**data** growth;

input salinity $ app @;

do rep = **1** to **3**;

input growth @;

output;

end;

cards;

c 1 11.29 11.08 11.1

c 2 7.37 6.55 8.5

6b 1 5.64 5.98 5.69

6b 2 4.2 3.34 4.21

12b 1 4.83 4.77 5.66

12b 2 3.28 2.61 2.69

;

**run**;

/\*

Answer to 2

The factors are application methods(app) and salinity treatments

both the factors were fixed as all the possible levels included in the study is present in the data

both the factors were nested as App(salinity) si possible but not Salinity(App); salinity is not constant across applications

App(salinity) implies, for each level of salinity, there is an app treatment

\*/

/\*

Answer to 3

y\_(i)jk = mu + alpha\_i + beta\_(i)j +epsilon\_(i)jk , and epsilon\_(I)jk ~iid N(0, sigma^2)

mu = true mean growth for all observed treatment contributions (as it is fixed)

alpha\_i = true effect of the ith sainity treatment on the mean growth

beta\_(i)j = true effect of the jth application in the ith salinity treatment on the mean growth

epsilon\_(i)jk = resildual or error for the random assignment of treatment (i)j to Barley specimen k

sigma^2 = true varaince of the residuals

\*/

/\*

Answer to 4

two factor, fixed effects, nested model or design

\*/

/\*

Answer to 5

Salinity does have a signifcant effect as seen from Type 1 error table. p-valeu is really small

Yes. We need to investiagte the LSMEANS of salinity and thsi will give three constrats for c vs 12b,

c vs. cb and 12b vs. 6b [main effects]

\*/

/\*

Answer to 6

Yes; as seen from Type 1 error table the p-value is really small

[simple effects as app is nsted within slainity ]

\*/

\*Code Block #1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **glm** data = growth;

class salinity app;

model growth = salinity app(salinity);

**run**;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*Code Block #2\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **glm** data = growth;

class salinity app;

model growth = salinity app(salinity) / clparm alpha = **0.00833**; \* 6 simultaneous CIs so we will do Bonferroni alpha =0.05/6 .... as 6 simultaneious CIs ;

\*for each theta we need coeff on\*\*\*\*\*mu\*\*\*\*\*\*\*\*alpha1\*\*alpha2\*\*alpha3\*\*\*\*\*\*\*\*\*\*\*beta11\*\*beta12\*\*beta21\*\*beta22\*\*beta31\*\*beta32\*\*\* use a divisor, if needed;

\* | | | | | | | | | | | ;

\* | | | | | | | | | | | ;

estimate '12b vs 6b' intercept **0** salinity **2** -**2** **0** app(salinity) **1** **1** -**1** -**1** **0** **0** / divisor = **2** e; /\*follow SAS order 1:12b , 2:6b, c:6b\*/

estimate '12b vs. c' intercept **0** salinity **2** -**2** **0** app(salinity) **1** / divisor = **2** e; \*incompleet ;

estimate;

estimate;

estimate;

estimate;

**run**;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*

breaking up the contrasts

mu\_12b-mu\_c = mu\_1-mu\_3 = ((mu\_(1)1 + mu\_(1)2)/2) - ((mu\_(3)1 + mu\_(3)2)/2) = ((mu + alpha\_1 +beta\_(1)1) + (mu + alpha\_1 +beta\_(1)2))/2 - ((mu + alpha\_3 +beta\_(3)1) + (mu + alpha\_3 +beta\_(3)2))/2

\*/