Burt Rosenberg

Problem Set 1

OUT: 18 SEPTEMBER, 1992
DUE: 25 SEPTEMBER, 1992

1. Assume that a and b are positive integers and $a > b$. Complete the proof that the Euclidean Algorithm is correct by showing that:

$$\gcd(a, b) = \gcd(a - b, b).$$

   Here are some hints for the solution. First show that any positive integer c which divides both a and b will divide $a - b$. Show that for this reason, $\gcd(a, b) \leq \gcd(a - b, b)$. Give a similar reason for $\gcd(a - b, b) \leq \gcd(a, b)$. Now complete the proof.

2. Write and debug a program on VAX/VMS that adds two fractions and then uses the Euclidean Algorithm to put the result in lowest terms. In order to get full credit, your solution should exhibit excellent program structure. You should build up your solution from the subroutines,

```
procedure swap( var i,j : integer ) ;
function gcd( i, j : integer ) : integer ;
```

   presented in class. You should also isolate in procedures the reading and writing of fractions and their addition. To be precise, write subroutines,

```
function read_fraction : FractionType ;
procedure write_fraction( frac : FractionType ) ;
function add_fraction( frac1,
    frac2 : FractionType ) : FractionType ;
```

   where FractionType is defined as a record of two integer elements, num and denom.

3. Revise the program to make it more efficient in the following manner. Note that you might often apply the rule $\gcd(a, b) = \gcd(a-b, b)$ several times in succession. Prove that

$$\gcd(a, b) = \gcd(a - kb, b)$$

   where k is a positive integer and $a - kb > 0$. Review the **mod** operator in Pascal and rewrite the procedure gcd keeping these remarks in mind.