



**Öğrencinin;**

**ADI: Selin**

**SOYADI: Daldaban**

**NO: 1421221014**

**BÖLÜM: Bilgisayar Mühendisliği**

**Projenin;**

**KONUSU: Linux Shell Yazımı**

**Dersin;**

**ADI: İşletim Sistemleri**

**EĞİTMEN:**

**Prof. Dr. A. Yılmaz ÇAMURCU**

**Ar. Gör. Samet KAYA**

**Ar. Gör. Okan KARA**

## **İçindekiler**

<b>1. Özet</b>	<b>3</b>
<b>2. Proje Konusu</b>	<b>3</b>
<b>3. Proje Çıktıları ve Başarı Ölçütleri</b>	<b>3</b>
<b>4. Proje Süresince Yapılanlar</b>	<b>5</b>
<b>5. Kaynakça</b>	<b>5</b>

# 1 Özet

## Pipe nedir?

Temel olarak process ler arası iletişim için kullanılır. Bir pipe tek yönlü iletişime izin verir. Veri pipe a yazıldığında aynı sıra ile veriler okunur. Tipik olarak, bir pipe ana proses ile çocuk proses arasında yada tek bir proses içerisinde 2 thread arasındaki iletişimi kurmak için kullanılır.[1]

Pipe oluşturmak için pipe komutunu çağırmalıyız. Pipe 2 boyutunda bir dizi sağlar. Bu dizideki 0 okumayı 1 yazmayı temsil eder. Alt programlardan ise 3 oku, 4 ana programa geri yaz anlamına gelir. İki belge arasında **write** ve **read** komutları ile boru hattı oluşturarak haberleşme sağlanır.

**Write:** Programlar arası veri yazmayı sağlar.

**Read:** Programlar arası veri okumayı sağlar.

## 2 Proje Konusu

Projede determinant3x3, kofaktor2x2 ve satsutsec adlı 3 c dosyasından pipe kullanarak girilen değerlerden determinantının hesaplanması istenmektedir. Satsutsec ile random sat1, sat2, sat3, sut1, sut2 veya sut3 determinanta pipe ile gönderilecek. Gelen değere göre 2x2 matrisler oluşturularak kofaktor hesaplama gönderilecek. 3 kere okuma yazma yaparak değerlerin toplamı bize determinantı verecektir. Kullanılan fonksiyonlar: `execv()`, `wait()`, `fork()`, `write()`, `read()` ile işlemler gerçekleştirilmiştir.

## 3 Proje Çıktıları ve Başarı Ölçütleri

Projede, veri akışları istenildiği gibi oluşturulup doğru çalışmaktadır. Aşağıda gözüken sut 2 sonuç kısmı sut2 için 3 tane oluşturduğum diziyi göstermektedir. En altta toplamdaki determinant sonucunu göstermektedir.

```
4
7
8
9 determinant3x3.c x kofakhesap2x2.c x satsutsec.c x
51
42 #include <stdio.h>
53 #include <string.h>
4 #include <stdlib.h>
matris #include <math.h>
54
55 int main()
56 {
57     random deger : sat3
58     sat3 sonuc: 8 // determinanttan gelen diziler icin yer ayirdim
59     sat3 sonuc: 9
60     sat3 sonuc: 5
61     sat 3 icin determinant sonuc : 22
62     : selin@selin-X555UB:~/Masaüstü/Selin_Daldaban_1421221014_opsis_pro2$ ./determinant3x3
63     9 eleman giriniz
64     4
65     7 int sonuc = 1;
66     8 read(3, &x, sizeof(int)*4); // determinanttan okudum
67     9 read(3, &y, sizeof(int)*4);
68     10 read(3, &z, sizeof(int)*4);
69     11 read(3, &xkonum, sizeof(int)); // us degerlerini okudum
70     12 read(3, &ykonum, sizeof(int));
71     13 read(3, &zkonum, sizeof(int));
72     14 sonucx = (x[0][0] * x[1][1]) - (x[1][0] * x[0][1]); //kofaktor icin hesaplan
73     15 sonucy = (y[0][0] * y[2][2]) - (y[1][0] * y[0][1]);
74     16 sonucz = (z[0][0] * z[2][2]) - (z[1][0] * z[0][1]);
75     17 // printf("x sonuc : %d \n ",sonucx); printf("y sonuc : %d \n ",sonucy); pr
76     18 for (i=0; i<xkonum; i++){sonuc =sonuc*(-1);} sonucx=sonucx*sonuc; //us degerl
77     19 for (i=0; i<ykonum; i++){sonuc =sonuc*(-1);} sonucy=sonucy*sonuc;
78     20 for (i=0; i<zkonum; i++){sonuc =sonuc*(-1);} sonucz=sonucz*sonuc;
79     21 random deger : sut2
80     22 sut2 sonuc: 1
81     23 printf("x sonuc : %d \n ",sonucx); printf("y sonuc : %d \n ",sonucy); pr
82     24 write(4, &sonucx, sizeof(int)); //determinanta sonuclari yolladim
83     25 write(4, &sonucy, sizeof(int));
84     26 sut2 icin determinant sonuc : 10
85     27
86     28
87     29
88     30
89     31
90     31
91     32
92     33
93     34
94     35
95     36
96     37
97     38
98     39
99     40
100     41
101     42
102     43
103     44
104     45
105     46
106     47
107     48
108     49
109     50
110     51
111     52
112     53
113     54
114     55
115     56
116     57
117     58
118     59
119     60
120     61
121     62
122     63
123     64
124     65
125     66
126     67
127     68
128     69
129     70
130     71
131     72
132     73
133     74
134     75
135     76
136     77
137     78
138     79
139     80
140     81
141     82
142     83
143     84
144     85
145     86
146     87
147     88
148     89
149     90
150     91
151     92
152     93
153     94
154     95
155     96
156     97
157     98
158     99
159     100
160     101
161     102
162     103
163     104
164     105
165     106
166     107
167     108
168     109
169     110
170     111
171     112
172     113
173     114
174     115
175     116
176     117
177     118
178     119
179     120
180     121
181     122
182     123
183     124
184     125
185     126
186     127
187     128
188     129
189     130
190     131
191     132
192     133
193     134
194     135
195     136
196     137
197     138
198     139
199     140
200     141
201     142
202     143
203     144
204     145
205     146
206     147
207     148
208     149
209     150
210     151
211     152
212     153
213     154
214     155
215     156
216     157
217     158
218     159
219     160
220     161
221     162
222     163
223     164
224     165
225     166
226     167
227     168
228     169
229     170
230     171
231     172
232     173
233     174
234     175
235     176
236     177
237     178
238     179
239     180
240     181
241     182
242     183
243     184
244     185
245     186
246     187
247     188
248     189
249     190
250     191
251     192
252     193
253     194
254     195
255     196
256     197
257     198
258     199
259     200
260     201
261     202
262     203
263     204
264     205
265     206
266     207
267     208
268     209
269     210
270     211
271     212
272     213
273     214
274     215
275     216
276     217
277     218
278     219
279     220
280     221
281     222
282     223
283     224
284     225
285     226
286     227
287     228
288     229
289     230
290     231
291     232
292     233
293     234
294     235
295     236
296     237
297     238
298     239
299     240
300     241
301     242
302     243
303     244
304     245
305     246
306     247
307     248
308     249
309     250
310     251
311     252
312     253
313     254
314     255
315     256
316     257
317     258
318     259
319     260
320     261
321     262
322     263
323     264
324     265
325     266
326     267
327     268
328     269
329     270
330     271
331     272
332     273
333     274
334     275
335     276
336     277
337     278
338     279
339     280
340     281
341     282
342     283
343     284
344     285
345     286
346     287
347     288
348     289
349     290
350     291
351     292
352     293
353     294
354     295
355     296
356     297
357     298
358     299
359     300
360     301
361     302
362     303
363     304
364     305
365     306
366     307
367     308
368     309
369     310
370     311
371     312
372     313
373     314
374     315
375     316
376     317
377     318
378     319
379     320
380     321
381     322
382     323
383     324
384     325
385     326
386     327
387     328
388     329
389     330
390     331
391     332
392     333
393     334
394     335
395     336
396     337
397     338
398     339
399     340
400     341
401     342
402     343
403     344
404     345
405     346
406     347
407     348
408     349
409     350
410     351
411     352
412     353
413     354
414     355
415     356
416     357
417     358
418     359
419     360
420     361
421     362
422     363
423     364
424     365
425     366
426     367
427     368
428     369
429     370
430     371
431     372
432     373
433     374
434     375
435     376
436     377
437     378
438     379
439     380
440     381
441     382
442     383
443     384
444     385
445     386
446     387
447     388
448     389
449     390
450     391
451     392
452     393
453     394
454     395
455     396
456     397
457     398
458     399
459     400
460     401
461     402
462     403
463     404
464     405
465     406
466     407
467     408
468     409
469     410
470     411
471     412
472     413
473     414
474     415
475     416
476     417
477     418
478     419
479     420
480     421
481     422
482     423
483     424
484     425
485     426
486     427
487     428
488     429
489     430
490     431
491     432
492     433
493     434
494     435
495     436
496     437
497     438
498     439
499     440
500     441
501     442
502     443
503     444
504     445
505     446
506     447
507     448
508     449
509     450
510     451
511     452
512     453
513     454
514     455
515     456
516     457
517     458
518     459
519     460
520     461
521     462
522     463
523     464
524     465
525     466
526     467
527     468
528     469
529     470
530     471
531     472
532     473
533     474
534     475
535     476
536     477
537     478
538     479
539     480
540     481
541     482
542     483
543     484
544     485
545     486
546     487
547     488
548     489
549     490
550     491
551     492
552     493
553     494
554     495
555     496
556     497
557     498
558     499
559     500
560     501
561     502
562     503
563     504
564     505
565     506
566     507
567     508
568     509
569     510
570     511
571     512
572     513
573     514
574     515
575     516
576     517
577     518
578     519
579     520
580     521
581     522
582     523
583     524
584     525
585     526
586     527
587     528
588     529
589     530
590     531
591     532
592     533
593     534
594     535
595     536
596     537
597     538
598     539
599     540
600     541
601     542
602     543
603     544
604     545
605     546
606     547
607     548
608     549
609     550
610     551
611     552
612     553
613     554
614     555
615     556
616     557
617     558
618     559
619     560
620     561
621     562
622     563
623     564
624     565
625     566
626     567
627     568
628     569
629     570
630     571
631     572
632     573
633     574
634     575
635     576
636     577
637     578
638     579
639     580
640     581
641     582
642     583
643     584
644     585
645     586
646     587
647     588
648     589
649     590
650     591
651     592
652     593
653     594
654     595
655     596
656     597
657     598
658     599
659     600
660     601
661     602
662     603
663     604
664     605
665     606
666     607
667     608
668     609
669     610
670     611
671     612
672     613
673     614
674     615
675     616
676     617
677     618
678     619
679     620
680     621
681     622
682     623
683     624
684     625
685     626
686     627
687     628
688     629
689     630
690     631
691     632
692     633
693     634
694     635
695     636
696     637
697     638
698     639
699     640
700     641
701     642
702     643
703     644
704     645
705     646
706     647
707     648
708     649
709     650
710     651
711     652
712     653
713     654
714     655
715     656
716     657
717     658
718     659
719     660
720     661
721     662
722     663
723     664
724     665
725     666
726     667
727     668
728     669
729     670
730     671
731     672
732     673
733     674
734     675
735     676
736     677
737     678
738     679
739     680
740     681
741     682
742     683
743     684
744     685
745     686
746     687
747     688
748     689
749     690
750     691
751     692
752     693
753     694
754     695
755     696
756     697
757     698
758     699
759     700
760     701
761     702
762     703
763     704
764     705
765     706
766     707
767     708
768     709
769     710
770     711
771     712
772     713
773     714
774     715
775     716
776     717
777     718
778     719
779     720
780     721
781     722
782     723
783     724
784     725
785     726
786     727
787     728
788     729
789     730
790     731
791     732
792     733
793     734
794     735
795     736
796     737
797     738
798     739
799     740
800     741
801     742
802     743
803     744
804     745
805     746
806     747
807     748
808     749
809     750
810     751
811     752
812     753
813     754
814     755
815     756
816     757
817     758
818     759
819     760
820     761
821     762
822     763
823     764
824     765
825     766
826     767
827     768
828     769
829     770
830     771
831     772
832     773
833     774
834     775
835     776
836     777
837     778
838     779
839     780
840     781
841     782
842     783
843     784
844     785
845     786
846     787
847     788
848     789
849     790
850     791
851     792
852     793
853     794
854     795
855     796
856     797
857     798
858     799
859     800
860     801
861     802
862     803
863     804
864     805
865     806
866     807
867     808
868     809
869     810
870     811
871     812
872     813
873     814
874     815
875     816
876     817
877     818
878     819
879     820
880     821
881     822
882     823
883     824
884     825
885     826
886     827
887     828
888     829
889     830
890     831
891     832
892     833
893     834
894     835
895     836
896     837
897     838
898     839
899     840
900     841
901     842
902     843
903     844
904     845
905     846
906     847
907     848
908     849
909     850
910     851
911     852
912     853
913     854
914     855
915     856
916     857
917     858
918     859
919     860
920     861
921     862
922     863
923     864
924     865
925     866
926     867
927     868
928     869
929     870
930     871
931     872
932     873
933     874
934     875
935     876
936     877
937     878
938     879
939     880
940     881
941     882
942     883
943     884
944     885
945     886
946     887
947     888
948     889
949     890
950     891
951     892
952     893
953     894
954     895
955     896
956     897
957     898
958     899
959     900
960     901
961     902
962     903
963     904
964     905
965     906
966     907
967     908
968     909
969     910
970     911
971     912
972     913
973     914
974     915
975     916
976     917
977     918
978     919
979     920
980     921
981     922
982     923
983     924
984     925
985     926
986     927
987     928
988     929
989     930
990     931
991     932
992     933
993     934
994     935
995     936
996     937
997     938
998     939
999     940
1000    941
1001    942
1002    943
1003    944
1004    945
1005    946
1006    947
1007    948
1008    949
1009    950
1010    951
1011    952
1012    953
1013    954
1014    955
1015    956
1016    957
1017    958
1018    959
1019    960
1020    961
1021    962
1022    963
1023    964
1024    965
1025    966
1026    967
1027    968
1028    969
1029    970
1030    971
1031    972
1032    973
1033    974
1034    975
1035    976
1036    977
1037    978
1038    979
1039    980
1040    981
1041    982
1042    983
1043    984
1044    985
1045    986
1046    987
1047    988
1048    989
1049    990
1050    991
1051    992
1052    993
1053    994
1054    995
1055    996
1056    997
1057    998
1058    999
1059    1000
1060    1001
1061    1002
1062    1003
1063    1004
1064    1005
1065    1006
1066    1007
1067    1008
1068    1009
1069    1010
1070    1011
1071    1012
1072    1013
1073    1014
1074    1015
1075    1016
1076    1017
1077    1018
1078    1019
1079    1020
1080    1021
1081    1022
1082    1023
1083    1024
1084    1025
1085    1026
1086    1027
1087    1028
1088    1029
1089    1030
1090    1031
1091    1032
1092    1033
1093    1034
1094    1035
1095    1036
1096    1037
1097    1038
1098    1039
1099    1040
1100    1041
1101    1042
1102    1043
1103    1044
1104    1045
1105    1046
1106    1047
1107    1048
1108    1049
1109    1050
1110    1051
1111    1052
1112    1053
1113    1054
1114    1055
1115    1056
1116    1057
1117    1058
1118    1059
1119    1060
1120    1061
1121    1062
1122    1063
1123    1064
1124    1065
1125    1066
1126    1067
1127    1068
1128    1069
1129    1070
1130    1071
1131    1072
1132    1073
1133    1074
1134    1075
1135    1076
1136    1077
1137    1078
1138    1079
1139    1080
1140    1081
1141    1082
1142    1083
1143    1084
1144    1085
1145    1086
1146    1087
1147    1088
1148    1089
1149    1090
1150    1091
1151    1092
1152    1093
1153    1094
1154    1095
1155    1096
1156    1097
1157    1098
1158    1099
1159    1100
1160    1101
1161    1102
1162    1103
1163    1104
1164    1105
1165    1106
1166    1107
1167    1108
1168    1109
1169    1110
1170    1111
1171    1112
1172    1113
1173    1114
1174    1115
1175    1116
1176    1117
1177    1118
1178    1119
1179    1120
1180    1121
1181    1122
1182    1123
1183    1124
1184    1125
1185    1126
1186    1127
1187    1128
1188    1129
1189    1130
1190    1131
1191    1132
1192    1133
1193    1134
1194    1135
1195    1136
1196    1137
1197    1138
1198    1139
1199    1140
1200    1141
1201    1142
1202    1143
1203    1144
1204    1145
1205    1146
1206    1147
1207    1148
1208    1149
1209    1150
1210    1151
1211    1152
1212    1153
1213    1154
1214    1155
1215    1156
1216    1157
1217    1158
1218    1159
1219    1160
1220    1161
1221    1162
1222    1163
1223    1164
1224    1165
1225    1166
1226    1167
1227    1168
1228    1169
1229    1170
1230    1171
1231    1172
1232    1173
1233    1174
1234    1175
1235    1176
1236    1177
1237    1178
1238    1179
1239    1180
1240    1181
1241    1182
1242    1183
1243    1184
1244    1185
1245    1186
1246    1187
1247    1188
1248    1189
1249    1190
1250    1191
1251    1192
1252    1193
1253    1194
1254    1195
1255    1196
1256    1197
1257    1198
1258    1199
1259    1200
1260    1201
1261    1202
1262    1203
1263    1204
1264    1205
1265    1206
1266    1207
1267    1208
1268    1209
1269    1210
1270    1211
1271    1212
1272    1213
1273    1214
1274    1215
1275    1216
1276    1217
1277    1218
1278    1219
1279    1220
1280    1221
1281    1222
1282    1223
1283    1224
1284    1225
1285    1226
1286    1227
1287    1228
1288    1229
1289    1230
1290    1231
1291    1232
1292    1233
1293    1234
1294    1235
1295    1236
1296    1237
1297    1238
1298    1239
1299    1240
1300    1241
1301    1242
1302    1243
1303    1244
1304    1245
1305    1246
1306    1247
1307    1248
1308    1249
1309    1250
1310    1251
1311    1252
1312    1253
1313    1254
1314    1255
1315
```

## 4 Proje Süresince Yapılanlar

Proje süresince, `execv()` ve `fork()` komutlarını kullanarak diğer programlara geçiş yaptım. Pipe yapısını ve kullanımını öğrendim. Arka arkaya veri alıp vermeyi öğrendim. Makefile dosyasını yaptım. Satsutseç kullanarak bir char dizisinden random index ile değeri aldım. Determinantta 2x2 matrisler oluşturdum. Pipe - integer dizi ile kofaktore diziye yolladım. Pointer aritmatığı kullanarak dizi yollarken belleğe göre yer ayırarak yolladım. Fazla yer ayırdığım için aşağıdaki hatayı alıyordum fakat doğru çalışmaya devam ediyordu. Bu sorunu malloc değerlerini düzelterek çözdüm. (Yukarıda doğru çalışan halleri mevcuttur.)

```
9 eleman giriniz
11 if (sistem yap == 'h'){break;} else
14 int pipefd[2];
15 int fv;
16 int x[2][2]; int y[2][2]; int z[2][2];
17 if (pipe(pipefd) < 0) {
18     perror("pipe");
19     exit(1);
20 }
21 int dizi[3][3];
22 printf("9 eleman giriniz\n");
23 int j=0;
24 int i = 0;
25 int c;
26 for(i=0;i<3;i++) //Burada ise döngü yardımıyla matrisimi
27 {
28     for(j=0;j<3;j++)
29     {
30         random deger : sut1
31         *** stack smashing detected ***: <unknown> terminated
32         sut1 sonuc: -3
33         sut1 sonuc: 14
34         sut1 sonuc: 9
35         sut1 için determinant sonuc : 20
36     }
37 }
```

Projemde döngü ile tekrar çalıştırmak ister misiniz kısmı bulunmamaktadır. Bunu yapmaya çalıştığımda, evet denildiği zaman eski matrisin değerlerini getiriyordu. Bunu çözmek için flush ile temizlemeye denedim. Fakat işe yaramadı.

## 5 Kaynakça

<https://ozdemiryazilim.wordpress.com/tag/pipe-nedir/>