CSC 766 Code Optimization for Scalar and Parallel Programs
Spring 2023 Xipeng Shen HW 2

1

Convert the following loop to a form where the loop indexes are each incremented by 1:

```
for(i=50;i>=10;i=i-7)
    X[i,i+1]=0;
```

```
for(i=0;i<=5;i=i+1)
    X[i*7+15,i*7+16]=0;
```

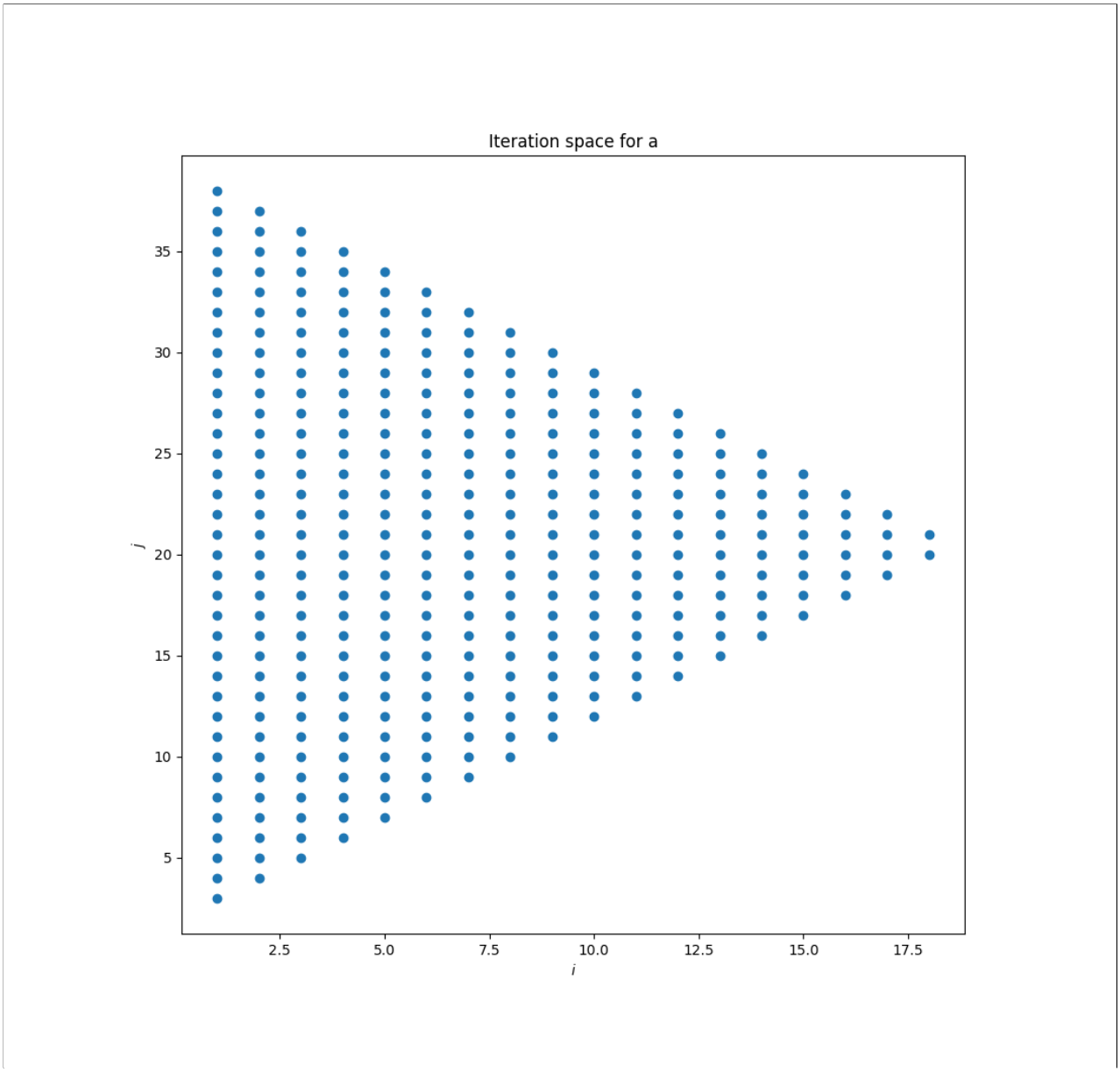
2

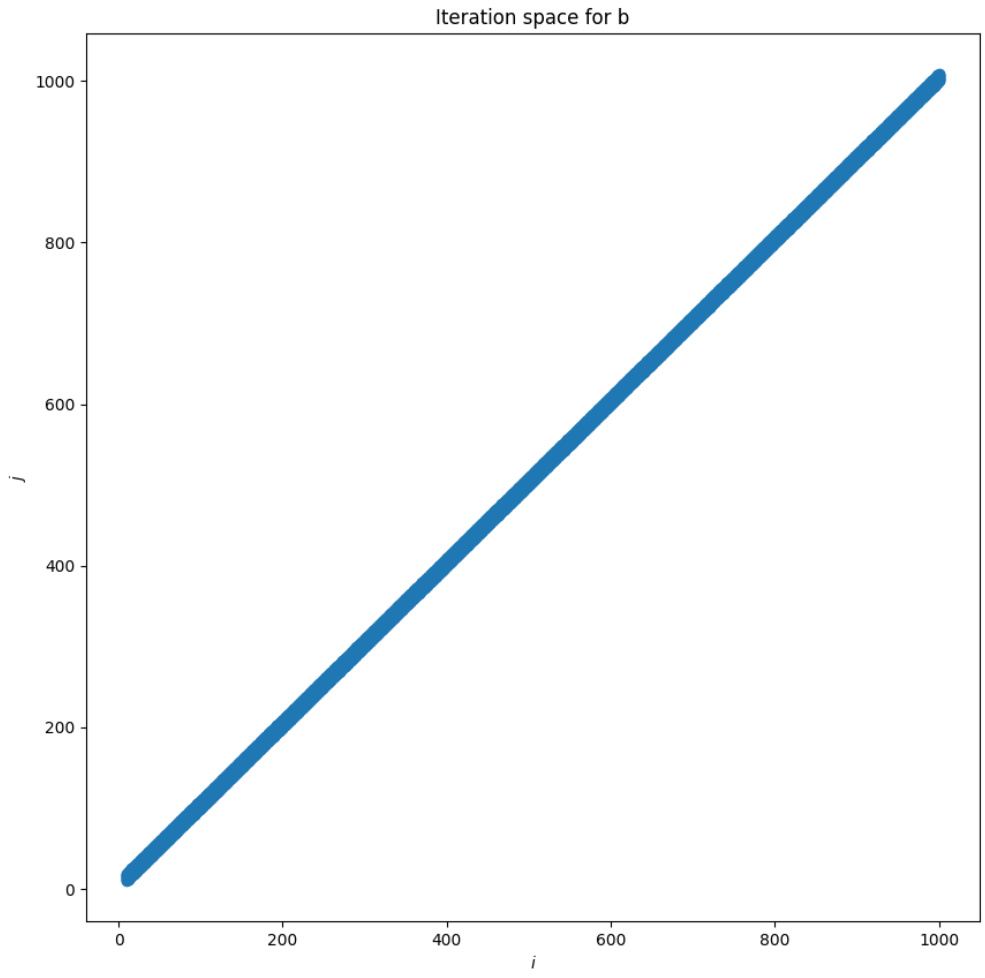
a)
```
for (i=1; i<30; i++)
    for (j=i+2; j<40-i; j++)
        X[i,j]=0;
```

b)
```
for (i=10; i<=1000;i++)
    for (j=i; j<i+10; j++)
        X[i,j]=0;
```

c)
```
for (i=1; i<100; i++)
    for (j=0; j<100+i; j++)
        for (k=i+j; k<100-i-j; k++)
            X[i,j,k]=0;
```

1. Draw the iteration spaces for (a) and (b).


Iteration space for a
35
30
25
20
15
10
5
j
2.5
5.0
7.5
10.0
12.5
15.0
17.5
i


Iteration space for b
1000
800
600
400
200
0
j
0
200
400
600
800
1000
i

2. Write the constraints in matrix form (i.e., give the values of the vectors i and b and the matrix B.)

a.

$$\begin{pmatrix} 1 & 0 \\ -1 & 0 \\ -1 & 1 \\ -1 & -1 \end{pmatrix} \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} -1 \\ -29 \\ -2 \\ 39 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

b.

$$\begin{pmatrix} 1 & 0 \\ -1 & 0 \\ -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} -10 \\ 1000 \\ 0 \\ 9 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

c.

$$\begin{pmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & -1 & 0 \\ -1 & -1 & 1 \\ -1 & -1 & -1 \end{pmatrix} \begin{pmatrix} i \\ j \\ k \end{pmatrix} + \begin{pmatrix} -1 \\ 99 \\ 0 \\ 99 \\ 0 \\ 99 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

3. Use the Fourier-Motzkin elimination algorithm to eliminate i from each of the sets of constraints obtained in the exercise (2).

Transform constants using $m = j - i$, $j = m + i$, $i = j - m$.

a.

$$i \geq 1$$
$$i \leq 29$$
$$j \geq i + 2$$
$$j \leq 39 - i$$

Transform constants:

$$j - m \geq 1$$
$$j - m \leq 29$$
$$j \geq j - m + 2$$
$$j \leq 39 - j + m$$

$$j \geq m + 1$$
$$j \leq m + 29$$
$$m \geq 2$$
$$m \geq 2j - 39$$
$$m \leq 37$$

(from 1nd inequality $j - m \geq 1$

$$L_m = 2$$
$$U_m = 37$$
$$L_j = m + 1$$
$$U_j = m + 29$$

b.

$$i \geq 10$$
$$i \leq 1000$$
$$j \geq i$$

$$j \leq i + 9$$

Transform constants:

$$j - m \geq 10$$
$$j - m \leq 1000$$
$$j \geq j - m$$
$$j \leq j - m + 9$$

$$j \geq m + 10$$
$$j \leq m + 1000$$
$$m \geq 0$$
$$m \leq 9$$

$$L_m = 0$$
$$U_m = 9$$
$$L_j = m + 10$$
$$U_j = m + 1000$$

c.

$$i \geq 1$$
$$i \leq 99$$
$$j \geq 0$$
$$j \leq 99 + i$$
$$k \geq i + j$$
$$k \leq 99 - i - j$$

$$j \geq m + 1$$
$$j \leq m + 99$$
$$j \geq 0$$
$$m \leq 99$$
$$k \geq 2j - m$$

$$k \leq 99 - 2j + m$$

$$m \geq j - 99$$
$$m \leq j - 1$$
$$j \geq 0$$
$$m \leq 99$$
$$k \geq 2j - m$$
$$k \leq 99 - 2j + m$$

$$j \geq 0$$
$$j \leq m + 99$$
$$m \leq -1$$
$$m \leq 99$$
$$k \geq 2j - m$$
$$k \leq 99 - 2j + m$$

$$L_m = -1$$
$$U_m = 99$$
$$L_j = 0$$
$$U_j = m + 99$$
$$L_k = 2j - m$$
$$U_k = 99 - 2j + m$$

4. For each of the three loop nests, rewrite the code so the axis i is replaced by the major diagonal, i.e., use loop index variable $m = j - i$. The new axis should correspond to the outermost loop.

Transform constants using $m = j - i$, $j = m + i$, $i = j - m$.

a.

$$L_m = 2$$
$$U_m = 37$$
$$L_j = m + 1$$
$$U_j = m + 29$$

```
for (m=2; m<=37;m++)
    for (j=m+1; j<=m+29; j++)
        X[j-m,j]=0;
```

b.

$$L_m = 0$$
$$U_m = 9$$
$$L_j = m + 10$$
$$U_j = m + 1000$$

```
for (m=0; m<=9;m++)
    for (j=m+10; j<1000+m; j++)
        X[j-m,j]=0;
```

c.

$$L_m = -1$$
$$U_m = 99$$
$$L_j = 0$$
$$U_j = m + 99$$
$$L_k = 2j - m$$
$$U_k = 99 - 2j + m$$

```
for (m=-1; m<99; m++)
    for (j=0; j<m+99; j++)
        for (k=2j - m; k<99 - 2j + m; k++)
            X[j-m,j,k]=0;
```