

Data Structures 2017-2018 Fall

Practice Session 2

RES. ASST. DİLARA TORUNOĞLU SELAMET

torunoglud@itu.edu.tr

03.10.2017

A solid green horizontal bar at the bottom of the slide.

Contents

Function Calls

- Call by value
- Call by reference

Scope of Variables

Sparse Matrix application with Linked Lists

Function Calls

There are 2 ways for passing a parameter to a function in C++.

1. Call by Value
2. Call by Reference

Function Calls

```
void duplicateByValue(int a, int b) {  
    a = a * 2;  
    b *= 2;  
}
```

```
void duplicateByReference(int &a, int &b) {  
    a = a * 2;  
    b *= 2;  
}
```

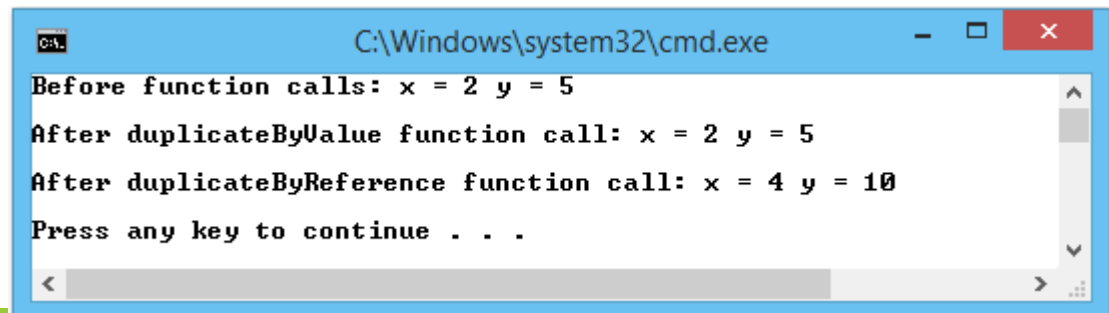
```
int main() {  
    int x = 2, y = 5;  
  
    cout << "Before function calls: x = " << x << " y = " << y << endl << endl;  
  
    duplicateByValue(x, y);  
  
    cout << "After duplicateByValue function call: x = " << x << " y = " << y << endl << endl;  
  
    duplicateByReference(x, y);  
  
    cout << "After duplicateByReference function call: x = " << x << " y = " << y << endl << endl;  
  
    return 0;  
}
```

Function Calls

```
void duplicateByValue(int a, int b) {  
    a = a * 2;  
    b *= 2;  
}
```

```
int main() {  
    int x = 2, y = 5;  
  
    cout << "Before function calls: x = " << x << " y = " << y << endl << endl;  
  
    duplicateByValue(x, y);  
  
    cout << "After duplicateByValue function call: x = " << x << " y = " << y << endl << endl;  
  
    duplicateByReference(x, y);  
  
    cout << "After duplicateByReference function call: x = " << x << " y = " << y << endl << endl;  
  
    return 0;  
}
```

```
void duplicateByReference(int &a, int &b) {  
    a = a * 2;  
    b *= 2;  
}
```



A screenshot of a Windows command prompt window titled "C:\Windows\system32\cmd.exe". The window displays the output of the C++ program. The output shows the initial values of x and y, the results after the duplicateByValue function call, and the results after the duplicateByReference function call. The window also shows a prompt for the user to press any key to continue.

```
C:\Windows\system32\cmd.exe  
Before function calls: x = 2 y = 5  
After duplicateByValue function call: x = 2 y = 5  
After duplicateByReference function call: x = 4 y = 10  
Press any key to continue . . .
```

Scope

A variable can be accessed *directly* only in the block it is defined.

Scope Resolution Operator ::

Scope Example

```
#include <iostream>

using namespace std;

int i = 100;

int main() {
    for (int i = 0; i < 5; i++) {
        cout << "Local i is: " << i << endl << endl;
        cout << "Global i is: " << ::i << endl << endl;
    }
    return 0;
}
```

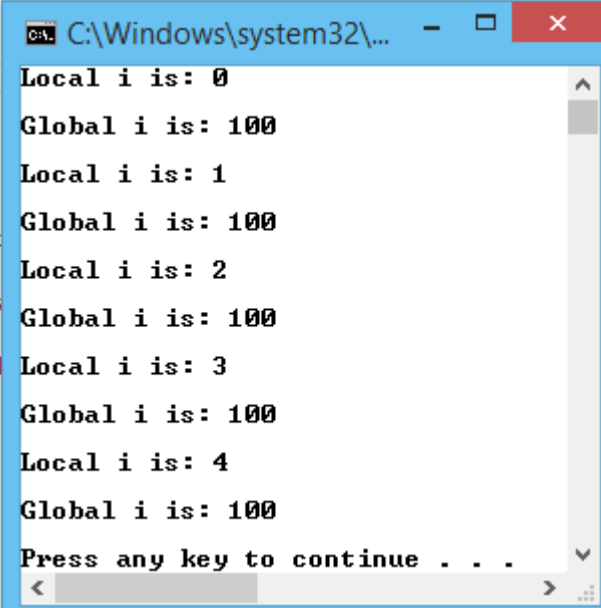
Scope Example

```
#include <iostream>

using namespace std;

int i = 100;

int main() {
    for (int i = 0; i < 5; i++) {
        cout << "Local i is: " << i << endl << endl;
        cout << "Global i is: " << ::i << endl << endl;
    }
    return 0;
}
```

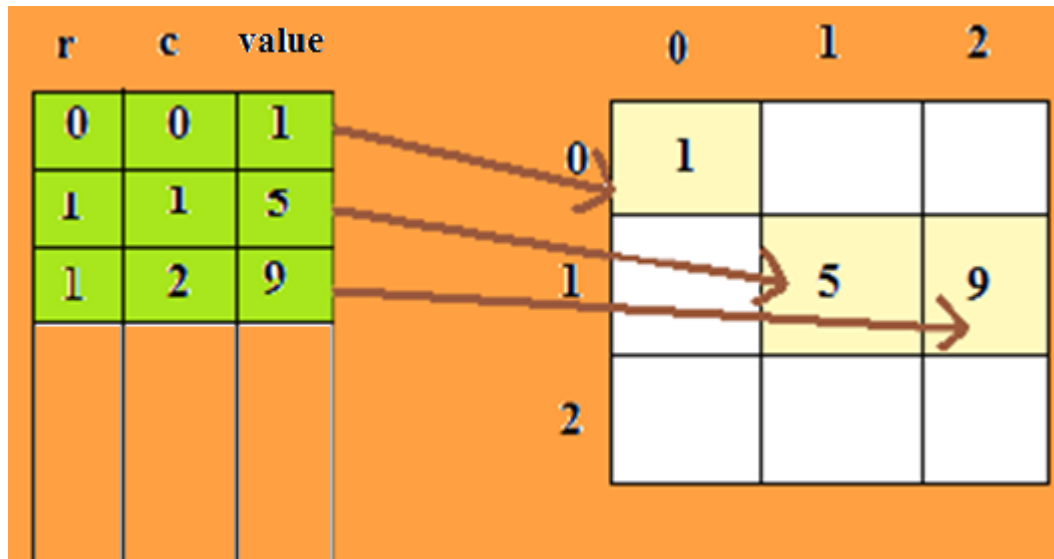


The screenshot shows a Windows command prompt window with the title bar "C:\Windows\system32\...". The output of the program is displayed as follows:

```
Local i is: 0
Global i is: 100
Local i is: 1
Global i is: 100
Local i is: 2
Global i is: 100
Local i is: 3
Global i is: 100
Local i is: 4
Global i is: 100
Press any key to continue . . .
```


Sparse Matrix Application

A **sparse matrix** is a matrix populated primarily with zeros and only a few of elements are different from zero.



Sparse Matrix Application

In this application,

- Row, column and value of non-zero elements are entered from the keyboard.
- These values are stored in a sorted linked list.
- While printing out the matrix on the screen, empty elements are printed as '0'.

```
Please enter row and column values of your sparse matrix:
4 4
Enter the row, column and value of the element you will add :
0 0 1
Will you enter any other elements to this matrix?
1
Enter the row, column and value of the element you will add :
2 3 5
Will you enter any other elements to this matrix?
1
Enter the row, column and value of the element you will add :
3 2 9
Will you enter any other elements to this matrix?
0
1 0 0 0
0 0 0 0
0 0 0 5
0 0 9 0
```

Creating a Sparse Matrix

```
struct Node{
    int row, col, value;
    Node* next;
};

struct LinkedList{
    int row_m, col_m;
    Node* head;
    void create();
    void add(int,int,int);
    void print();
};
```

```
void LinkedList::create(){
    head = NULL;
    cout << "Please enter row and coloumn values of your sparce matrix:" << endl;
    cin >> row_m >> col_m;
}
```

Creating Linked List

```
void LinkedList::add(int r, int c, int val) {  
    Node* ptr;  
    ptr = new Node;  
    ptr->row = r;  
    ptr->col = c;  
    ptr->val = val;  
    ptr->next = NULL;  
  
    if (head == NULL) {  
        head = ptr;  
    }  
  
    else {  
        Node* temp = head;  
        Node* prev;  
        if (ptr->row*row_m + ptr->col < temp->row*row_m + temp->col) {  
            ptr->next = head;  
            head = ptr;  
        }  
        else {  
            while (temp && ptr->row*row_m + ptr->col > temp->row*row_m + temp->col) {  
                prev = temp;  
                temp = temp->next;  
            }  
            if (temp) {  
                ptr->next = temp;  
                prev->next = ptr;  
            }  
            else {  
                prev->next = ptr;  
            }  
        }  
    }  
}
```

Printing out on the Screen

```
void LinkedList::print() {  
    Node *temp = head;  
    int t = 0;  
    while (temp) {  
        for (int i = t; i < temp->row*row_m + temp->col; i++) {  
            cout << "0 ";  
            if (i % (col_m) == col_m - 1)  
                cout << endl;  
        }  
        cout << temp->val << " ";  
        t = (temp->row)*row_m + temp->col + 1;  
        if ((t - 1) % col_m == col_m - 1)  
            cout << endl;  
        temp = temp->next;  
    }  
    for (int i = t; i < row_m * col_m; i++){  
        cout << "0 ";  
        if (i % col_m == col_m - 1)  
            cout << endl;  
    }  
    cout << endl;  
}
```

Main Program

```
int main()
{
    LinkedList l1;
    l1.create();
    int flag = 1;
    Node *ptr;
    while(flag)
    {
        int r,c,v;
        cout << "Enter the row, coloumn and value information of your data:" << endl;
        cin >> r >> c >> v;
        l1.add(r,c,v);
        cout << "Will you enter any other elements to this matrix?" << endl;
        cin >> flag;
    }
    l1.print();
    return 0;
}
```

Example Screenshot

```
Please enter row and column values of your sparce matrix:
4 4
Enter the row, column and value of the element you will add :
0 0 1
Will you enter any other elements to this matrix?
1
Enter the row, column and value of the element you will add :
2 3 5
Will you enter any other elements to this matrix?
1
Enter the row, column and value of the element you will add :
3 2 9
Will you enter any other elements to this matrix?
0
1 0 0 0
0 0 0 0
0 0 0 5
0 0 9 0
```