1: #include <stdio.h>

2: #include <stdlib.h>

3: #include <unistd.h>

4: #include <sys/wait.h>

5: #include <time.h>

6:

7: void bubble\_sort(int arr[], int n){ 8: int i, j, temp;

9: for(i = 0; i < n - 1; i++){

10: for(j = 0; j < n - i - 1; j++){

11: if(arr[j] > arr[j + 1]){

12: temp = arr[j];

13: arr[j] = arr[j + 1];

14: arr[j + 1] = temp;

15: }

16: }

17: }

18: }

19:

20: void insertion\_sort(int arr[], int n){ // Corrected the typo 21: int i, key, j;

22: for(i = 1; i < n; i++){ 23: key = arr[i];

24: j = i - 1;

25: while(j >= 0 && arr[j] > key){ 26: arr[j + 1] = arr[j];

27: j = j - 1;

28: }

29: arr[j + 1] = key;

30: }

31: }

32:

33: void print\_array(int arr[], int n){ 34: for(int i = 0; i < n; i++){

35: printf("%d ", arr[i]); 36: }

37: printf("\n");

38: }

39:

40: int main() { 41: int n;

42: printf("Enter number of elements: "); 43: scanf("%d", &n);

44:

45: int arr1[n], arr2[n];

46: printf("Enter the elements:\n"); 47: for(int i = 0; i < n; i++){

48: scanf("%d", &arr1[i]); // Corrected to arr1 49: arr2[i] = arr1[i];

50: }

51:

52: pid\_t pid = fork(); 53:

54: if(pid == 0){

55: printf("Child process sorting using insertion sort...\n"); 56: insertion\_sort(arr2, n);

57: printf("Child process sorted array: "); 58: print\_array(arr2, n);

59: sleep(5);

60: printf("Child process completed.\n"); 61: exit(0);

62: }

63: else{

64: printf("Parent process sorting using bubble sort...\n"); 65: bubble\_sort(arr1, n);

66: printf("Parent process sorted array: "); 67: print\_array(arr1, n);

68: printf("Parent sleeping for 10 seconds (child becomes zombie)...\n"); 69: sleep(10);

70: wait(NULL);

71: printf("Parent process completed.\n"); 72: }

73:

74: return 0;

75: }

