

```

// BtSample.cpp - BALANCED TREE SAMPLE PROGRAM
//
// MAINTENANCE HISTORY
// DATE          PROGRAMMER AND DETAILS
// 27-09-04      BYG      Original
//
//-----

#include "BtSample.h"

//-----

// GLOBAL VARIABLES

BtCust_c *custFile; // Customer file
BtPur_c *purFile; // Purchase file

//-----

// CLEAR AND RESET ALL BALANCED TREES

void
ResetAll () {
    custFile->BtReset ();
    custFile->BtClrCntr ();
    purFile->BtReset ();
    purFile->BtClrCntr ();
}

//-----

// PRINT TOTAL TIME TAKEN

void
PrintTime () {
    printf ("Total time taken: %1.1f ms\n",
        custFile->totTime + purFile->totTime);
}

//-----
//
// SAMPLE 1
//
// select custName, purVal
// from custFile, purFile
// where custId >= 100127 and
// purId = custId;
//
// Total time taken: 9253.7 ms
//
//-----

void
Sample1 () {
    Cust_t cfKey, cfRec; // Customer file search key and record
    Pur_t pfKey, pfRec; // Purchase file search key and record

    ResetAll ();

    // Load and search for customer

    cfKey.custId = 100127;
    custFile->BtFind (&cfKey);
    while (
        custFile->BtRead (&cfRec)
    ) {
        // Load and search for purchase record

        pfKey.purId = cfRec.custId;
        pfKey.purNo = MIN_PUR_NO;

        purFile->BtReset ();
        purFile->BtFind (&pfKey);
    }
}

```

```

        // Print purchase records

        while (
            purFile->BtRead (&pfRec) &&
            pfRec.purId == pfKey.purId
        ) {
            printf ("%s %1.2f\n",
                cfRec.custName, pfRec.purVal/100.00);
        }
    }

    PrintTime ();
}

//-----
//
// SAMPLE 2
//
// select custName, purVal
// from custFile, purFile
// where custId >= 100127 and
// purId = custId;
//
// Total time taken: 7683.8 ms
//
//-----

void
Sample2 () {
    Cust_t cfKey, cfRec; // Customer file search key and record
    Pur_t pfKey, pfRec; // Purchase file search key and record
    bool pfOK; // Purchase file read OK flag

    ResetAll ();

    // Load and search for customer

    cfKey.custId = 100127;
    custFile->BtFind (&cfKey);
    pfKey.purId = 100127;
    pfKey.purNo = MIN_PUR_NO;
    purFile->BtFind (&pfKey);
    pfOK = purFile->BtRead (&pfRec);
    while (custFile->BtRead (&cfRec)) {
        while (pfOK && pfRec.purId <= cfRec.custId) {
            if (pfRec.purId == cfRec.custId)
                printf ("%s %1.2f\n",
                    cfRec.custName, pfRec.purVal/100.00);
            pfOK = purFile->BtRead (&pfRec);
        }
    }
    PrintTime ();
}

//-----

// MAIN LINE

int
main () {
    srand (121);

    // Initialise customer and purchase files

    custFile = new BtCust_c ();
    purFile = new BtPur_c (custFile);
    // custFile->BtPrint ();
    // purFile->BtPrint ();

    Sample1 ();
    Sample2 ();

    return 0;
}

```