

```

# 3)
#Run renumf90
echo renum.par | renumf90 | tee renum.log

#Run blupf90
echo renf90.par | blupf90 | tee blup.log

#Print solutions and accuracy using new ID
awk '{print $1,$10}' renadd02.ped | sort +0 -1 > p.temp
awk '$2==2{print $3,$4,(1-($5*$5/20)),(1-($5*$5/20))**0.5,$5}' solutions | sort +0 -1 > s.temp
join -1 +1 -2 +1 p.temp s.temp | sort +1 -2 > sol.new

```

```

# 6) Run remlf90 and airemlf90 using exmr99s1
time echo exmr99s1 | remlf90 | tee reml.log
time echo exmr99s1 | airemlf90 | tee aireml.log

```

```

time echo exmr99s2 | remlf90 | tee reml2.log
time echo exmr99s2 | airemlf90 | tee aireml2.log

```

```

cp exmr99s2 renf90.par
echo 'OPTION se_covar_function h2_t1 G_3_3_1_1/(G_2_2_1_1+G_3_3_1_1+R_1_1)
OPTION se_covar_function h2_t2 G_3_3_2_2/(G_2_2_2_2+G_3_3_2_2+R_2_2)
OPTION se_covar_function rg12 G_3_3_1_2/(G_3_3_1_1*G_3_3_2_2)**0.5' >> renf90.par

```

```

time echo renf90.par | airemlf90 | tee aireml3.log

```

```

#7) Run gibbs2f90
gibbs2f90 exmr99s1 --rounds 1000 --burnin 0 --thin 1 | tee gibbs2.log
#or
printf "%s\n" "exmr99s1" "1000 0" "1" | gibbs2f90 > gibbs2.log

```

```

printf "%s\n" "exmr99s1" "0" "1" "1" "1 2 3" | postgibbsf90
#or
postgibbsf90 exmr99s1
0
1
1
1 2 3
0

```

```

printf "%s\n" "exmr99s1" "200" "1" "1" "1 2 3" | postgibbsf90
#or

```

```
postgibbsf90 exmr99s1
```

```
200
```

```
1
```

```
1
```

```
1 2 3
```

```
0
```

```
#G matrix for effect =      2
```

```
# 891.64
```

```
# G matrix for effect =      3
```

```
# 2578.6
```

```
# R matrix
```

```
# 1816.7
```

```
#8) Run gibbs2f90 with fixed variance components for EBV
```

```
cp exmr99s1 gibbs.par
```

```
sed -i '17s/1806.4187/1816.7/' gibbs.par
```

```
sed -i '25s/1091.6244/891.64/' gibbs.par
```

```
sed -i '33s/2403.6412/2578.6/' gibbs.par
```

```
echo 'OPTION fixed_var mean 3' >> gibbs.par
```

```
gibbs2f90 gibbs.par --rounds 1000 --burnin 200 --thin 1 | tee gibbs2_ebv.log
```

```
#9) Run RRM
```

```
echo renrr.par | renumf90 | tee renum.log
```

```
echo renf90.par | blupf90 | tee blup.log
```