// Merging two ordered lists to a third list **VERSION 1**

// This program will merge two sorted lists

// If the same element appears in both lists it will appear twice in the new list

// written by Meir Komar

// last update 19 Kislev 5780

1. #include <iostream>
2. #include <ctime>
3. const int MAXARRSIZE = 80;
4. using namespace std;
5. int main()
6. {
7. int arr1[MAXARRSIZE]; // the first list
8. int arr2[MAXARRSIZE]; // the second list
9. int arr3[MAXARRSIZE\*2]; // the new list
10. int i;
11. int pArr1 = 0; // The current index of arr1
12. int pArr2 = 0; // The current index of arr2
13. int pArr3 = 0; // The current index of arr3
14. int size1; // The size of the first array
15. int size2; // The size of the second array
16. srand((unsigned)time(NULL));
17. size1 = rand()%50 +1; // random size of arr1
18. size2 = rand()%50 +1; // random size of arr2
19. arr1[0] = rand()%10;
20. arr2[0] = rand()%10;
21. // fill the first array
22. for (i=1; i < size1; i++)
23. {
24. arr1[i] = arr1[i-1] + rand()%10;
25. }
26. // fill the second array
27. for (i=1; i < size2; i++)
28. {
29. arr2[i] = arr2[i-1] + rand()%10;
30. }
31. // print the arrays
32. cout << "\nFirst List: "<< endl;
33. for (i=0; i < size1; i++)
34. {
35. cout<<arr1[i]<<", ";
36. }
37. cout << "\nSecond List:" << endl;
38. for (i=0;i<size2;i++)
39. {
40. cout << arr2[i] << ", ";
41. }
42. // Merge the items until the end of one of the arrays
43. while (pArr1 < size1 && pArr2 < size2)
44. {
45. if( arr1[pArr1] < arr2[pArr2])
46. {
47. arr3[pArr3++] = arr1[pArr1++];
48. }
49. else
50. {
51. arr3[pArr3++] = arr2[pArr2++];
52. }
53. }
54. // Copy the remainder of the first array
55. while(pArr1 < size1)
56. {
57. arr3[pArr3++] = arr1[pArr1++];
58. }
59. // Copy the remainder of the second array
60. while(pArr2 < size2)
61. {
62. arr3[pArr3++] = arr2[pArr2++];
63. }
64. // Print the Merged List
65. cout<<"\nThe merged array: "<<endl;
66. for (i=0; i<size1+size2; i++)
67. {
68. cout << arr3[i] << ", ";
69. }
70. return 0;
71. }

// Merging two ordered lists to a third list **VERSION 2**

// This program will merge two sorted lists

// If the same element appears in both lists it will appear twice in the new list

// written by Meir Komar

// last update 3 shevat 5769

#include <iostream>

#include <ctime>

const int MAXARRSIZE = 80;

using namespace std;

void printArray(const int \*a, int size)

{

for (int i = 0; i < size; i++)

cout << a[i] << ", ";

}

int main()

{

int arr1[MAXARRSIZE]; // the first list

int arr2[MAXARRSIZE]; // the second list

int arr3[MAXARRSIZE\*2]; // the new list

int i;

int pArr1 = 0; // The current index of arr1

int pArr2 = 0; // The current index of arr2

int pArr3 = 0; // The current index of arr3

int size1; // The size of the first array

int size2; // The size of the second array

srand((unsigned)time(NULL));

size1=rand()%50; // random size of arr1

size2=rand()%50; // random size of arr2

arr1[0]=rand()%10;

arr2[0]=rand()%10;

// fill the first array

for (i=1;i<size1;i++)

{

arr1[i]= arr1[i-1]+rand()%10;

}

// fill the second array

for (i=1;i<size2;i++)

{

arr2[i]= arr2[i-1]+rand()%10;

}

// print the arrays

cout<<"\nFirst List: "<<endl;

printArray(arr1,size1);

cout<<"\nSecond List:"<<endl;

printArray(arr2,size2);

// Merge the items until the end of one of the arrays

while(pArr1<size1 && pArr2<size2)

{

if(arr1[pArr1]<arr2[pArr2])

{

arr3[pArr3++]=arr1[pArr1++];

}

else

{

arr3[pArr3++]=arr2[pArr2++];

}

}

// Copy the remainder of the first array

while(pArr1<size1)

{

arr3[pArr3++]=arr1[pArr1++];

}

// Copy the remainder of the second array

while(pArr2<size2)

{

arr3[pArr3++]=arr2[pArr2++];

}

// Print the Merged List

cout<<"\nThe merged array: "<<endl;

**printArray(arr3,size1+size2);**

return 0;

}