THEORY PROBLEMS FOR DATA STRUCTURES AND ALGORITHMS (COMP1009)

1 Consider the program (valid for inputs $n \geq 1$)

```
foo(int n) {
   bar();
   if (n==1)
      return;
   foo(n-1);
   foo(n-1);
   foo(n-1);
}
```

(a) Let $C(n)$ be the number of times the function `bar()` is called when we call `foo(n)`. Write down a recurrence relation for $C(n)$. *(2 marks)*

$C(n) =$

(b) Write down the boundary condition for the recurrence relation. *(1 marks)*

$C(1) =$

(c) Using the recurrence relation to compute $C(2)$, $C(3)$ and $C(4)$. *(3 marks)*

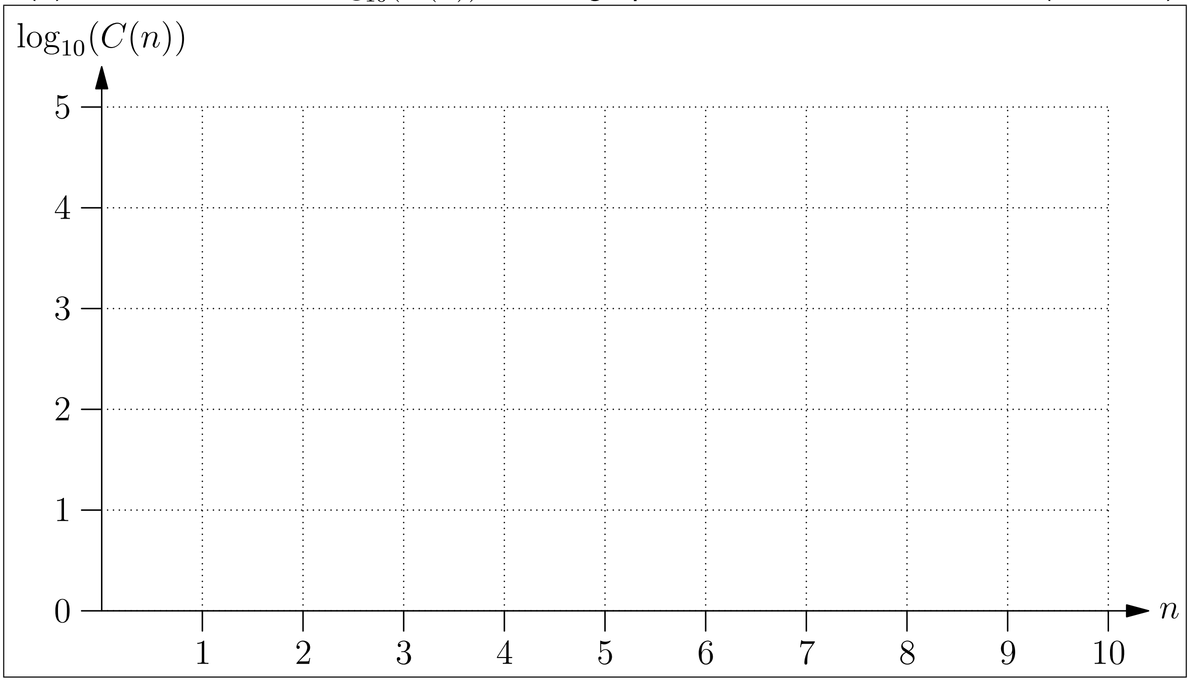
$C(2) =$

$C(3) =$

$C(4) =$

(d) Prove by induction that $f(n) = \frac{3^n - 1}{2}$ satisfies the recurrence relation for $C(n)$. *(5 marks)*

(e) Sketch the curve $\log_{10}(C(n))$ on the graph below. *(2 marks)*



(f) Assume that the most time consuming operation is calling function `bar()` then, if it takes 100s to compute `foo(5)` approximately how long will it take to compute `foo(10)`? *(2 marks)*

2 Consider the program (valid for inputs $n \geq 1$)

```
foo(int n) {
   for(i=1; i<=2n-1; i++)
      bar();
   if (n==1)
      return;
   foo(n-1);
}
```

(a) Let $C(n)$ be the number of times the function `bar()` is called when we call `foo(n)`. Write down a recurrence relation for $C(n)$. *(2 marks)*

$C(n) =$

(b) Write down the boundary condition for the recurrence relation. *(1 marks)*

$C(1) =$

(c) Using the recurrence relation to compute $C(2)$, $C(3)$ and $C(4)$. *(3 marks)*

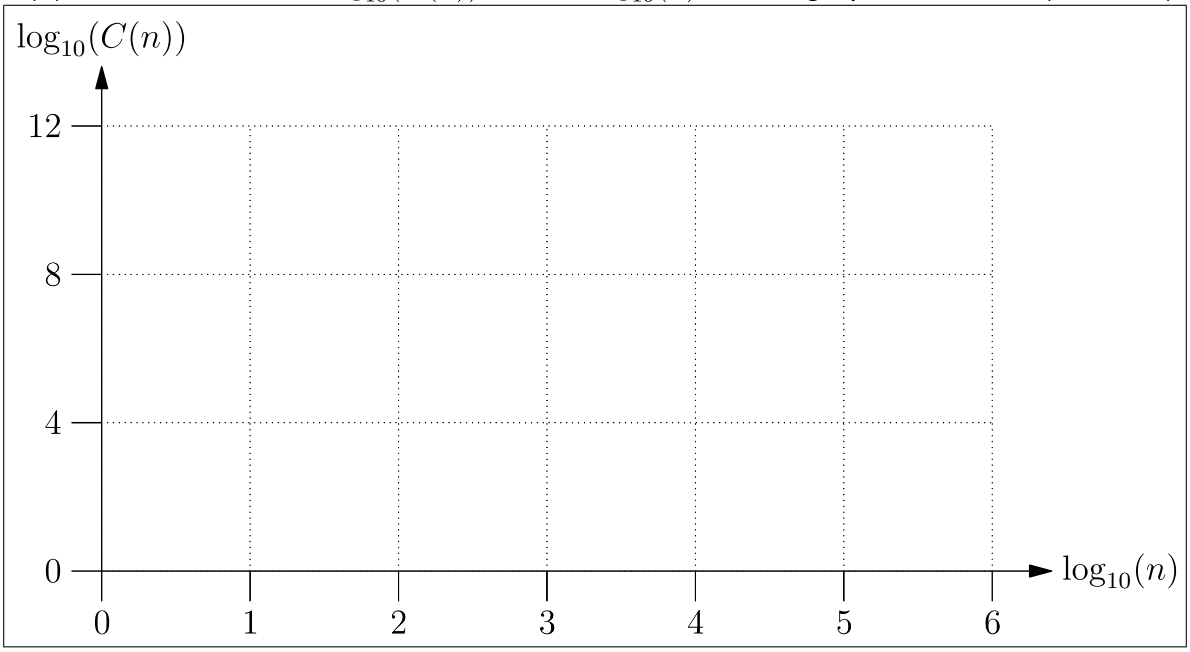
$C(2) =$

$C(3) =$

$C(4) =$

(d) Guess the solution for $C(n)$ and prove by induction that it satisfies the recurrence relation for $C(n)$. *(5 marks)*

(e) Sketch the curve $\log_{10}(C(n))$ versus $\log_{10}(n)$ on the graph below. *(2 marks)*



(f) If it takes 100s to compute `foo(1000)` approximately how long will it take to compute `foo(2000)`? *(2 marks)*

3 Consider the program (valid for inputs $n \geq 1$)

```
foo(int n) {
   bar();
   if (n==1)
      return;
   int m = (int) n/2
   foo(m);
}
```

where `(int) n/2` returns the greatest integer less than or equal to n/2 (i.e. $\lfloor n/2 \rfloor$).

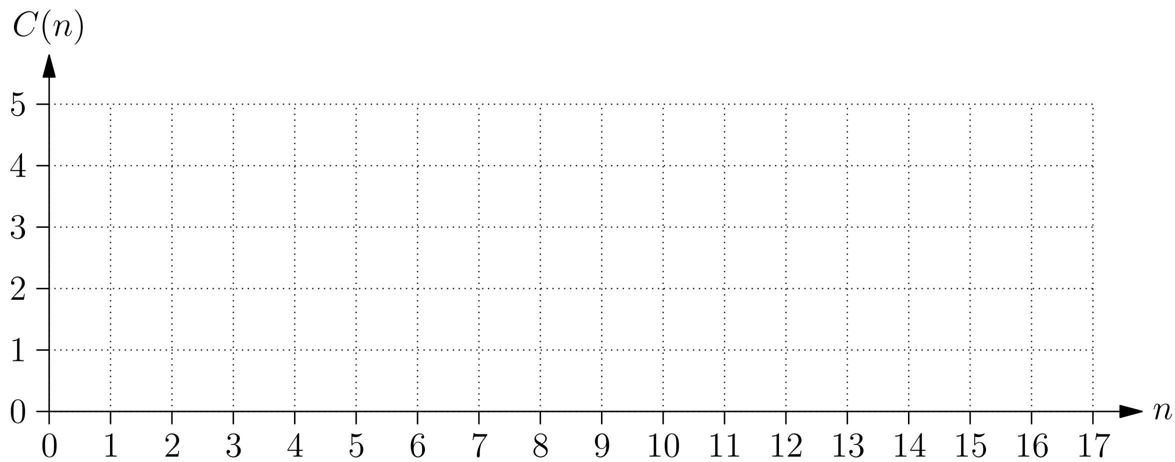
(a) Let $C(n)$ be the number of times the function `bar()` is called when we call `foo(n)`. Write down a recurrence relation for $C(n)$. *(2 marks)*

$C(n) =$

(b) Write down the boundary condition for the recurrence relation. *(1 marks)*

$C(1) =$

(c) Using the recurrence compute values of $C(n)$ to draw the graph below. *(4 marks)*



(d) Prove by induction that $f(n) = \lfloor \log_2(n) \rfloor + 1$ satisfies the recurrence relation for $C(n)$. This is simplified if we perform the inductive step over the set of integers $S_m = \{2^m, 2^m + 1, \ldots, 2^{m+1} - 1\}$. *(6 marks)*

(e) If it takes 100s to compute `foo(512)` approximately how long will it take to compute `foo(1024)`? *(2 marks)*

END OF PAPER