

Program 1

```
{x = A ∧ y = B ∧ A ≥ B}
while y ≠ 0 do
  begin
    x := x - 1;
    y := y - 1
  end
{x = A - B}
```

Program 2

```
{x = A ∧ y = B}  
begin  
z := 1;  
while y ≠ 0 do  
  begin  
    z := z * x;  
    y := y - 1  
  end  
end  
{z = AB}
```

Program 3

```
{x = A ∧ y = B}
begin
q := 0;
r := x;
while r ≥ y do
  begin
    q := q + 1;
    r := r - y
  end
end
{(q · y + r = A) ∧ (r < y)}
```

Program 4

```
{x = A}
begin
z := 1;
while x ≠ 0 do
  begin
    z := z * x;
    x := x - 1
  end
end
{z = A!} (z is "A factorial.")
```

Program 5

```
{x = A ∧ y = B}
begin
z := 1;
while y ≠ 0 do
  if even?(y)
    then begin y := y/2; x := x * x end
    else begin y := y - 1; z := z * x end
  end
end
{z = AB}
```

Program 6

```
{x = A ∧ y = B}
begin
z := 1;
while y ≠ 0 do
  begin
    while even?(y) do
      begin
        y := y/2
        x := x * x
      end ;
    z := z * x;
    y := y - 1
  end
end
{z = AB}
```

Program 7

```
{x = A ∧ y = B}  
begin  
while x ≠ y do  
  if x < y  
    then y := y - x  
    else x := x - y  
  end  
end  
{x = gcd(A, B)}
```

Program 8

Program taken from, Saud Alagic' and Michael A. Arbib, The Design of Well-structured and Correct Programs, Springer, New York, 1978, (Ch. 2, Sec. 2.7, p 44)

```
begin
{ $x > 0 \wedge y > 0$ }
 $r := x$ ;
 $q := 0$ ;
 $w := y$ ;
while  $w \leq r$  do
   $w := 2w$ ;
while  $w \neq y$  do
  begin
 $q := 2q$ ;
 $w := w \div 2$ ;
if  $w \leq r$  then
  begin
 $r := r - w$ ;
 $q := 1 + q$ ;
  end
  else
 $r := r$ ; [i.e., do nothing]
  end
end
{  $(x = qy + r) \wedge (0 \leq r < y)$  }
end
```