

OUTPUT AND INPUT OF FIRST PART

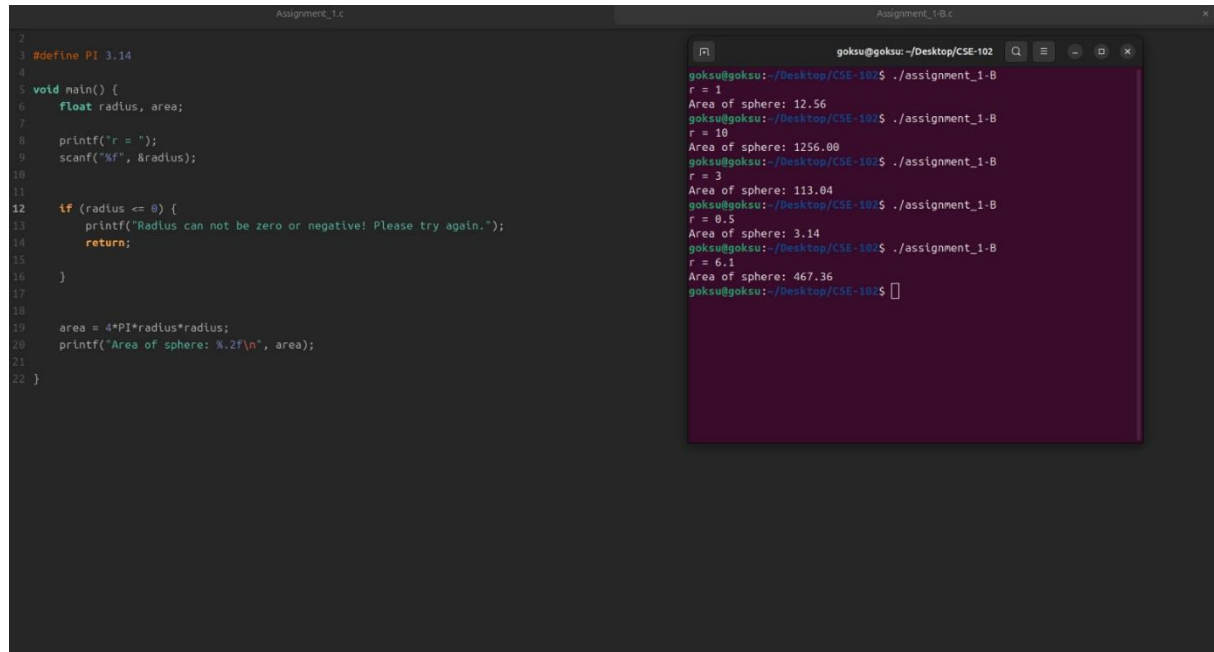
In this code firstly take three inputs as number1 number 2 and operator.

After that, check the operator at if-else statements and calculate experimental and expected results. According to experimental and expected results calculate error and error percentage and print results to the console. Repeat that four times. At the end calculate the average_error_value and finish program.

```
1 // main.c
2
3 int main() {
4     int number1, number2;
5     float experimental_result, expected_result, error, error_rate, total_error, total_expected_result, average_error_value;
6     char operator;
7     total_error = 0;
8     // 1000
9
10    printf("Enter an operation (Example: 3+2):");
11    scanf("%d %d %c", &number1, &operator, &number2);
12
13    if (operator == '+')
14    {
15        experimental_result = number1 + number2;
16        expected_result = number1 + number2;
17    }
18    else if (operator == '-')
19    {
20        experimental_result = number1 - number2;
21        expected_result = number1 - number2;
22    }
23    else if (operator == '*')
24    {
25        experimental_result = number1 * number2;
26        expected_result = number1 * number2;
27    }
28    else if (operator == '/')
29    {
30        if (number2 != 0) {
31            experimental_result = number1 / number2;
32            expected_result = number1 / number2;
33        }
34        else {
35            printf("Second number can not be zero! Please Try Again.\n");
36            return;
37        }
38    }
39    else
40    {
41        printf("Operator could not be found.\n");
42    }
43    error = expected_result - experimental_result;
44
45    if (error != 0)
46    {
47        error = error * (-1);
48    }
49    if (expected_result != 0) {
50        error_rate = (error / expected_result) * 100;
51        total_error = error_rate;
52        printf("Expected Result: %.2f\n", expected_result);
53        printf("Experimental Result: %.2f\n", experimental_result);
54        printf("Error: %.2f\n", error);
55        printf("Error Percentage: %.2f%%\n", error_rate);
56    }
57    else {
58        printf("Expected Result: %.2f\n", expected_result);
59    }
60
61    // 1000
62
63    // 1000
64
65    // 1000
66
67    // 1000
68
69    // 1000
70
71    // 1000
72
73    // 1000
74
75    // 1000
76
77    // 1000
78
79    // 1000
80
81    // 1000
82
83    // 1000
84
85    // 1000
86
87    // 1000
88
89    // 1000
90
91    // 1000
92
93    // 1000
94
95    // 1000
96
97    // 1000
98
99    // 1000
100
101    // 1000
102
103    // 1000
104
105    // 1000
106
107    // 1000
108
109    // 1000
110
111    // 1000
112
113    // 1000
114
115    // 1000
116
117    // 1000
118
119    // 1000
120
121    // 1000
122
123    // 1000
124
125    // 1000
126
127    // 1000
128
129    // 1000
130
131    // 1000
132
133    // 1000
134
135    // 1000
136
137    // 1000
138
139    // 1000
140
141    // 1000
142
143    // 1000
144
145    // 1000
146
147    // 1000
148
149    // 1000
150
151    // 1000
152
153    // 1000
154
155    // 1000
156
157    // 1000
158
159    // 1000
160
161    // 1000
162
163    // 1000
164
165    // 1000
166
167    // 1000
168
169    // 1000
170
171    // 1000
172
173    // 1000
174
175    // 1000
176
177    // 1000
178
179    // 1000
180
181    // 1000
182
183    // 1000
184
185    // 1000
186
187    // 1000
188
189    // 1000
190
191    // 1000
192
193    // 1000
194
195    // 1000
196
197    // 1000
198
199    // 1000
200
201    // 1000
202
203    // 1000
204
205    // 1000
206
207    // 1000
208
209    // 1000
210
211    // 1000
212
213    // 1000
214
215    // 1000
216
217    // 1000
218
219    // 1000
220
221    // 1000
222
223    // 1000
224
225    // 1000
226
227    // 1000
228
229    // 1000
230
231    // 1000
232
233    // 1000
234
235    // 1000
236
237    // 1000
238
239    // 1000
240
241    // 1000
242
243    // 1000
244
245    // 1000
246
247    // 1000
248
249    // 1000
250
251    // 1000
252
253    // 1000
254
255    // 1000
256
257    // 1000
258
259    // 1000
260
261    // 1000
262
263    // 1000
264
265    // 1000
266
267    // 1000
268
269    // 1000
270
271    // 1000
272
273    // 1000
274
275    // 1000
276
277    // 1000
278
279    // 1000
280
281    // 1000
282
283    // 1000
284
285    // 1000
286
287    // 1000
288
289    // 1000
290
291    // 1000
292
293    // 1000
294
295    // 1000
296
297    // 1000
298
299    // 1000
300
301    // 1000
302
303    // 1000
304
305    // 1000
306
307    // 1000
308
309    // 1000
310
311    // 1000
312
313    // 1000
314
315    // 1000
316
317    // 1000
318
319    // 1000
320
321    // 1000
322
323    // 1000
324
325    // 1000
326
327    // 1000
328
329    // 1000
330
331    // 1000
332
333    // 1000
334
335    // 1000
336
337    // 1000
338
339    // 1000
340
341    // 1000
342
343    // 1000
344
345    // 1000
346
347    // 1000
348
349    // 1000
350
351    // 1000
352
353    // 1000
354
355    // 1000
356
357    // 1000
358
359    // 1000
360
361    // 1000
362
363    // 1000
364
365    // 1000
366
367    // 1000
368
369    // 1000
370
371    // 1000
372
373    // 1000
374
375    // 1000
376
377    // 1000
378
379    // 1000
380
381    // 1000
382
383    // 1000
384
385    // 1000
386
387    // 1000
388
389    // 1000
390
391    // 1000
392
393    // 1000
394
395    // 1000
396
397    // 1000
398
399    // 1000
400
401    // 1000
402
403    // 1000
404
405    // 1000
406
407    // 1000
408
409    // 1000
410
411    // 1000
412
413    // 1000
414
415    // 1000
416
417    // 1000
418
419    // 1000
420
421    // 1000
422
423    // 1000
424
425    // 1000
426
427    // 1000
428
429    // 1000
430
431    // 1000
432
433    // 1000
434
435    // 1000
436
437    // 1000
438
439    // 1000
440
441    // 1000
442
443    // 1000
444
445    // 1000
446
447    // 1000
448
449    // 1000
450
451    // 1000
452
453    // 1000
454
455    // 1000
456
457    // 1000
458
459    // 1000
460
461    // 1000
462
463    // 1000
464
465    // 1000
466
467    // 1000
468
469    // 1000
470
471    // 1000
472
473    // 1000
474
475    // 1000
476
477    // 1000
478
479    // 1000
480
481    // 1000
482
483    // 1000
484
485    // 1000
486
487    // 1000
488
489    // 1000
490
491    // 1000
492
493    // 1000
494
495    // 1000
496
497    // 1000
498
499    // 1000
500
501    // 1000
502
503    // 1000
504
505    // 1000
506
507    // 1000
508
509    // 1000
510
511    // 1000
512
513    // 1000
514
515    // 1000
516
517    // 1000
518
519    // 1000
520
521    // 1000
522
523    // 1000
524
525    // 1000
526
527    // 1000
528
529    // 1000
530
531    // 1000
532
533    // 1000
534
535    // 1000
536
537    // 1000
538
539    // 1000
540
541    // 1000
542
543    // 1000
544
545    // 1000
546
547    // 1000
548
549    // 1000
550
551    // 1000
552
553    // 1000
554
555    // 1000
556
557    // 1000
558
559    // 1000
560
561    // 1000
562
563    // 1000
564
565    // 1000
566
567    // 1000
568
569    // 1000
570
571    // 1000
572
573    // 1000
574
575    // 1000
576
577    // 1000
578
579    // 1000
580
581    // 1000
582
583    // 1000
584
585    // 1000
586
587    // 1000
588
589    // 1000
590
591    // 1000
592
593    // 1000
594
595    // 1000
596
597    // 1000
598
599    // 1000
600
601    // 1000
602
603    // 1000
604
605    // 1000
606
607    // 1000
608
609    // 1000
610
611    // 1000
612
613    // 1000
614
615    // 1000
616
617    // 1000
618
619    // 1000
620
621    // 1000
622
623    // 1000
624
625    // 1000
626
627    // 1000
628
629    // 1000
630
631    // 1000
632
633    // 1000
634
635    // 1000
636
637    // 1000
638
639    // 1000
640
641    // 1000
642
643    // 1000
644
645    // 1000
646
647    // 1000
648
649    // 1000
650
651    // 1000
652
653    // 1000
654
655    // 1000
656
657    // 1000
658
659    // 1000
660
661    // 1000
662
663    // 1000
664
665    // 1000
666
667    // 1000
668
669    // 1000
670
671    // 1000
672
673    // 1000
674
675    // 1000
676
677    // 1000
678
679    // 1000
680
681    // 1000
682
683    // 1000
684
685    // 1000
686
687    // 1000
688
689    // 1000
690
691    // 1000
692
693    // 1000
694
695    // 1000
696
697    // 1000
698
699    // 1000
700
701    // 1000
702
703    // 1000
704
705    // 1000
706
707    // 1000
708
709    // 1000
710
711    // 1000
712
713    // 1000
714
715    // 1000
716
717    // 1000
718
719    // 1000
720
721    // 1000
722
723    // 1000
724
725    // 1000
726
727    // 1000
728
729    // 1000
730
731    // 1000
732
733    // 1000
734
735    // 1000
736
737    // 1000
738
739    // 1000
740
741    // 1000
742
743    // 1000
744
745    // 1000
746
747    // 1000
748
749    // 1000
750
751    // 1000
752
753    // 1000
754
755    // 1000
756
757    // 1000
758
759    // 1000
760
761    // 1000
762
763    // 1000
764
765    // 1000
766
767    // 1000
768
769    // 1000
770
771    // 1000
772
773    // 1000
774
775    // 1000
776
777    // 1000
778
779    // 1000
780
781    // 1000
782
783    // 1000
784
785    // 1000
786
787    // 1000
788
789    // 1000
790
791    // 1000
792
793    // 1000
794
795    // 1000
796
797    // 1000
798
799    // 1000
800
801    // 1000
802
803    // 1000
804
805    // 1000
806
807    // 1000
808
809    // 1000
810
811    // 1000
812
813    // 1000
814
815    // 1000
816
817    // 1000
818
819    // 1000
820
821    // 1000
822
823    // 1000
824
825    // 1000
826
827    // 1000
828
829    // 1000
830
831    // 1000
832
833    // 1000
834
835    // 1000
836
837    // 1000
838
839    // 1000
840
841    // 1000
842
843    // 1000
844
845    // 1000
846
847    // 1000
848
849    // 1000
850
851    // 1000
852
853    // 1000
854
855    // 1000
856
857    // 1000
858
859    // 1000
860
861    // 1000
862
863    // 1000
864
865    // 1000
866
867    // 1000
868
869    // 1000
870
871    // 1000
872
873    // 1000
874
875    // 1000
876
877    // 1000
878
879    // 1000
880
881    // 1000
882
883    // 1000
884
885    // 1000
886
887    // 1000
888
889    // 1000
890
891    // 1000
892
893    // 1000
894
895    // 1000
896
897    // 1000
898
899    // 1000
900
901    // 1000
902
903    // 1000
904
905    // 1000
906
907    // 1000
908
909    // 1000
910
911    // 1000
912
913    // 1000
914
915    // 1000
916
917    // 1000
918
919    // 1000
920
921    // 1000
922
923    // 1000
924
925    // 1000
926
927    // 1000
928
929    // 1000
930
931    // 1000
932
933    // 1000
934
935    // 1000
936
937    // 1000
938
939    // 1000
940
941    // 1000
942
943    // 1000
944
945    // 1000
946
947    // 1000
948
949    // 1000
950
951    // 1000
952
953    // 1000
954
955    // 1000
956
957    // 1000
958
959    // 1000
960
961    // 1000
962
963    // 1000
964
965    // 1000
966
967    // 1000
968
969    // 1000
970
971    // 1000
972
973    // 1000
974
975    // 1000
976
977    // 1000
978
979    // 1000
980
981    // 1000
982
983    // 1000
984
985    // 1000
986
987    // 1000
988
989    // 1000
990
991    // 1000
992
993    // 1000
994
995    // 1000
996
997    // 1000
998
999    // 1000
1000
1001    // 1000
1002
1003    // 1000
1004
1005    // 1000
1006
1007    // 1000
1008
1009    // 1000
1010
1011    // 1000
1012
1013    // 1000
1014
1015    // 1000
1016
1017    // 1000
1018
1019    // 1000
1020
1021    // 1000
1022
1023    // 1000
1024
1025    // 1000
1026
1027    // 1000
1028
1029    // 1000
1030
1031    // 1000
1032
1033    // 1000
1034
1035    // 1000
1036
1037    // 1000
1038
1039    // 1000
1040
1041    // 1000
1042
1043    // 1000
1044
1045    // 1000
1046
1047    // 1000
1048
1049    // 1000
1050
1051    // 1000
1052
1053    // 1000
1054
1055    // 1000
1056
1057    // 1000
1058
1059    // 1000
1060
1061    // 1000
1062
1063    // 1000
1064
1065    // 1000
1066
1067    // 1000
1068
1069    // 1000
1070
1071    // 1000
1072
1073    // 1000
1074
1075    // 1000
1076
1077    // 1000
1078
1079    // 1000
1080
1081    // 1000
1082
1083    // 1000
1084
1085    // 1000
1086
1087    // 1000
1088
1089    // 1000
1090
1091    // 1000
1092
1093    // 1000
1094
1095    // 1000
1096
1097    // 1000
1098
1099    // 1000
1100
1101    // 1000
1102
1103    // 1000
1104
1105    // 1000
1106
1107    // 1000
1108
1109    // 1000
1110
1111    // 1000
1112
1113    // 1000
1114
1115    // 1000
1116
1117    // 1000
1118
1119    // 1000
1120
1121    // 1000
1122
1123    // 1000
1124
1125    // 1000
1126
1127    // 1000
1128
1129    // 1000
1130
1131    // 1000
1132
1133    // 1000
1134
1135    // 1000
1136
1137    // 1000
1138
1139    // 1000
1140
1141    // 1000
1142
1143    // 1000
1144
1145    // 1000
1146
1147    // 1000
1148
1149    // 1000
1150
1151    // 1000
1152
1153    // 1000
1154
1155    // 1000
1156
1157    // 1000
1158
1159    // 1000
1160
1161    // 1000
1162
1163    // 1000
1164
1165    // 1000
1166
1167    // 1000
1168
1169    // 1000
1170
1171    // 1000
1172
1173    // 1000
1174
1175    // 1000
1176
1177    // 1000
1178
1179    // 1000
1180
1181    // 1000
1182
1183    // 1000
1184
1185    // 1000
1186
1187    // 1000
1188
1189    // 1000
1190
1191    // 1000
1192
1193    // 1000
1194
1195    // 1000
1196
1197    // 1000
1198
1199    // 1000
1200
1201    // 1000
1202
1203    // 1000
1204
1205    // 1000
1206
1207    // 1000
1208
1209    // 1000
1210
1211    // 1000
1212
1213    // 1000
1214
1215    // 1000
1216
1217    // 1000
1218
1219    // 1000
1220
1221    // 1000
1222
1223    // 1000
1224
1225    // 1000
1226
1227    // 1000
1228
1229    // 1000
1230
1231    // 1000
1232
1233    // 1000
1234
1235    // 1000
1236
1237    // 1000
1238
1239    // 1000
1240
1241    // 1000
1242
1243    // 1000
1244
1245    // 1000
1246
1247    // 1000
1248
1249    // 1000
1250
1251    // 1000
1252
1253    // 1000
1254
1255    // 1000
1256
1257    // 1000
1258
1259    // 1000
1260
1261    // 1000
1262
1263    // 1000
1264
1265    // 1000
1266
1267    // 1000
1268
1269    // 1000
1270
1271    // 1000
1272
1273    // 1000
1274
1275    // 1000
1276
1277    // 1000
1278
1279    // 1000
1280
1281    // 1000
1282
1283    // 1000
1284
1285    // 1000
1286
1287    // 1000
1288
1289    // 1000
1290
1291    // 1000
1292
1293    // 1000
1294
1295    // 1000
1296
1297    // 1000
1298
1299    // 1000
1300
1301    // 1000
1302
1303    // 1000
1304
1305    // 1000
1306
1307    // 1000
1308
1309    // 1000
1310
1311    // 1000
1312
1313    // 1000
1314
1315    // 1000
1316
1317    // 1000
1318
1319    // 1000
1320
1321    // 1000
1322
1323    // 1000
1324
1325    // 1000
1326
1327    // 1000
1328
1329    // 1000
1330
1331    // 1000
1332
1333    // 1000
1334
1335    // 1000
1336
1337    // 1000
1338
1339    // 1000
1340
1341    // 1000
1342
1343    // 1000
1344
1345    // 1000
1346
1347    // 1000
1348
1349    // 1000
1350
1351    // 1000
1352
1353    // 1000
1354
1355    // 1000
1356
1357    // 1000
1358
1359    // 1000
1360
1361    // 1000
1362
1363    // 1000
1364
1365    // 1000
1366
1367    // 1000
1368
1369    // 1000
1370
1371    // 1000
1372
1373    // 1000
1374
1375    // 1000
1376
1377    // 1000
1378
1379    // 1000
1380
1381    // 1000
1382
1383    // 1000
1384
1385    // 1000
1386
1387    // 1000
1388
1389    // 1000
1390
1391    // 1000
1392
1393    // 1000
1394
1395    // 1000
1396
1397    // 1000
1398
1399    // 1000
1400
1401    // 1000
1402
1403    // 1000
1404
1405    // 1000
1406
1407    // 1000
1408
1409    // 1000
1410
1411    // 1000
1412
1413    // 1000
1414
1415    // 1000
1416
1417    // 1000
1418
1419    // 1000
1420
1421    // 1000
1422
1423    // 1000
1424
1425    // 1000
1426
1427    // 1000
1428
1429    // 1000
1430
1431    // 1000
1432
1433    // 1000
1434
1435    // 1000
1436
1437    // 1000
1438
1439    // 1000
1440
1441    // 1000
1442
1443    // 1000
1444
1445    // 1000
1446
1447    // 1000
1448
1449    // 1000
1450
1451    // 1000
1452
1453    // 1000
1454
1455    // 1000
1456
1457    // 1000
1458
1459    // 1000
1460
1461    // 1000
1462
1463    // 1000
1464
1465    // 1000
1466
1467    // 1000
1468
1469    // 1000
1470
1471    // 1000
1472
1473    // 1000
1474
1475    // 1000
1476
1477    // 1000
1478
1479    // 1000
1480
1481    // 1000
1482
1483    // 1000
1484
1485    // 1000
1486
1487    // 1000
1488
1489    // 1000
1490
1491    // 1000
1492
1493    // 1000
1494
1495    // 1000
1496
1497    // 1000
1498
1499    // 1000
1500
1501    // 1000
1502
1503    // 1000
1504
1505    // 1000
1506
1507    // 1000
1508
1509    // 1000
1510
1511    // 1000
1512
1513    // 1000
1514
1515    // 1000
1516
1517    // 1000
1518
1519    // 1000
1520
1521    // 1000
1522
1523    // 1000
1524
1525    // 1000
1526
1527    // 1000
1528
1529    // 1000
1530
1531    // 1000
1532
1533    // 1000
1534
1535    // 1000
1536
1537    // 1000
1538
1539    // 10
```

OUTPUT AND INPUT OF SECOND PART

Take input from the user as radius. Then calculate the area and print to the console.



The image shows a C program in a text editor and its execution in a terminal. The program calculates the surface area of a sphere based on a user-provided radius. It includes a validation step for non-negative radii.

```
2
3 #define PI 3.14
4
5 void main() {
6     float radius, area;
7
8     printf("r = ");
9     scanf("%f", &radius);
10
11
12     if (radius <= 0) {
13         printf("Radius can not be zero or negative! Please try again.");
14         return;
15     }
16
17
18     area = 4*PI*radius*radius;
19     printf("Area of sphere: %.2f\n", area);
20
21
22 }
```

The terminal output shows the program being run with various inputs and the corresponding calculated area of a sphere:

```
goksu@goksu: ~/Desktop/CSE-102$ ./assignment_1-B
r = 1
Area of sphere: 12.56
goksu@goksu: ~/Desktop/CSE-102$ ./assignment_1-B
r = 10
Area of sphere: 1256.00
goksu@goksu: ~/Desktop/CSE-102$ ./assignment_1-B
r = 3
Area of sphere: 113.04
goksu@goksu: ~/Desktop/CSE-102$ ./assignment_1-B
r = 0.5
Area of sphere: 3.14
goksu@goksu: ~/Desktop/CSE-102$ ./assignment_1-B
r = 6.1
Area of sphere: 467.36
goksu@goksu: ~/Desktop/CSE-102$
```