

Programming in C

Supervision 2

Supervisor: Joe Isaacs (josi2).

All work should be submitted in PDF form 36 hours before the supervision to the email `josi2@cam.ac.uk`

Please attempt all exam questions *without a compiler* at first, then once you are happy with the solution then use a compiler to check for errors.

1. Define a `struct` that could be used to define a graph $G = (V, E)$ with an `int` at each vertex $v \in V$ and 0 or more edges $e \in E$ to vertices $v \in V$. Try to define an interface for G in a similar fashion to the queue from last supervision.
2. Implement a depth first search (DFS) on $\llbracket G \rrbracket$ (the C interpretation of G) defined above.
3. How could you go about allocating and deleting this graph on the heap? You don't need to write a full solution only ideas.
4. What are the performance characteristics of the DFS implementation, your answer should include reference to the memory layout given a standard memory hierarchy as in lectures. How could we improve the performance?
5. <http://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2010p3q6.pdf> Data structures and pointers.
6. <http://www.cl.cam.ac.uk/teaching/exams/pastpapers/y1997p5q5.pdf> Program with errors. You need not consider the part of the question which refers to C++.
7. <http://www.cl.cam.ac.uk/teaching/exams/pastpapers/y1993p5q5.pdf>, Redefining `struct` in C.
8. When and why can pointer dereferences to an unknown memory location be expensive? How would we restructure a program to reduce the amortized cost of pointer dereferences?
9. What is the different between Arrays of Structs (AoS) and Struct of Arrays (SoA). In your own words why is one memory layout preferable to the other when pointers are involved. In what cases would you prefer AoS to SoA and vice versa?