Jonathan W. Valvano

(25) Question 1. Design an analog amplifier with the following relationship

\[ V_{out} = 100 \cdot (V_1 - V_2) + 2.5 \]

Step one, rewrite with reference chip voltage shown as a third input.

\[ V_{out} = 100 \cdot V_1 - 100 \cdot V_2 + 0.5 \cdot V_{ref} \]

Step two, add a ground as a third input, with a gain such that the sum of the gains is 1.

\[ V_{out} = 100 \cdot V_1 - 100 \cdot V_2 + 0.5 \cdot V_{ref} + 0.5 \cdot V_g \]

Step three, choose a feedback resistor which is the least common multiple of 100, 0.5. \( R_f = 100k \Omega \).

Step four, select four input resistors to get the desired gains.

--Diagram--

(25) Question 2.

Sema4Type InValid; // 1 means In1, In2 are valid, 0 means not valid
Sema4Type OutValid; // 1 means Out is valid, 0 means not valid

```c
void client(void){
  OS_InitSemaphore(&InValid,0);
  OS_InitSemaphore(&OutValid,0);
  DDRA=DDRB=0; // Ports A, B are input
  DDRC=0xFF;   // Port C is an output
  while(1){
    In1=PORTA; // read first input
    In2=PORTB; // read second input
    OS_Signal(&InValid);
    OS_Wait(&OutValid);
    PORTC=Out; // output the result
  }
}
void server(void){
  while(1){
    OS_Wait(&InValid);
    if(In1>=In2){
      Out = In1; // In1 is larger
    } else{
      Out = In2; // In2 is larger
    }
    OS_Signal(&OutValid);
  }
}
```

(25) Question 3. The error due to the finite time between outputs is 100V/s*0.001s = 0.1V

The error due to the finite DAC precision is 5V/1024 = 0.005 V.

Combined the error is 0.105V.

(25) Question 4. Convert constants to integers

\[ T = \frac{83 \cdot R \cdot R}{10000} - \frac{12465 \cdot R}{10000} + 6632 \]

Use signed long math because \( 83 \cdot R \cdot R \) will exceed 16 bits. Must be signed because \( 83 \cdot R \cdot R < 124650 \cdot R \)

```c
short Convert(short R){
  long bigT,bigR;
  bigR = (long)R;      // promote to 32-bits
  bigT = (83L*bigR*bigR–124650L*bigR)/10000+6632;
  return (short) bigT; // demote
}
```

You could use unsigned long math if you rearrange the terms to keep the intermediate calculations positive.

```c
unsigned short Convert(unsigned short R){
  unsigned long bigT,bigR;
  bigR = (unsigned long)R; // promote to 32-bits
  bigT = (66320000L+83L*bigR*bigR–124650L*bigR)/10000;
  return (unsigned short) bigT; // demote
}
```