

Submit your program and your output as email attachments.

Write a complete 32-bit Masm program to do the following arithmetic. Declare all variables as dword. Print EXACTLY where the C++ code says to print. Don't print earlier or later. Values printed in a single line to cout should be printed on the same line in the assembler output. The comments I have provided are important guidelines to good assembler programming style. If something is in a register, reuse it (as long as the reuse is close). Put spaces between messages and values. Do NOT change anything the C++ does not change.

```
int num = 51, val = 35, ans, res, ans2; // initialize num and val in the declaration

ans = val + 23;
res = ans + num; // ans is already in a register; use it
val += 20; // don't use registers here
num++;
cout << "ans = " << ans <<
    " res = " << res <<
    " val = " << val << " num = " << num << endl;

num = -12; // assign this value in the code; no register
val = -72 - num;
res = -ans;
cout << "num = " << num << " val = " << val <<
    " res = " << res << " ans = " << ans << endl;

res = -num - val;
ans = - (num + val); // follow precedence rules
ans2 = (86 - num) - (-val - 15); // follow precedence rules; use 2 registers
cout << "res = " << res << " ans = " << ans <<
    " ans2 = " << ans2 << " num = " << num << endl;

// use arithmetic shifts here to do the computations
num = 10;
num = num * 4; // you must change num; don't use a register
cout << "num = " << num << endl;

num = -39;
num = num / 2; // you must change num; don't use a register
cout << "num = " << num << endl;

val = 28;
ans = val * 8; // you may NOT change val; use a copy in a register
num = val / 4; // you may NOT change val; use a copy in a register
cout << "ans = " << ans
    << " num = " << num << endl;
```