

$$\begin{array}{l} a_i*\\ x_{i-1}+\\ b_i*\\ x_i+\\ c_i*\\ x_{i+1}=\\ F_i \end{array}$$

$$i=$$

$$\begin{array}{l} 1\vdots\\ 1\vdots\\ n=\\ 2^p=\\ 1\\ \log_2 n1 \end{array}$$

$$\begin{array}{l} x_1\\ x_3\\ x_n\\ x_1\\ x_3\\ \alpha_2^2\\ \beta_2\\ \gamma_2\\ b_1*\\ x_1+\\ c_1*\\ x_2'=\\ F_1 \end{array}$$

$$*(\alpha_2)$$

$$\begin{array}{l} a_2*\\ x_1+\\ b_2*\\ x_2+\\ c_2*\\ x_3'=\\ F_2 \end{array}$$

$$a_3* \qquad \qquad *(\beta_2)$$

$$\begin{array}{l} x_2+\\ b_3*\\ x_3+\\ c_3*\\ x_4'=\\ F_3 \end{array}$$

$$*(\gamma_2)$$

$$\begin{array}{l} x_j\\ a_i=\\ -a_{i-1}*\\ k_1\\ b_i'=\\ b_i-\\ c_{i-1}*\\ k_1-\\ a_{i+1}*\\ k_2\\ c_i=\\ -c_{i+1}*\\ k_2 \end{array}$$

$$\begin{array}{l} d_i'=\\ d_i-\\ d_{i-1}*\\ k_1-\\ d_{i+1}*\\ k_2 \end{array}$$

$$\begin{array}{l} k_1=\\ \frac{a_i}{b_{i-1}},k_2=\\ \frac{c_i}{b_{i+1}} \end{array}$$

$$\begin{array}{l} x_j'=\\ \frac{d_i'-a_i'*x_{i-1}-c_i'*x_{i+1}}{b_i'} \end{array}$$

```

(i =
0; i <
log2(size+
1)-
1; i+
+){
    (j =
2i+1-
1; j <
size; j =
j+
2i+1){
    offset =
pow(2, i);
    index1 =
j-
offset;
    k1 =
a[j]/b[index1];
    k2 =
c[j]/b[j];
    b[j] =
b[j]-
c[j-
offset]*
k1-
a[j+
offset]*
k2;
    F[j] =
F[j]-
F[j-
offset]*
k1-
F[j+
offset]*
k2;
    a[j] =
-a[j-
offset]*
k1;
    c[j] =
-c[j+
offset]*
k2;
}
}
intindex =
(size-
1)/2;
x[index] =
F[index]/b[index];
(i =
log2(size+
1)-
2; i >=
0; i-
-){
    (j =
2i+1-
1; j <
size; j =
j+
2i+1){
    offset =
2i;
    index1 =
j-
offset;
    index2 =
j+
offset;
    if(j! =
index1)
        x[index1] =
(F[index1]-
a[index1]*
x[index1-
offset]-
c[index1]*
x[index1+
offset])/b[index1];
    if(j! =
index2)
        x[index2] =
(F[index2]-
a[index2]*
x[index2-
offset]-
c[index2]*
x[index2+
offset])/b[index2];
    }
}
}

```

```

q =
blocksize
size
jq =
log2(n+
1);
P =
zeros(n, q, jq)
Q =
zeros(n, q, jq);
U =
zeros(n, q)
F =
ones(n, q);
(j =
1:
1:
2^jq-
1)
Q(j,:
)^(0) =
F(j,:
)^(0);
P(j,:
)^(0) =
0;
(k =
1:
1:
jq-
1)
(j =
1:
1:
2^jq-k-
1)
idx1 =
2^k*
j;
idx2 =
2^(k-1);
P^(k+1) =
P^(k) *
(T^(k-1)) *
P^(k)
P^(k)
P^(k)
Q^(k)
Q^(k+1) =
T^(k-1) *
(Q^(k)
Q^(k)
Q^(k)
2 *
T^(k-1) *
P^(k+1));
pointer =
2^jq-2^(jq-1);
U_pointer =
[B^-1]^(jq-1) *
Q^(jq)
Q_pointer +
Q^(jq)
Q_pointer;
(k =
jq-
1:
1:
1)
(j =
2:
1:
2^jq-k-
1)
pointer =
2^k*
j-
2^(k-1);
U_pointer =
[B^-1]^(k-1) *
(Q^(k)
Q_pointer -
T^(k-1) *

```