Description:

C program that uses the Miller–Rabin test to check if a number is probably prime and prints the CPU time.

Source Code:

```c
#include <stdio.h>

#include <stdlib.h>

#include<time.h>

// Function to perform modular exponentiation

long long power_mod(long long base, long long exp, long long mod) {

long long result = 1;

base = base % mod;

while (exp > 0) {

if (exp % 2 == 1) {

result = (result * base) % mod;

}

exp = exp >> 1;

base = (base * base) % mod;

}

return result;

}


// Miller-Rabin test for a single iteration

int miller_rabin_test(long long n, long long d) {

long long a = 2 + rand() % (n - 4); // Random base in range [2, n-2]

long long x = power_mod(a, d, n);
```

```c
        if (x == 1 || x == n - 1) {
            return 1; // Probably prime
        }

        while (d != n - 1) {
            x = (x * x) % n;
            d *= 2;

            if (x == 1) return 0; // Composite
            if (x == n - 1) return 1; // Probably prime
        }

        return 0; // Composite
    }

// Miller-Rabin Primality Test
int is_prime(long long n, int k) {
    if (n <= 1 || n == 4) return 0; // Not prime
    if (n <= 3) return 1; // Prime

    long long d = n - 1;
    while (d % 2 == 0) {
        d /= 2;
    }
```

```c
    int i;

    for (i = 0; i < k; i++) {

        if (!miller_rabin_test(n, d)) {

            return 0; // Composite


        }
    }


    return 1; // Probably prime

}


int main() {

    long long n;

    clock_t start,end;

    double cpu_time_used;

    start=clock();

    int k = 5; // Number of iterations for accuracy


    printf("Enter a number to check if it is prime: ");

    scanf("%lld", &n);


    if (is_prime(n, k)) {

        printf("%lld is a prime number.\n", n);

    } else {

        printf("%lld is not a prime number.\n", n);

    }
```

```c
end=clock();

cpu_time_used=((double)(end-start))/CLOCKS_PER_SEC;

printf("Execution %2f seconds",cpu_time_used);

return 0;

}
```

Output:

```
Enter a number to check if it is prime: 4
4 is not a prime number.
Execution time:0.000000 second
```