Assignment Series 5

Code Transformation and Optimisation

**Assignment 18: Static Single Assignment Form**

i\_0 = 0;  
p\_0 = i\_0 < n;  
  
while ( phi(p\_0, p\_1) )  
{

j\_0 = 0;  
 q\_0 = j\_0 < m;  
  
 while( phi(q\_0, q\_1) )  
 {

if (i\_0 < j\_0)  
 val\_0 = phi( val\_0, val\_1, val\_3, val\_5 );   
 val\_1 = val\_0 + i\_0;

else if (j\_0 == i\_0)  
 val\_2 = phi( val\_0, val\_1, val\_3, val\_5 );  
 val\_3 = val\_2 + 1;

else  
 val\_4 = phi( val\_0, val\_1, val\_3, val\_5 );  
 val\_5 = val\_4 + j\_0;

j\_1 = j\_0 + 1;  
 q\_1 = j\_1 < m;

}  
 i\_1 = i\_0 + 1;  
 p\_1 = i\_1 < n;

}

**Assignment 19: Machine-Independent Optimisation**

Original:

i = 0;  
while (i < n)  
{  
 j = 0;  
 while(j < m)  
 {  
 if (i < j)  
 val = val + i;

else if (j == i)  
 val = val - 1

else  
 val = val + j;

j = j + 1;  
 }  
 i = i + 1;  
}

After loop-unswitching:  
  
i = 0;  
while (i < n ) {  
 j = 0;

if (i < j ) {  
 while (j < m ) {  
 val = val + i;  
 j = j + 1;  
 }  
 }  
   
 if ( j == i ) {  
 while (j < m) {  
 val = val - 1;  
 j = j + 1;  
 }  
 }  
  
 if (i > j) {  
 while (j < m) {  
 val = val + j;  
 j = j + 1;  
 }  
 }  
  
 i = i + 1;  
}

**Assignment 20: Compilation Schemes**

…