CS536: Homework 7

Keith Funkhouser

November 17th, 2015

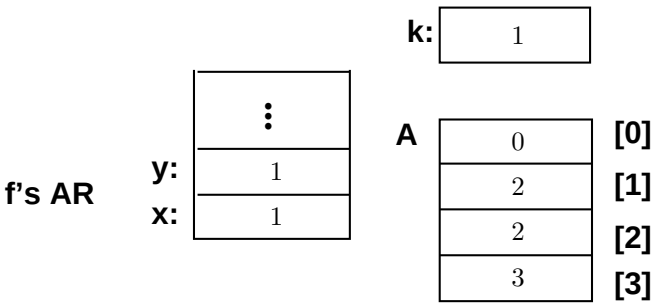
Assume we have a language similar to YES that allows arrays and that parameters in this language can be passed by value, by reference, by value-result, or by name. Consider the following program:

```
int A[4];
int k;

void f( int x, int y ) {
    x = x + 1;
    k = k + 1;
    A[k] = A[k] * 2;
    y = y + 1;
    cout << x;
    cout << y;
    cout << k;
    cout << A[k];
}

void main() {
    k = 0;
    while (k < 4) {
        A[k] = k;
        k = k + 1;
    }
    k = 0;
    f(k, A[k]);
    cout << k;
    cout << A[0];
    cout << A[1];
    cout << A[2];
    cout << A[3];
}
```

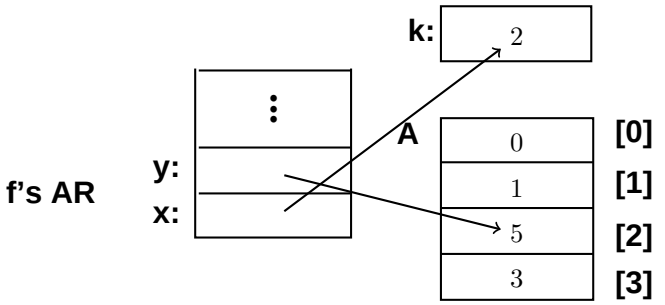
Below is a link to a file that contain six pictures. The first four pictures contain outlines of `f`'s activation record, as well as the space in the static data area for globals `k` and `A`. The last two pictures contain space for recording the output of the program.

You are to print the file and complete the pictures:

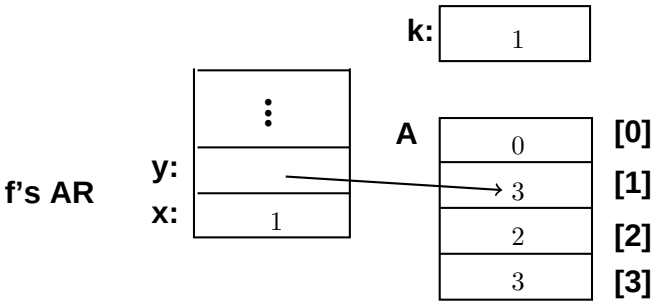
- For the first four pictures, you should fill in the values for all of the variables as they would be just before function `f` returns. For each picture, assume the parameter-passing modes for `f`'s parameters indicated in that picture.
- For the last two pictures, you should fill in the values that would be printed by functions `f` and `main`, assuming that `f`'s parameters are both passed by value-result or both passed by name (as indicated in the pictures).



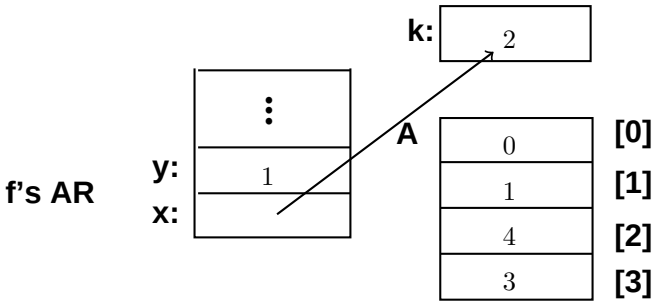
x and y passed by value



x and y passed by reference



x passed by value, y by reference



x passed by reference, y by value

printed in f

x:	1
y:	1
k:	1
A[k]:	2

printed in main

k:	1
A[0]:	0
A[1]:	1
A[2]:	2
A[3]:	3

x and y passed by value−result

printed in f

x:	2
y:	5
k:	2
A[k]:	5

printed in main

k:	2
A[0]:	0
A[1]:	1
A[2]:	5
A[3]:	3

x and y passed by name