 integers are allowed. No negative numbers are allowed. For example, the call

```
myout("They are: ",1,2,4);
```

would print

```
They are: 1 2 4
```

- 5 Assume we have people with names and ages. Create a class `Person` with proper information hiding. Name should be dynamically allocated array of chars. Make sure to allow the following:

```
Person john; // create person with name "No Name" and age -1
Person susan("Susan"); create person with name "Susan" with age 0
Person adam("Adam",27); // create person with name "Adam" and age 27
foo(john); // some function foo gets john on the interface by copy
susan.show(); // will print Name: Susan Age: 0
```

Write separately the header and the implementation files.

You dont have to provide any other access methods except for those required for the above to work.