Write a program to solve the following problems related to algorithm analysis:

1. Given input sizes \( n_0 \) and \( n_1 \), \( T(n_0) \), the running time for and input of size \( n_0 \), and the big-O running time of the method, compute and print \( T(n_1) \), the running time for and input of size \( n_1 \).

2. Given input size \( n_0 \), its running time \( T(n_0) \), and the the big-O running time of the method, compute and print the size \( n_1 \) of the largest problem that can be solved in a given a maximum allowed time \( T(n_1) \) (remember that \( n_1 \) has to be an integer).

3. Given input sizes \( n_0 \), and \( n_1 \) and their respective running times \( T(n_0) \) and \( T(n_1) \), determine the big-O running time of the method.

4. Given the constants \( a, b, \) and \( k \) describing the running time of a recursive method, find and output the big-O running time of the method

Here's a simulated run of your program:

Please select:
1) Given \( n_0 \), \( n_1 \), \( T(n_0) \), and \( f(n) \), compute \( T(n_1) \)
2) Given \( n_0 \), \( T(n_0) \), \( T(n_1) \), and \( f(n) \), find \( n_1 \)
3) Given \( n_0 \), \( n_1 \), \( T(n_0) \), and \( T(n_1) \), find \( f(n) \)
4) Given \( a, b, \) and \( k \), find \( f(n) \)
5) Quit the program

Selection: 1
Enter the following parameters:
\( n_0 = 100 \)
\( n_1 = 500 \)
\( T(n_0) \) (in seconds) = 0.0001
Select \( f(n) \):
   a) \( \log n \)
   b) \( n \)
   c) \( n \log n \)
   d) \( n^2 \)
   e) \( n^3 \)
   f) \( 2^n \)

Selection: e
Result: If a \( O(n^3) \) method takes 0.0001 seconds to run for an input of size 100, it will take 0.0125 seconds to run for an input of size 500.

Please select:
1) Given \( n_0 \), \( n_1 \), \( T(n_0) \), and \( f(n) \), compute \( T(n_1) \)
2) Given \( n_0 \), \( T(n_0) \), \( T(n_1) \), and \( f(n) \), find \( n_1 \)
3) Given \( n_0 \), \( n_1 \), \( T(n_0) \), and \( T(n_1) \), find \( f(n) \)
4) Given \( a, b, \) and \( k \), find \( f(n) \)
5) Quit the program
Selection: 2
Enter the following parameters:
n0 = 1000
T(n0) (in seconds) = 0.0001
T(n1) (in seconds) = 0.0002
Select f(n):
a) log n
b) n
c) n log n
d) n^2
e) n^3
f) 2^n
Selection: a
Result: If a $O(\log n)$ method takes 0.0001 seconds to run for an input of size 1000, the largest problem it can solve in 0.0002 seconds is 1000000.

Please select:
1) Given n0, n1, T(n0), and f(n), compute T(n1)
2) Given n0, T(n0), T(n1), and f(n), find n1
3) Given n0, n1, T(n0), and T(n1), find f(n)
4) Given a, b, and k, find f(n)
5) Quit the program
Selection: 3
Enter the following parameters:
n0 = 1000
n1 = 2000
T(n0) (in seconds) = 0.0001
T(n1) (in seconds) = 0.0004
Result: If a method takes 0.0001 seconds to run for an input of size 1000 and 0.0004 seconds for an input of size 2000, its running time is $O(n^2)$.

Please select:
1) Given n0, n1, T(n0), and f(n), compute T(n1)
2) Given n0, T(n0), T(n1), and f(n), find n1
3) Given n0, n1, T(n0), and T(n1), find f(n)
4) Given a, b, and k, find f(n)
5) Quit the program
Selection: 4
Enter the following parameters:
a = 4
b = 2
k = 2
A recursive program with running time described by recurrence $T(n) = 4 T(n/2) + n^2$ has running time $T(n) = O(n^2 \log n)$.

Please select:
1) Given n0, n1, T(n0), and f(n), compute T(n1)
2) Given n0, T(n0), T(n1), and f(n), find n1
3) Given n0, n1, T(n0), and T(n1), find f(n)
4) Given a, b, and k, find f(n)
5) Quit the program
Selection: 5
Bye!
Show several runs demonstrating the functionality of your program. Write a report discussing your work, as described in the syllabus.