CS150 Introduction to Computer Science I

Arrays

Last Time

- We
- Started looking at arrays and how they can be used
- Today we will
- Cover more examples of arrays, as well as investigating how to pass arrays to functions

Passing Array Elements

int i, j;
int arry1[] = {1,2,3,4,5};
int arry2[] = {5,4,3,2,1};
i = 2;
j = 3;
...
void swap (int& num1, int& num2)
{   
    int temp;
    temp = num1;
    num1 = num2;
    num2 = temp;
}

19.1: What happens?
- a) swap(i, arry1[1]);
- b) swap(arry1[2], arry1[3]);
- c) swap(arry1[i], arry2[i+1]);
- d) swap(i, arry1[1]+i);
- e) swap(arry1, arry2);

Passing arrays to functions

- Can pass individual elements
- If the whole array is passed, it is automatically passed by reference. Why do you think that is?
- The address of the array is passed to the function so that any element of the array can be accessed
- The address of an array is the memory location of the first element

Problem

- Write the function definition for a ‘large’ function that stores the larger of each element in arry1 and arry2 in arry3, where arry1 contains {1, 2, 3, 4, 5} and arry2 {5, 4, 3, 2, 1}. If the program works, arry3 should have {5,4,3,4,5}.

Solution

void large (int size, int arry1[], int arry2[], int arry3[]);
void main()
{
    const int size = 5;
    int arry1[] = {1,2,3,4,5};
    int arry2[] = {5,4,3,2,1};
    int arry3[size];

    large (size, arry1, arry2, arry3);
Arrays used as input

- What happens when we want to pass arrays to a function to be used only as output? We can’t pass it by value…
  - `void large (int size, const int arry1[], const int arry2[], int arry3[]);`
- We can protect array arguments by putting `const` in front of them in prototype and function definition

Problem

- 19.2: Assume you have two arrays of doubles called `vals1` and `vals2`. They both contain `maxels` elements. Write a C++ bool function identical that will accept both arrays and return true if both arrays are identical; otherwise, return false. The call to your function might be by a statement of the following form:
  ```cpp
  if (identical (maxels, vals1, vals2))
      cout << "Arrays are identical" << endl;
  else
      cout << "Arrays are not identical" << endl;
  ```

Summary

- In today’s lecture we covered
  - Passing arrays into functions
- Readings
  - Chapter 4