Large loop array的优化算法原理

int i = 0, j = 0;

```
while ( i < COUNT) {
    if (i % 10) {
        a = 0.0;
        b = 1.0;
    } else {
        a = a + 0.1;
        b = b + 0.2;
    }
    while ( j < len) {
        x[j] = a + j;
        y[j] = b + j;
        j = j + 1;
    }
    total = total + loop(x, y, len);
    i = i + 1;
}
```

内层循环j未初始化

len是最外层循环的循环不变量

```
if ( i < COUNT) {
    if (i % 10) {
        a = 0.0;
        b = 1.0;
    } else {
        a = a + 0.1;
        b = b + 0.2;
    }
    while ( j < len) {
        x[j] = a + j;
        y[j] = b + j;
        j = j + 1;
    }
    total = total + loop(x, y, len);
    i = i + 1;
    while (i < COUNT) {
        if (i % 10) {
            a = 0.0;
            b = 1.0;
        } else {
            a = a + 0.1;
            b = b + 0.2;
        }
        total = total + loop(x, y, len);
        i = i + 1;
    }
}
```

激进死代码消除
指针分析
循环不变量外提
稀疏条件常量传播
冗余消除

```
if ( i < COUNT) {
    a = a + 0.1;
    b = b + 0.2;
    while ( j < len) {
        x[j] = a + j;
        y[j] = b + j;
        j = j + 1;
    }
    int tmp = loop(x, y, len);
    total = total + tmp;
    i = i + 1;
    while (i < COUNT) {
        total = total + tmp;
        i = i + 1;
    }
}
```