1. Value Numbering

(a) What are in the CODE, SYMBOLS, AVAIL, and CONSTANTS tables?

<table>
<thead>
<tr>
<th>CODE Instructions</th>
<th>CODE resultValNum</th>
<th>CODE isConst</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) i + 1</td>
<td>3</td>
<td>no</td>
</tr>
<tr>
<td>(2) a := (1)</td>
<td>3</td>
<td>no</td>
</tr>
<tr>
<td>(3) a + b</td>
<td>5</td>
<td>no</td>
</tr>
<tr>
<td>(4) c := (3)</td>
<td>5</td>
<td>no</td>
</tr>
<tr>
<td>(5) d := 1</td>
<td>2</td>
<td>yes deleted</td>
</tr>
<tr>
<td>(6) i + d</td>
<td>3</td>
<td>no</td>
</tr>
<tr>
<td>(7) e := (6)</td>
<td>3</td>
<td>no</td>
</tr>
<tr>
<td>(8) d + 2</td>
<td>7</td>
<td>yes deleted</td>
</tr>
<tr>
<td>(9) f := (8)</td>
<td>7</td>
<td>yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SYMBOLS name</th>
<th>SYMBOLS ValNum</th>
<th>SYMBOLS isConst</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>1</td>
<td>no</td>
</tr>
<tr>
<td>c1</td>
<td>2</td>
<td>yes</td>
</tr>
<tr>
<td>a</td>
<td>3</td>
<td>no</td>
</tr>
<tr>
<td>b</td>
<td>4</td>
<td>no</td>
</tr>
<tr>
<td>c</td>
<td>5</td>
<td>no</td>
</tr>
<tr>
<td>d</td>
<td>2</td>
<td>yes</td>
</tr>
<tr>
<td>e</td>
<td>3</td>
<td>no</td>
</tr>
<tr>
<td>c2</td>
<td>6</td>
<td>yes</td>
</tr>
<tr>
<td>f</td>
<td>7</td>
<td>yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CONSTANTS ValNum</th>
<th>CONSTANTS bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AVAIL lhsValNum</th>
<th>AVAIL op</th>
<th>AVAIL rhsValNum</th>
<th>AVAIL resultValNum</th>
<th>AVAIL instr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+</td>
<td>2</td>
<td>3</td>
<td>(1)</td>
</tr>
<tr>
<td>3</td>
<td>+</td>
<td>4</td>
<td>5</td>
<td>(3)</td>
</tr>
</tbody>
</table>

(b) What is the output code?

```
(1) i + 1                        // a := i + 1
(2) a := (1)                     // a := (1)
(3) a + b                        // c := a + b
(4) c := (3)                     // c := (3)
(5) d := 1                       // d := 1
(6) e := (1)                     // e := (1)
(7) f := 3                       // f := 3
```
2. **Static Single Assignment.** Consider the following program

(a) Construct a control flow graph for the program.

```
a = 6
b = 12
c = 14
i = a*b
j = 9
read n
if (i > c) then
    k = 0
else
    i = c/2
if (c < 0) then
    c = -c
k = 0
while (k < n) do
    i = i + (i & (i << k))
k = k + 1
if (j == 9) then
    return (b + i)
else
    i = i/2
b = b + 2
return (b + i)
```
(b) Construct a dominator tree for the resulting CFG.

```
1
  /|
 / |\     |
2  3  4  5  6
  |   |   |
  7  8  9
  /   |  |
10   11
```

(c) What is the iterated dominance frontier for variables c and i?

\[ DF(c) = DF(1,5) = \{ 6 \} \]
\[ DF(i) = DF(1,2,3,8) = \{ 4,7 \} \]
(d) Generate the SSA form for this program.

```
procedure foo
    a, b, c, i, j, k, n : integer

    a = 6
    b1 = 12
    c1 = 14
    i1 = a * b1
    j = 9
    read n

    if (i1 > c1) then
        i2 = i1/2
    else
        i3 = c1/2

    i4 = phi(i2,i3)
    if (c1 < 0) then
        c2 = -c1

    c3 = phi(c1,c2)
    k1 = 0

    L: i5 = phi(i4,i6)
    k2 = phi(k1,k3)
    if (k2 < n) then
        i6 = i5 + (i5 & (1 << k2))
        k3 = k2 + 1
        goto L

    if (j == 9) then
        b2 = b1 + a

    b3 = phi(b1,b2)
    return (b3 + i5)
end foo
```
(e) Show the result of conditional constant propagation.

procedure foo
  a, b, c, i, j, k, n : integer

  a = 6
  b = 12
  c = 14
  i = 72       // a * b
  j = 9
  read n

  if (true) then  // 72 > 14
    i = 36       // 72 / 2
  else
    i = c/2
  fi

  if (false) then  // 14 < 0
    c = -c
  fi

  k = 0
  while (k < n) do
    i = i + (i & (1 << k))
    k = k + 1
  end while

  if (true) then  // 9 == 9
    b = 18       // 12 + 6
  fi

  return (18 + i)  // b + i
end foo