

## CLASS WORK PROGRAMS

### 1. C Program to Reverse a Number Using FOR Loop.

The screenshot shows the Dev-C++ IDE interface. On the left, the code editor displays the following C program:

```
1 #include <stdio.h>
2
3 void main()
4 {
5     int num,r,sum=0,t;
6
7     printf("Input a number: ");
8     scanf("%d",&num);
9
10    for(t=num;num!=0;num=num/10){
11        r=num % 10;
12        sum=sum*10+r;
13    }
14    printf("The number in reverse order is : %d \n",sum);
15 }
```

To the right of the code editor is a terminal window titled "C:\Users\Del\Documents\class work-1.exe". It shows the output of the program when run with the input "79". The output is:

```
Input a number: 79
The number in reverse order is : 978

Process exited after 3.08 seconds with return value 37
Press any key to continue . . .
```

Below the terminal window, the "Compiler" tab of the status bar shows compilation results:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Del\Documents\class work-1.exe
- Output Size: 128.1015625 KiB
- Compilation Time: 0.73s
```

The status bar at the bottom of the IDE shows the current line (Line: 14), column (Col: 2), selection (Sel: 0), lines (Lines: 14), length (Length: 266), and the message "Done parsing in 0.031 seconds".

### 2. Write a program in C such that the program will read the elements of a one-dimensional array, then compares the elements and finds which the largest two elements are in a given array.

```

C:\Users\gnane\OneDrive\Documents\2nd ques.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Profiling

Project Classes Debug /* reversein.c.cpp errors1.cpp error2.cpp error3.cpp error4.cpp Untitled6.cpp error11.cpp Untitled22.cpp Untitled23.cpp 2nd ques.cpp
main() int
1 #include <stdio.h>
2 int main ()
3 {
4     int n = 0, i = 0, largest1 = 0, largest2 = 0, temp = 0;
5     printf ("Enter the size of the array\n");
6     scanf ("%d", &n);
7     int array[n];
8     printf ("Enter the elements\n");
9     for (i = 0; i < n; i++)
10    {
11        scanf ("%d", &array[i]);
12    }
13
14    printf ("The array elements are : \n");
15    for (i = 0; i < n; i++)
16    {
17        printf ("%d\t", array[i]);
18    }
19
20    printf ("\n");
21
22    largest1 = array[0];
23    largest2 = array[1];
25
26    if (largest1 < largest2)
27    {
28        temp = largest1;
29        largest1 = largest2;
30        largest2 = temp;
31    }
32
33
34
35
36
37
38
39
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
99
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
118
119
119
120
121
122
123
124
125
126
127
128
129
129
130
131
132
133
134
135
136
137
138
139
139
140
141
142
143
144
145
146
147
148
149
149
150
151
152
153
154
155
156
157
158
159
159
160
161
162
163
164
165
166
167
168
169
169
170
171
172
173
174
175
176
177
178
179
179
180
181
182
183
184
185
186
187
188
189
189
190
191
192
193
194
195
196
197
198
199
199
200
201
202
203
204
205
206
207
208
209
209
210
211
212
213
214
215
216
217
218
219
219
220
221
222
223
224
225
226
227
228
229
229
230
231
232
233
234
235
236
237
238
239
239
240
241
242
243
244
245
246
247
248
249
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
338
339
339
340
341
342
343
344
345
346
347
348
349
349
350
351
352
353
354
355
356
357
358
359
359
360
361
362
363
364
365
366
367
368
369
369
370
371
372
373
374
375
376
377
378
379
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
418
419
419
420
421
422
423
424
425
426
427
428
429
429
430
431
432
433
434
435
436
437
438
439
439
440
441
442
443
444
445
446
447
448
449
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
479
479
480
481
482
483
484
485
486
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
518
519
519
520
521
522
523
524
525
526
527
528
529
529
530
531
532
533
534
535
536
537
538
539
539
540
541
542
543
544
545
546
547
548
549
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
688
689
689
690
691
692
693
694
695
696
697
697
698
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
788
789
789
790
791
792
793
794
795
796
797
797
798
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
838
839
839
840
841
842
843
844
845
846
847
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
869
869
870
871
872
873
874
875
876
877
878
879
879
880
881
882
883
884
885
886
887
888
889
889
890
891
892
893
894
895
896
897
897
898
899
900
901
902
903
904
905
906
907
908
909
909
910
911
912
913
914
915
916
917
918
919
919
920
921
922
923
924
925
926
927
928
929
929
930
931
932
933
934
935
936
937
938
939
939
940
941
942
943
944
945
946
947
948
949
949
950
951
952
953
954
955
956
957
958
959
959
960
961
962
963
964
965
966
967
968
969
969
970
971
972
973
974
975
976
977
978
979
979
980
981
982
983
984
985
986
987
987
988
989
989
990
991
992
993
994
995
995
996
997
998
999
999
1000
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1089
1090
1091
1092
1093
1094
1095
1096
1097
1097
1098
1099
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1189
1190
1191
1192
1193
1194
1195
1196
1196
1197
1198
1199
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1297
1298
1299
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1388
1389
1390
1391
1392
1393
1394
1395
1396
1396
1397
1398
1399
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1488
1489
1490
1491
1492
1493
1494
1495
1496
1496
1497
1498
1499
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1559
1560
1561
1562
1563
1564
1565
1566
156
```

4. Write a program to search the given element using binary search method and display its position in a linear array.

Sample Input;

Array of elements = {28, 12, 15, 122, 10, 33, 11}

Element to search = 15

SampleOutput:

Given element 23 is found at 3 rd position

The screenshot shows the Dev-C++ IDE interface. On the left, the code editor displays a C program named '4th qua.cpp'. The code implements a linear search algorithm. It defines an array 'a' with elements {28, 40, 30, 11, 57, 41, 25, 14, 52}, initializes 'val' to 41, and finds the index of 'val' in the array. The output window shows the array elements, the search value, the result indicating the element is present at index 6, and the execution time. The bottom status bar shows compilation results with 0 errors and 0 warnings, and the output file size as 152.939453125 Kib.

```
#include <stdio.h>
int linearSearch(int a[], int n, int val) {
    // Going through array sequentially
    for (int i = 0; i < n; i++) {
        if (a[i] == val)
            return i+1;
    }
    return -1;
}
int main() {
    int a[] = {28, 40, 30, 11, 57, 41, 25, 14, 52}; // given array
    int val = 41; // value to be searched
    int n = sizeof(a) / sizeof(a[0]); // size of array
    int res = linearSearch(a, n, val); // Store result
    printf("The elements of the array are - ");
    for (int i = 0; i < n; i++)
        printf("%d ", a[i]);
    printf("\nElement to be searched is - %d", val);
    if (res == -1)
        printf("\nElement is not present in the array");
    else
        printf("\nElement is present at %d position of array", res);
    return 0;
}
```

5. C Program to Find Sum of Natural Numbers Using While Loop

```

C:\Users\gnane\OneDrive\Documents\5 ques.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDN-GCC 4.9.2 64-bit Profiling

(globals)

Project Classes Debug [*] Untitled! 3rd ques.cpp 4th ques.cpp 5 ques.cpp
main(): int
1 #include <stdio.h>
2 #include <conio.h>
3 int main()
4 {
5     int num, i, sum = 0; // declare local variables
6     printf(" Enter a positive number: ");
7     scanf("%d", &num);
8     for (i = 0; i <= num; i++)
9     {
10         sum = sum + i;
11     }
12     // display the sum of natural number
13     printf("\n Sum of the first %d number is: %d", num, sum);
14     getch();
15 }

Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\gnane\OneDrive\Documents\5 ques.exe
- Output Size: 153.2412109375 KiB
- Compilation Time: 0.24s

Line: 3 Col: 4 Sel: 0 Lines: 15 Length: 400 Insert Done parsing in 0.015 seconds

```

## 6. C Program to Find GCD of two Numbers Using For Loop

```

C:\Users\gnane\OneDrive\Documents\6th qua.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDN-GCC 4.9.2 64-bit Profiling

(globals)

Project Classes Debug [*] Untitled! 3rd ques.cpp 4th ques.cpp 5 ques.cpp 6th ques.cpp
main(): int
1 #include <stdio.h>
2
3 int main()
4 {
5     int Num1, Num2, i, GCD;
6
7     printf("Please Enter two integer Values \n");
8     scanf("%d %d", &Num1, &Num2);
9
10    for(i = 1; i <= Num1 && i <= Num2; i++)
11    {
12        if(Num1 % i == 0 && Num2 % i == 0)
13        {
14            GCD = i;
15        }
16    }
17    printf("GCD of %d and %d is = %d", Num1, Num2, GCD);
18    return 0;
}

Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\gnane\OneDrive\Documents\6th qua.exe
- Output Size: 153.240234375 KiB
- Compilation Time: 0.17s

Line: 18 Col: 2 Sel: 0 Lines: 18 Length: 359 Insert Done parsing in 0 seconds

```

## 7. Predict the output or error(s) for the following program

```

main()
{
int i=5;
printf("%d%d%d%d%d",i++,i--,++i,--i,i);
}

```

The screenshot shows the Dev-C++ IDE interface. The code editor displays the following C program:

```

1 #include<stdio.h>
2 int main()
3 {
4     int i=5;
5     printf("%d%d%d%d%d",i++,i--,++i,--i,i);
6     return 0;
7 }
8

```

To the right of the code editor is a terminal window titled "Select C:\Users\gnane\OneDrive\Documents\errors 1.exe". It shows the output of the program:

```

Process exited after 0.2008 seconds with return value 0
Press any key to continue . . .

```

Below the code editor is the compiler results window, which shows:

- Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\gnane\OneDrive\Documents\errors 1.exe
- Output Size: 152.9013671875 KiB
- Compilation Time: 0.17s

The system tray at the bottom right shows the date and time as 23-08-2023 12:23.

### 8. Predict the output or error(s) for the following program

```

main()
{
char s[ ]="man";
int i;
for(i=0;s[ i ];i++)
printf("\n%c%c%c%c",s[ i ],*(s+i),*(i+s),i[s]);
}

```

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named 'error 2.cpp' with the following code:

```
1 #include<stdio.h>
2 main()
3 {
4     char s[]="man";
5     int i;
6     for(i=0;s[ i ];i++)
7         printf("\n%d%d%d%d",s[ i ],*(s+i),*(i+s),i[s]);
8     return 0;
9 }
10
```

To the right of the code editor is a terminal window titled 'C:\Users\gnane\OneDrive\Documents\error 2.exe' showing the program's output:

```
10100100100
07970797
100100100100
Process exited after 0.1896 seconds with return value 0
Press any key to continue . . .
```

Below the code editor is a 'Compiler' tab with the following compilation results:

- Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\gnane\OneDrive\Documents\error 2.exe
- Output Size: 152.9013671875 KiB
- Compilation Time: 0.17s

The taskbar at the bottom of the screen shows various application icons.

9. Predict the output or error(s) for the following program

```
#include<stdio.h>

main()
{
    char s[]={‘a’,’b’,’c’,’\n’,’c’,’\0’};
    char *p,*str,*str1;
    p=&s[3];

    str=p;
    str1=s;
    printf("%d",++*p + ++*str1-32);
}
```

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named 'error 3.cpp' with the following code:

```
1 #include<stdio.h>
2 main()
3 {
4     char s[]={ 'a','b','c','\n','c','\0'};
5     char *p,*str,*str1;
6     p=&s[3];
7
8     str=p;
9     str1=s;
10    printf("%d",++*p + ++*str1-32);
11    return 0;
12 }
13
```

To the right of the code editor is a terminal window showing the execution results:

```
C:\Users\gnane\OneDrive\Documents\error 3.exe
Process exited after 0.1731 seconds with return value 0
PRESS ANY KEY TO CONTINUE . . .
```

Below the code editor is the Compiler Log window, which shows the following compilation results:

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\gnane\OneDrive\Documents\error 3.exe
- Output Size: 152.9013671875 Kib
- Compilation Time: 0.17s

The system tray at the bottom right indicates the date and time as 23-03-2023 12:36.

10. Predict the output or error(s) for the following program

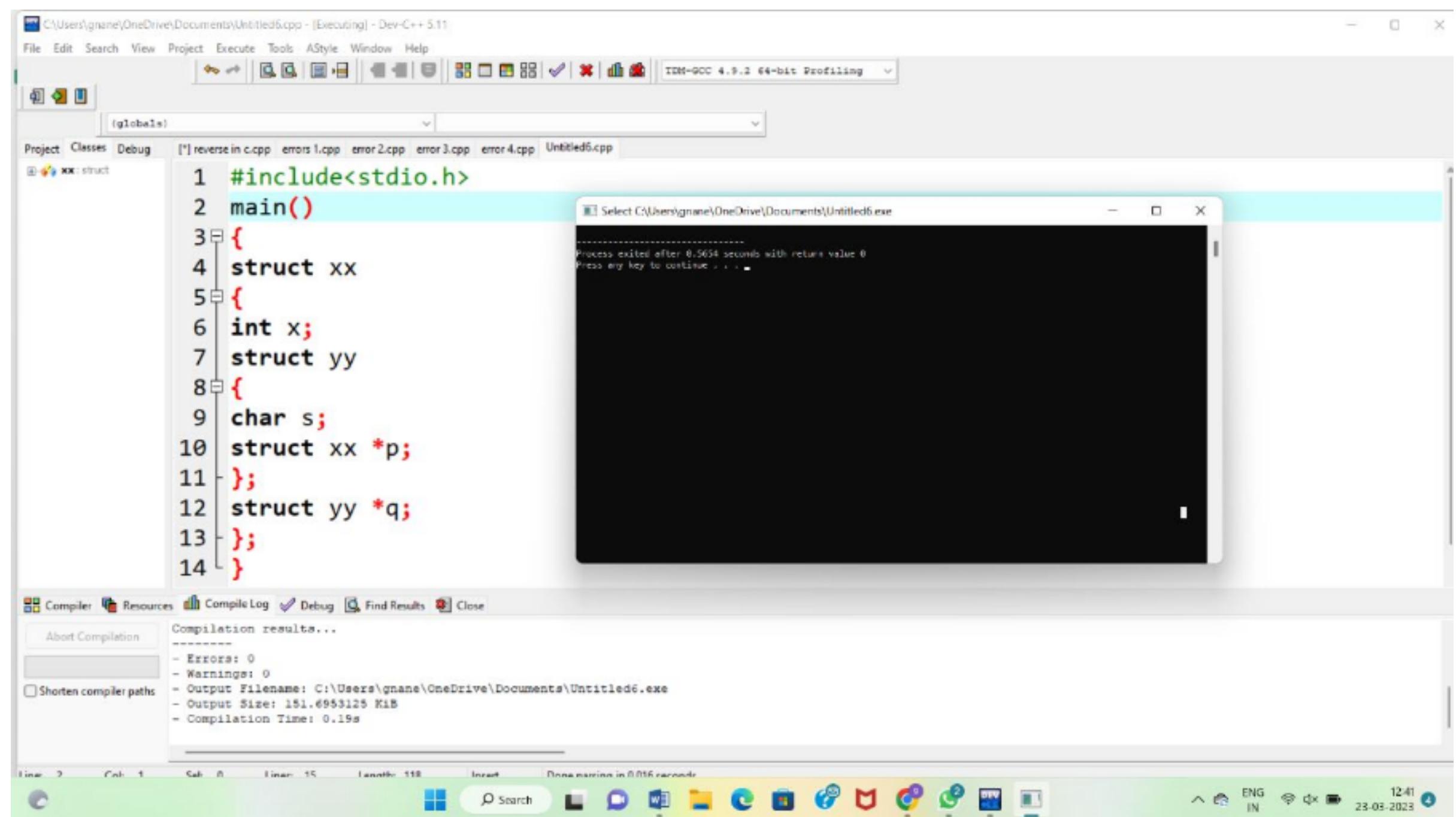
```
main()

{

struct xx
{
int x;
struct yy
{
char s;
struct xx *p;
};

struct yy *q;
};

}
```



11. Predict the output or error(s) for the following program

```
main()
{
char *p;
p="Hello";
printf("%c\n",*&p);
}
```

The screenshot shows the Dev-C++ IDE interface. The code editor window displays the following C code:

```

1 #include<stdio.h>
2 main()
3 {
4     char *p;
5     p=("Hello");
6     printf("%c\n",*&p);
7     return 0;
8 }
9

```

To the right of the code editor is a terminal window titled "C:\Users\gname\OneDrive\Documents\error 11.exe" showing the output:

```

Process exited after 0.101 seconds with return value 0
Press any key to continue . . .

```

Below the code editor is the compiler log window, which shows a warning:

```

Line Col File Message
5   2   C:\Users\gname\OneDrive\Documents\error 11.cpp In function 'int main()':
      [Warning] deprecated conversion from string constant to 'char' [-Wwrite-strings]

```

12. Write a program to enter the marks of n student in four subjects using structure. Then calculate the total and aggregate, display the grade obtained by the student. If the student scores an aggregate greater than 75%, then the grade is Distinction. If aggregate is  $60 \geq$  and  $< 75$ , then the grade is First Division. If aggregate is  $50 \geq$  and  $< 60$ , then the grade is Second Division. If aggregate is  $40 \geq$  and  $< 50$ , then the grade is Third Division. Else the grade is Fail.

**Sample Input & Output:**

Enter the marks in python: 90

Enter the marks in c programming: 91

Enter the marks in Mathematics: 92

Enter the marks in Physics: 93

Total= 366

Aggregate = 91.5

DISTINCTION

**Test cases:**

1. 18, 76, 93, 65
2. 73, 78, 79, 75
3. 98, 106, 120, -95
4. 96, 73, AA, 95

## 5. 78,59.8,\*\*,79

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named '12 th qu.cpp' with code for calculating grades based on marks. The output window shows the program's execution results.

```
#include<stdio.h>
int main()
{
    int marks;
    printf("Enter your marks ");
    scanf("%d",&marks);
    if(marks<0 || marks>100)
    {
        printf("Wrong Entry");
    }
    else if(marks<50)
    {
        printf("Grade F");
    }
    else if(marks>=50 && marks<60)
    {
        printf("Grade D");
    }
    else if(marks>=60 && marks<70)
    {
        printf("Grade C");
    }
    else if(marks>=70 && marks<80)
    {
        printf("Grade B");
    }
}
```

Output window content:

```
C:\Users\gnane\OneDrive\Documents\12 th qu.exe
Enter your marks 98
Grade A
Process exited after 7.736 seconds with return value 0
Press any key to continue . . .
```

## 13. Print the following hill pattern

A B C D E D C B A

A B C D C B A

A B C B A

A B A

A

C:\Users\gnane\OneDrive\Documents\13 th.cpp - [Executing] - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Profiling

(globals)

Project Classes Debug 12th quc.cpp 13 th.cpp

```

1 #include <stdio.h>
2 int main() {
3     int i, j, rows;
4     printf("Enter the number of rows: ");
5     scanf("%d", &rows);
6     for (i = 1; i <= rows; ++i) {
7         for (j = 1; j <= i; ++j) {
8             printf("* ");
9         }
10        printf("\n");
11    }
12    return 0;
13 }
14

```

Process exited after 4.846 seconds with return value 0  
Press any key to continue . . .

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\gnane\OneDrive\Documents\13 th.exe
- Output Size: 153.2431640625 KiB
- Compilation Timer: 0.22s

Line: 14 Col: 1 Sel: 0 Lines: 14 Length: 267 Insert Done parsing in 0.016 seconds

## 14. Write a program for matrix multiplication?

C:\Users\gnane\OneDrive\Documents\14.cpp - [Executing] - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Profiling

(globals)

Project Classes Debug 12th quc.cpp 13 th.cpp 14.cpp

```

1 #include<stdio.h>
2 #include<stdlib.h>
4 int main()
5 {
6     int a[10][10], b[10][10], mul[10][10], r, c, i, j, k;
7     system("cls");
8     printf("enter the number of rows");
9     scanf("%d",&r);
10    printf("enter the number of columns");
11    scanf("%d",&c);
12    printf("enter the first matrix element-\n");
13    for(i=0;i<r;i++)
14    {
15        for(j=0;j<c;j++)
16        {
17            scanf("%d",&a[i][j]);
18        }
19    }
20    printf("enter the second matrix element-\n");
21    for(i=0;i<r;i++)
22    {
23        for(j=0;j<c;j++)
24        {
25            scanf("%d",&b[i][j]);
26        }
27    }
28    printf("multiply of the matrix-\n");
29    for(i=0;i<r;i++)
30    {

```

Process exited after 23.54 seconds with return value 0  
Press any key to continue . . .

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\gnane\OneDrive\Documents\14.exe
- Output Size: 154.0830078125 KiB
- Compilation Timer: 0.19s

Line: 50 Col: 2 Sel: 0 Lines: 50 Length: 930 Insert Done parsing in 0 seconds

**15.** Write a Program to display the diagonal elements in a matrix array and also find the sum of them.

Sample input:

1 4 3

4 2 6

7 1 3

Output:

Diagonal Elements are 1 2 3

Sum of diagonal elements = 6

The screenshot shows the Dev-C++ IDE interface. The code editor window displays a C program named '15th ques.cpp' which reads a square matrix from the user and prints its diagonal elements along with their sum. The output window shows the execution of the program, including user input for matrix size and elements, the printed matrix, and the calculated sum of 6. The compiler log window at the bottom shows a successful compilation with no errors or warnings.

```
#include<stdio.h>
int main()
{
    int mat[12][12];
    int i,j,row,col,sum=0;
    printf("Enter the number of rows and columns for 1st matrix\n");
    scanf("%d%d",&row,&col);
    printf("Enter the elements of the matrix\n");
    for(i=0;i<row;i++)
    {
        for(j=0;j<col;j++)
        {
            scanf("%d",&mat[i][j]);
        }
    }
    printf("The matrix\n");
    for(i=0;i<row;i++)
    {
        for(j=0;j<col;j++)
        {
            printf("%d\t",mat[i][j]);
        }
        printf("\n");
    }
    //To add diagonal elements
    for(i=0;i<row;i++)
    {
        for(j=0;j<col;j++)
        {
            if(i==j)
                sum+=mat[i][j];
        }
    }
    printf("The sum of diagonal elements of a square matrix = %d",sum);
}
```

**16.** Write a program in C to swap elements using call by reference.

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named 'work-12.c' which swaps two integers. The code uses pointers to swap values. The output window shows the program's execution: it asks for input values (x=7, y=9), prints the initial state, calls the swap function, and then prints the swapped state (x=9, y=7). The compiler log window shows no errors or warnings.

```
#include<stdio.h>
void swap(int *a, int *b);
int main()
{
    int x, y;
    printf("Enter the value of x and y: ");
    scanf("%d %d",&x,&y);
    printf("Before Swapping: x = %d, y = %d\n", x, y);
    swap(&x, &y); // passing addresses of x and y to the function
    printf("After Swapping: x = %d, y = %d\n", x, y);
    return 0;
}
void swap(int *a, int *b)
{
    int temp;
    temp = *a;
    *a = *b;
    *b = temp;
}
```

## 17. Write C program to Copy file contents to another file.

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named 'work-17.c' which copies the contents of 'source.txt' to 'destination.txt'. The code uses FILE pointers to read from one file and write to another. The output window shows an error message: 'Error opening files!' followed by the program's exit status (1). The compiler log window shows no errors or warnings.

```
#include <stdio.h>
int main()
{
    FILE *source_file, *destination_file;
    char ch;

    source_file = fopen("source.txt", "r");
    destination_file = fopen("destination.txt", "w");
    if (source_file == NULL || destination_file == NULL)
    {
        printf("Error opening files!");
        return 1;
    }
    while ((ch = fgetc(source_file)) != EOF)
    {
        fputc(ch, destination_file);
    }
    printf("File copied successfully!");
    fclose(source_file);
    fclose(destination_file);
    return 0;
}
```

## 18. Write a program in C to read the file and store the lines into an array.

**Test Data :**

Input the file name to be opened : test.txt

**Expected Output :**

The content of the file test.txt are :

test line 1  
test line 2  
test line 3  
test line 4

**Test cases**

Sample.txt  
sample1.txt  
S123.txt  
S@12.txt  
Sample S.txt

The screenshot shows the Dev-C++ IDE interface. The code editor displays a C++ program named 'qus.cpp' with code for reading lines from a file and printing them. The terminal window shows the application running, prompting for a filename and displaying the content of 'test.txt'. The compiler results window shows no errors or warnings.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define LSIZ 128
#define RSIZ 128

int main(void)
{
    char line[RSIZ][LSIZ];
    char fname[20];
    FILE *fptr = NULL;
    int i = 0;
    int tot = 0;
    printf("\n\n Read the file and store the lines into an array :\n");
    printf("-----\n");
    printf(" Input the filename to be opened : ");
    scanf("%s", fname);

    fptr = fopen(fname, "r");
    while(fgets(line[i], LSIZ, fptr))
    {
        line[i][strlen(line[i]) - 1] = '\0';
        i++;
    }
    tot = i;
    printf("\n The content of the file %s are : \n", fname);
    for(i = 0; i < tot; +i)
    {
        printf("%s\n", line[i]);
    }
}
```

**19.** Two strings consisting binary value of two numbers. Print the sum of the numbers in binary.

Input1: s1=" 1101" , s2=" 100101"

Result: " 110010"

Input2: s1=" 11100100111101" ,s2=" 10000"

Result: " 111001010001101"

The screenshot shows the Dev-C++ IDE interface. The code editor displays a C program named '19.cpp' which adds two binary strings. The main function includes a 'sum' function that takes two character arrays and their lengths. It uses a loop to iterate through the strings, calculate the sum of bits, and handle carry. A separate loop handles any leftover characters from the first string. The compiler log shows no errors or warnings, and the output window shows the execution of the program, inputting binary numbers 11 and 10, and displaying the result 11 + 10 = 110.

```
#include <stdio.h>
#include <string.h>
int main()
{
    sum("11", "10");
    return 0;
}
void sum(char b1[], char b2[], int l1, int l2)
{
    int carry = 0, temp, num1, num2, i;
    char result[100];
    result[l1 + 1] = '\0';
    while (l2 > 0) {
        num1 = b1[l1 - 1] - '0';
        num2 = b2[l2 - 1] - '0';
        temp = num1 + num2 + carry;
        if (temp >= 2) {
            carry = 1;
            temp = temp % 2;
        }
        result[l1] = temp + '0';
        l1--;
        l2--;
    }
    // This loop will add leftover
    // characters of first strings.
    while (l1 - 1 >= 0) {
        temp = b1[l1 - 1] + carry - '0';
        if (temp >= 2) {
            carry = 1;
            temp = temp % 2;
        }
        result[l1] = temp + '0';
        l1--;
    }
}
```

20. Write a Program to illustrates the use of continue statement.

The screenshot shows the Dev-C++ IDE interface. The code editor displays a C program named 'class work-20.c' that prints odd numbers from 1 to 10. It uses a for loop with a continue statement for even numbers. The output window shows the execution of the program, printing the numbers 1, 3, 5, 7, and 9.

```
#include <stdio.h>
int main()
{
    int i;
    for (i = 1; i <= 10; i++)
    {
        if (i % 2 == 0)
        {
            continue;
        }
        printf("%d\n", i);
    }
    return 0;
}
```

21. Write a program to calculate square and cube of an entered number. Use function as an argument

The screenshot shows the Dev-C++ IDE interface. The code editor window displays a C program named 'class work-21.c'. The code defines a main function that prompts the user for a number, calls a 'calculate' function to get its square and cube, and then prints the results. The 'calculate' function uses pointers to return the values. The compiler log window at the bottom shows that the compilation was successful with no errors or warnings.

```
#include <stdio.h>
void calculate(int num, int (*square)(int), int (*cube)(int));
int square(int num);
int cube(int num);
int main()
{
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    calculate(num, square, cube);
    return 0;
}
void calculate(int num, int (*square)(int), int (*cube)(int))
{
    int sq = square(num);
    int cb = cube(num);
    printf("The square of %d is %d\n", num, sq);
    printf("The cube of %d is %d\n", num, cb);
}
int square(int num)
{
    return num * num;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\DELL\Documents\class work-21.exe
- Output Size: 129.1640625 KiB
- Compilation Time: 0.47s

22. What is the output of the program given below?

```
main()
{
    signed char i=0;
    for(;i>=0;i++);
    printf(" %d\n" ,i);
}
```

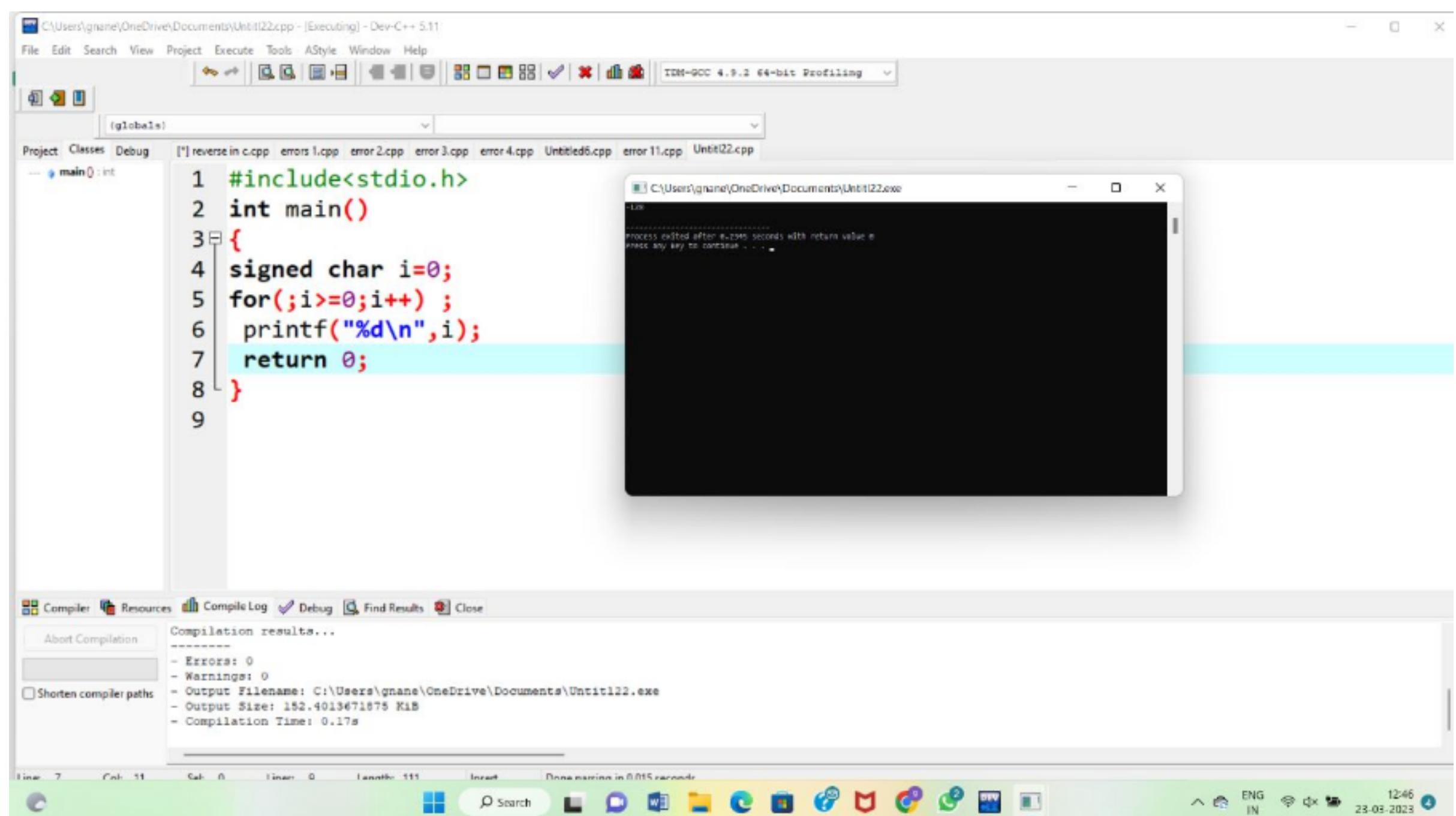
22. What is the output of the program given below?

```
int aaa()
{
    printf(" Hi" );
}
int bbb()
{
    printf(" hello" );
}
int ccc()
{
```

```

printf(" bye" );
}
main()
{
    int ( * ptr[3] ) ();
    ptr[0] = aaa;
    ptr[1] = bbb;
    ptr[2] =ccc;
    ptr[2]();
}

```



**23.** Find out the logic missing statements and display the final output.

```

#include <stdio.h>
#include<conio.h>

main()
{
    int i, n, d, sum;
    clrscr();

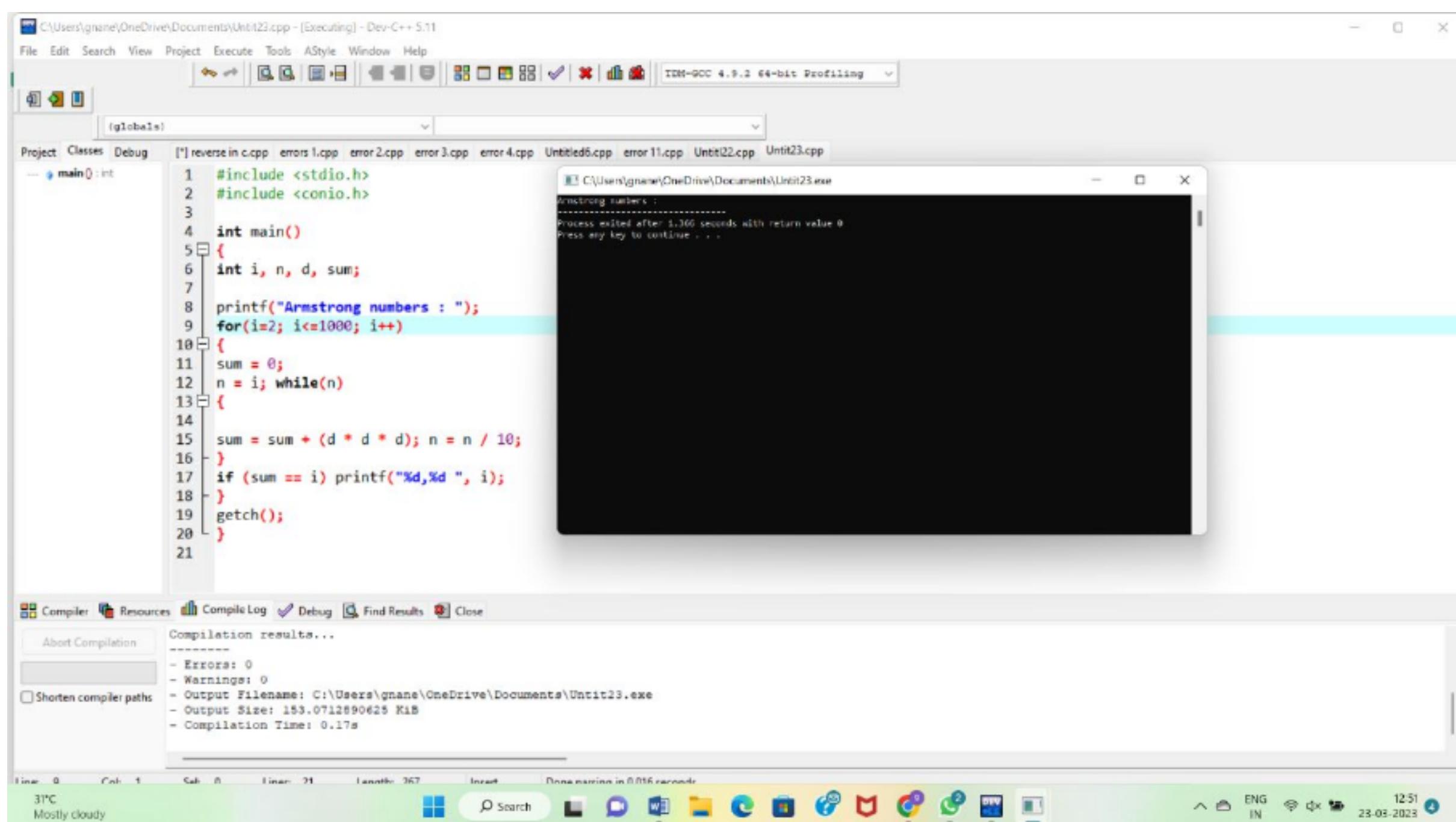
    printf("Armstrong numbers :"); for(i=2;
    i<=1000; i++)
    {

```

```

sum = 0;
n = i; while(n)
{
    sum = sum + (d * d *d); n = n /
    10;
}
if (sum == i) printf("%d ",i);
}
getch();
}

```



## 24. To generate employee payroll for an organization using structure.

1. Define *employee* structure with fields *empid*, *ename*, *basic*, *hra*, *da*, *it*, *gross* and *netpay*
2. Read number of employees *n*
3. Read *empid*, *ename*, and *basic* for *n* employees in an array of structure.
4. For each employee, compute
  - hra* = 2% of *basic*
  - da* = 1% of *basic*
  - gross* = *basic* + *hra* +
  - da it* = 5% of *basic*
  - netpay* = *gross* - *it*
5. Print *empid*, *ename*, *basic*, *hra*, *da*, *it*, *gross* and *netpay* for all employees

**25.** To implement a simple calculator using switch case statement.

Read the *operator* symbol and operands  $n1, n2$

If *operator* = + then calculate

*result* =  $n1 + n2$

Else if *operator* = – then

calculate *result* =  $n1 - n2$

Else if *operator* = \* then

calculate *result* =  $n1 * n2$

Else if *operator* = / then

calculate *result* =  $n1 / n2$

Else if *operator* = % then

calculate *result* =  $n1 \% n2$

Else

print "Invalid operator"

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named 'class work-25.c' with code for performing arithmetic operations (+, -, \*, /) on two double precision numbers. The code includes input validation for division by zero and invalid operators. The output window shows the execution of the program, where it prompts for an operator and two operands, then prints the result of the operation.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     char operator;
6     double num1, num2, result;
7
8     printf("Enter an operator (+, -, *, /): ");
9     scanf("%c", &operator);
10
11    printf("Enter two operands: ");
12    scanf("%lf %lf", &num1, &num2);
13
14    switch (operator)
15    {
16        case '+':
17            result = num1 + num2;
18            printf("%.2lf + %.2lf = %.2lf", num1, num2, result);
19            break;
20
21        case '-':
22            result = num1 - num2;
23            printf("%.2lf - %.2lf = %.2lf", num1, num2, result);
24            break;
25
26        case '*':
27            result = num1 * num2;
28            printf("%.2lf * %.2lf = %.2lf", num1, num2, result);
29            break;
30
31        case '/':
32            if (num2 == 0)
33            {
34                printf("Error: Division by zero");
35            }
36            else
37            {
38                result = num1 / num2;
39                printf("%.2lf / %.2lf = %.2lf", num1, num2, result);
40            }
41            break;
42
43        default:
44            printf("Error: Invalid operator");
45    }
46 }

```

**26.** To find the sum of the digits of a given number using while statement. . .

Read  $num$

Initialize  $sum$  to 0.

Repeat until  $num = 0$

    Obtain last digit  $d = num \% 10$

    Add  $d$  to  $sum$

$num = num / 10$

Print  $sum$

C:\Users\Dell\Documents\class work-26.c - [Executing] - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

Project Classes Debug [\*]structure.c LCM and HCF.c class work-1.c class work-16.cpp class work-17.c class work-20.c class work-21.c class work-25.c class work-26.c

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int num, sum = 0, digit;
6
7     printf("Enter a number: ");
8     scanf("%d", &num);
9
10    while (num != 0)
11    {
12        digit = num % 10;
13        sum += digit;
14        num /= 10;
15    }
16
17    printf("Sum of digits: %d", sum);
18
19    return 0;
20
21 }
```

C:\Users\Dell\Documents\class work-26.exe

Enter a number: 5  
Sum of digits: 5  
-----  
Process exited after 8.152 seconds with return value 0  
Press any key to continue . . .

Compiler Resources Compile Log Debug Find Results

Line: 21 Col: 1 Sel: 0 Lines: 21 Length: 297 Insert Done parsing in 0.031 seconds

