Computer Science Tripos Part II 2002
Paper 7 Question 4
AM – Optimising Compilers

Solution Notes y2002p7.tex

(a) Generate naive 3-address code in a flowgraph assuming all variables (and temporaries) are allocated a different (virtual) register. Derive a graph (the 'clash graph') whose nodes are virtual registers. There is an edge between two virtual registers which are ever simultaneously live. Now try to colour (= give a different value for adjacent nodes) the clash graph using the real (target architecture) registers as colours. The combination is an effective method to do register allocation on machines with large-ish orthogonal register sets.

(b) First take the hint and get liveness info.

Label/which vars are live at it:

Vars:	L0	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11
r0	1	0	0	0	0	0	0	0	0	0	0	1
p	0	1	1	1	1	1	1	0	0	1	0	0
i	0	0	1	1	0	0	1	1	1	1	1	0
t3	0	0	0	0	1	0	0	0	0	0	0	0
t4	0	0	0	0	0	1	0	0	0	0	0	0
t5	0	0	0	0	0	0	0	1	0	0	0	0
t6	0	0	0	0	0	0	0	0	1	1	0	0

Then draw a graph:

```
                 L2,L3,L6,L9
             p ----------------- i                   r0
            /|\___L9            / \
       L4 /  \      \__   L8,L9 /   \ L7
         /  L5\       \___   /      \
       t3      t4           t6        t5
```

Incidentally, this colours with three registers.

(c) Ah, well, since converting to SSA form merely changes a uses of a single variable into uses of many variables connected by MOV or ϕ-functions, then any colouring of the original form can be converted into a colouring of the SSA form, merely by colouring all the SSA variants of a variable with the same colour. But perhaps one can do better (example in notes). Thus one would *expect* $l \leq k$, and indeed this would happen if register colouring were perfect and k and l were the respective chromaticities. But colouring is-NP complete, so that real colourers are approximations, and thus one might find a program whose SSA form colouring eagerly made bad decisions which the original did not make, hence possibly $l > k$, but this should be rare for a good compiler.