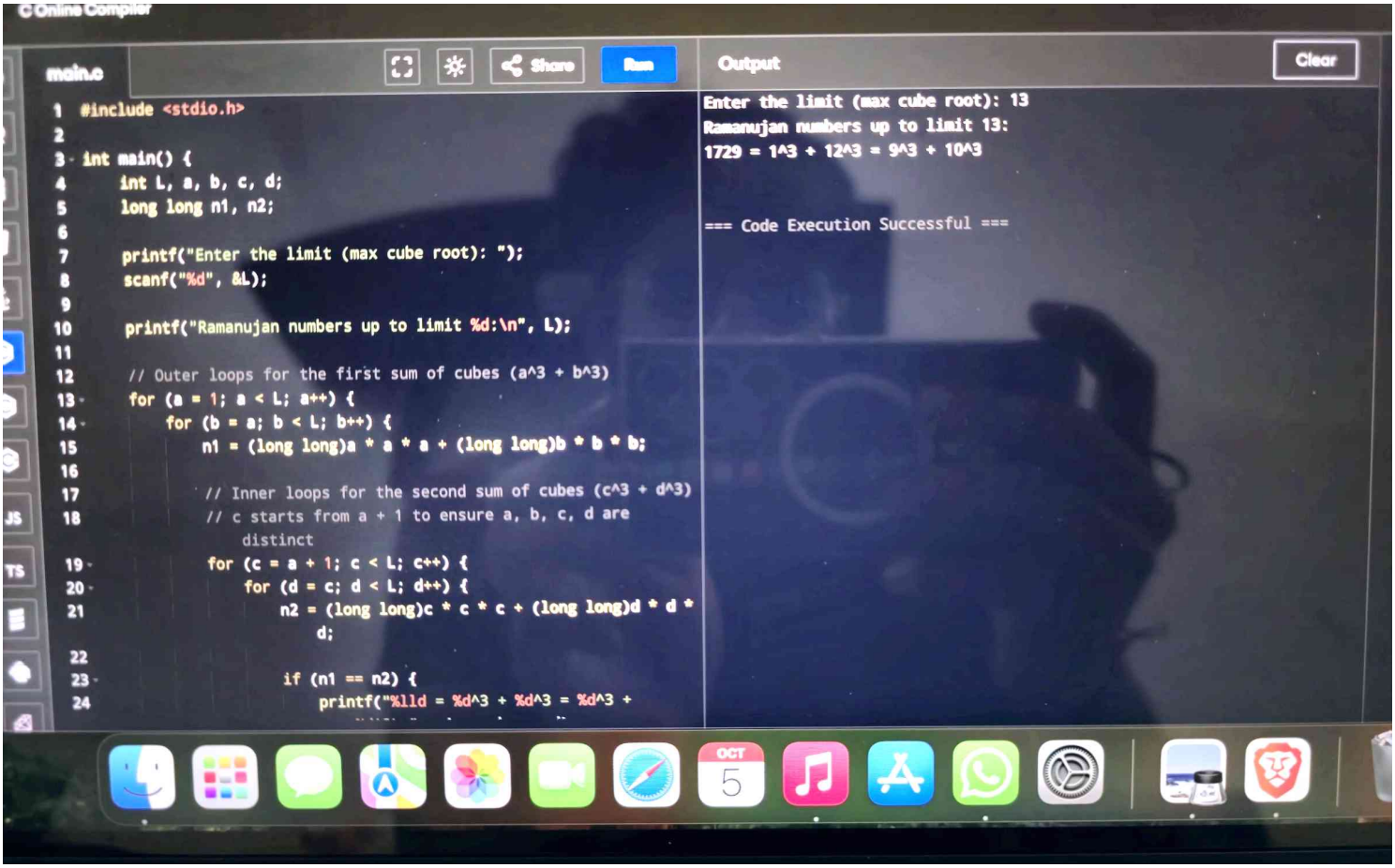
PAGE NO.:
DATE: / /

Q.5 Ramanujan Number is the smallest number that can be expressed as the sum of two cubes in two different ways. WAP to print all such numbers up to a reasonable limit

```
#include <stdio.h>
int main() {
int L, a, b, c, d;
int n1, n2;

Printf("Enter the limit (max cube root); ");
Scanf("%d", &L);

Printf("Ramanujan Numbers up to limit %d: \n", L);

for (a=1; a<=L; a++) {
  for (b=a; b<=L; b++) {
    n1 = a*a*a + b*b*b;

for (c=a+1; c<=L; c++) {
  for (d=c; d<=L; d++) {
    n2 = c*c*c + d*d*d;

    if (n1 == n2) {
    printf("%d = %d^3 + %d^3 = %d^3
        + %d^3\n", n1, a, b, c, d);
Remarks:  } return 0;                    Teacher's Signature
```


C Online Compiler
main.c
Share
Run
Output
Clear
#include <stdio.h>
int main() {
int L, a, b, c, d;
long long n1, n2;
printf("Enter the limit (max cube root): ");
scanf("%d", &L);
printf("Ramanujan numbers up to limit %d:\n", L);
// Outer loops for the first sum of cubes (a^3 + b^3)
for (a = 1; a < L; a++) {
for (b = a; b < L; b++) {
n1 = (long long)a * a * a + (long long)b * b * b;
// Inner loops for the second sum of cubes (c^3 + d^3)
// c starts from a + 1 to ensure a, b, c, d are distinct
for (c = a + 1; c < L; c++) {
for (d = c; d < L; d++) {
n2 = (long long)c * c * c + (long long)d * d * d;
if (n1 == n2) {
printf("%lld = %d^3 + %d^3 = %d^3 +
Enter the limit (max cube root): 13
Ramanujan numbers up to limit 13:
1729 = 1^3 + 12^3 = 9^3 + 10^3
=== Code Execution Successful ===