CS102 Beginning Programming Using C++

Quiz 2

Frank Mock

1. To sum the rows of a 2D array of integers, int days[29][5] one should use a for loop nested inside another for loop. The outer loop will move through each row and the inner loop will move through each column. Use an accumulator integer variable to keep track of the total.

int total = 0; //Accumulator

for(int row = 0; row < 29; row++)

{

for(int col = 0; col < 5; col++)

total += days[row][col];

}

//Display the sum

cout << “The total is “ << total << endl;

In the above example it’s preferred to use constant intergers when defining the array to represent the number of rows and the number of columns.

1. The following is the Binary Search function modified to search an

Array of strings. The sortStringArray function which this function calls follows.

int binarySearch(char array[][SIZE], int numElements, char value[][SIZE])

{

int first = 0,

last = numElements - 1,

middle,

position = - 1;

bool found = false;

//Sort array before starting search

sortStringArray(array, numElements);

while(!found && first <= last)

{

middle = (first + last) / 2; //calculate mid point

if(strcmp(array[middle], value[0]) == 0) //If value is found at middle

{

found = true;

position = middle;

}

else if(strcmp(array[middle], value[0]) > 0) //If value is in lower half

{

last = middle - 1;

}

else

{

first = middle + 1; //If value is in upper half

}

}

return position;

}

Here is the sortStringArray function that the binarySearch function calls.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Function definition for sortStringArray \*

//This function will alphabetically sort an array of \*

//strings. size is the number of elements in the \*

//array. \*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void sortStringArray(char stringArray[][SIZE], int size)

{

bool swap;

char temp[1][SIZE];

do

{

swap = false;

for(int count = 0; count < (size - 1); count++ )

{

if(strcmp(stringArray[count],stringArray[count + 1]) > 0)

{

strcpy(temp[0],stringArray[count]);

strcpy(stringArray[count],stringArray[count + 1]);

strcpy(stringArray[count + 1], temp[0]);

swap = true;

}

}

}while(swap);

}