Traditional assembly language:

```
...
rlwinm r27,r21,11,0x7f8
lwzx r27,r7,r27
rlwinm r31,r29,0,0xff000000
xor r29,r20,r25
rlwimi r31,r30,0,0xff0000
...
```

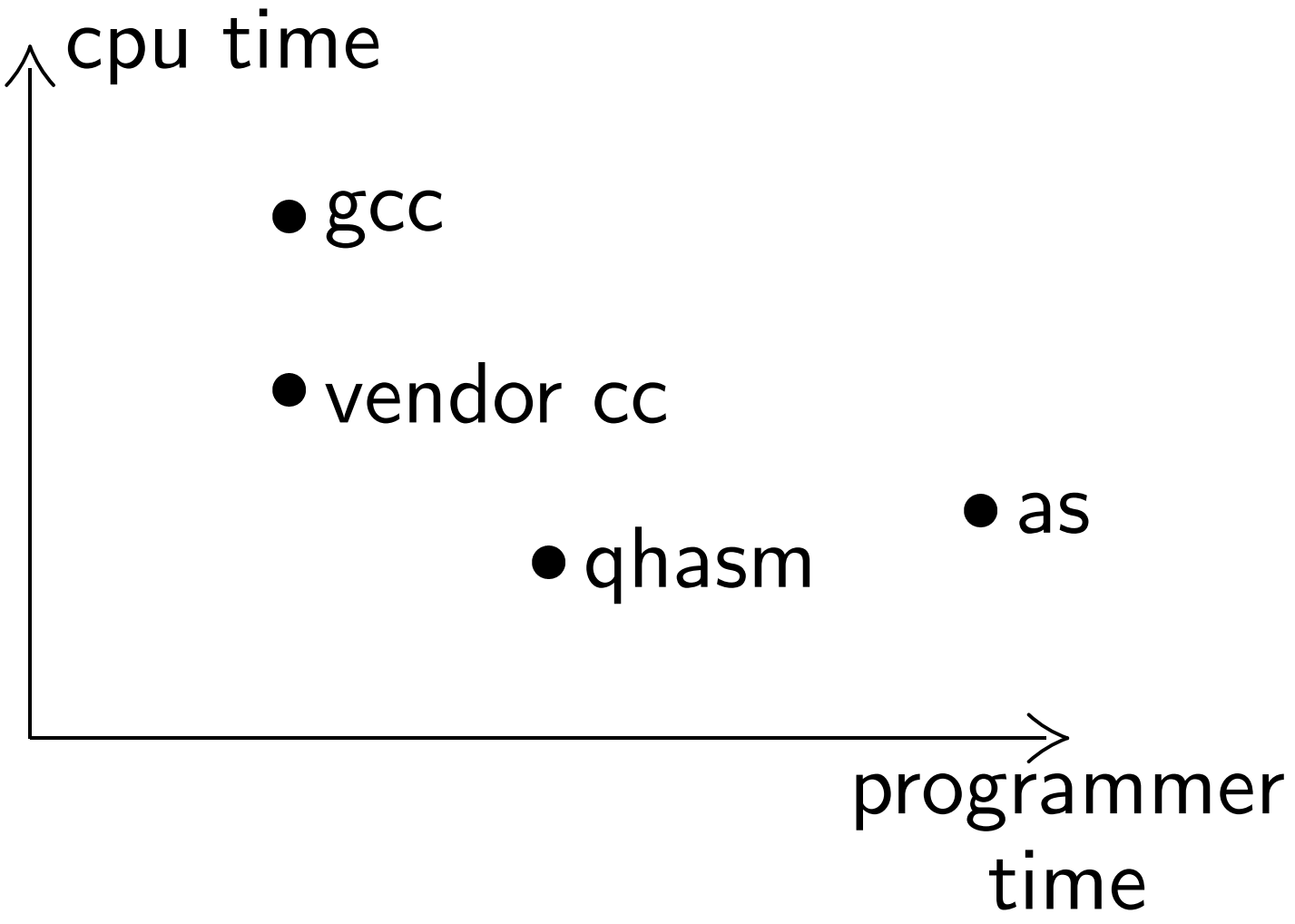
The qhasm assembly language:

```
...
q3 = 0x7f8 & (x3 <<< 11)
q3 = *(uint32 *)(tab1 + q3)
f = 0xff000000 & q0
y3 = z3 ^ p30
f bits 0xff0000 = q1
...
```

Fifth instruction in C:

```
f =
(f&~0xff0000)|(q1&0xff0000)
```

User cooperates with qhasm tools
to easily achieve
excellent register allocation,
excellent instruction scheduling,
automated range verification, etc.



Have any challenges for qhasm?

"This C function is time-critical:

...

Can you make it faster using qhasm?"

Send to the qhasm mailing list:
`http://cr.yp.to/qhasm.html`