Password Hashing, Part 2
IN618 Security

Introduction

Answer the questions below to demonstrate your understanding of password hashing and salting. Submit your answers, either **neatly** written or typed and printed in class next week.

1 Unsalted hashes

Consider the username/password table below:

```
bob:CDF2FD4730D3366AF0951DD1DD1F25444FA2CD007053844D5829D91893F7E20B
sarah:E201065D0554652615C320C00A1D5BC8EDCA469D72C2790E24152D0C1E2B6189
greg:4FE6CEC6EEB3D73B5166D38BC643CD1897C9BF3F76D668DB694AAAEBB29CF18E
tony:1CF272DE115F5B3A800E3A1EAB4D2AACAF85492950162A3D29C8DE76FBE05EE1
lisa:A61BCE1033160B76C30090B9B1D3B9868C85F0506F705D944F1574B47FC59F04
morgan:1B8E7510541393A3D960B8B174E7D8D645BBB0950FE8BA5AC276A04AC56849F5
jeff:E201065D0554652615C320C00A1D5BC8EDCA469D72C2790E24152D0C1E2B6189
oscar:9CB79C8C62ABD2E26D7E8A07FF7647ED54052F5CFE82659B266F8764B37626FC
lena:5EBF23D1FD1911558CD95287362EED404D5DC2980AABCD98FC2D92B7BD4D5531
robert:EDAE941B81B5385E559C537C5EA9EF92A9DB4427C3A9CFE8658CE9A27AE95064
scott:ACDE941CC29B7385E559C537C5EA9EF92A9DB4427C3A9CFE8658CE9A2745EA7A
jenn:ADB42C99D5FD68AFC63DA94AE893CADACF045CC6794864768AB068551275B78D
```

1. Two of these users have the same password. Can you tell which two? If so, how can you tell? If not, why not?
2. Suppose that all of these passwords are eight characters long. Write pseudocode for a program that will find all of the username/password pairs.

2 Salted hashes

Now consider this table, which has the form `username:salt:password hash`.

```
bob:gW7jskzypyw=:0E87647B2AA7CFD759ABB24260D03DD2D489707531B3ABA9968C011EABC5945A
sarah:szpfjhADFBw=:C77BE15E57A670744263FF9095B542BEDBA353C2A6025B1E7F8B265F7369B315
greg:N7Np0O9L7o=:CB4BE07B7B5A2EA6B56ABC7857166008261987A153F97B8AC8F7A606B8A42A25
tony:UZ7PD2oWy7Q=:2783B1BA9F2799F4CC40CC5DFB7A33E4112366097F3D8F61AD40ACCD65721E13
lisa:qKki+EVwB0o=:AF4324A5812C80CE8F409328236727044A841563A35FEB65C6AD24A3A7D0766F
morgan:8O2TrK6lUU8=:6DB7D53F6632292FB8595086EF1D1A4D3E08C3F35A3CEC946A9B0A4FE51479BA
jeff:uH47I2SWp5c=:C30D0D9EB33473E48794A255D1351CE54F7294D1CEB94B87CC9DA52A1DE12925
oscar:ESAAu9k4Q7o=:0873D5437B27EFCB139F96A91BA8C06FBFE2D576159CB986096BC9405C87F359
lena:8BVIgID9AA4=:678576E7BD14A53C45D0F329DEC525F8A67FFE48EFB963CF8EA921586D4A6FF8
robert:NQmR1BiGgCQ=:0352496FF14FFD638EA074EA612E8B9A88F9A9DE8E0F977963068836563A0ADC
scott:K3502f81NPs=:4C66FA90247772AC540EAA4B6A8BA6C1D44D4A8377A67A292ACA11C66E85F1DB
jenn:jl86xL8HnDA=:F7FDA25C8B2600D0A3A6B2AAC2C8EBDC40B4F2CB42579DC8A7EB1A09CA87EC2F
```

1. Again, two users have the same password (not the same two) *Just by looking at the table*, can you tell which two? If so, how can you tell? If not, why not?
2. Will the program that you pseudocoded above work on this table with only minimal changes? Describe briefly the changes that would be required to make your program work on this table.