Orphan.c

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <stdlib.h>

// Bubble sort function
void bubblesort(int arr[], int n) {
   int i, j, temp;
   for (i = 0; i < n; i++) {
      for (j = 0; j < n - 1; j++) {
         if (arr[j] > arr[j + 1]) {
            temp = arr[j];
            arr[j] = arr[j + 1];
            arr[j + 1] = temp;
         }
      }
   }
}

// Insertion sort function
void insertionsort(int arr[], int n) {
   int i, j, temp;
   for (i = 1; i < n; i++) {
      temp = arr[i];
      j = i - 1;
      while (j >= 0 && temp < arr[j]) {
         arr[j + 1] = arr[j];
         j--;
      }
      arr[j + 1] = temp;
   }
}

int main() {
   int arr[25], n, i;

   printf("Enter the number of values in array: ");
   scanf("%d", &n);

   printf("Enter the array elements: ");
   for (i = 0; i < n; i++)
      scanf("%d", &arr[i]);

   printf("\n--- Starting Fork ---\n");
   pid_t pid = fork();
```

```
  if (pid < 0) {
      perror("fork failed");
      exit(1);
  }
  else if (pid == 0) {
      // CHILD process
      sleep(7);  // Sleep longer than parent so parent exits first
      printf("\nCHILD: My process ID is %d\n", getpid());
      printf("CHILD: My parent's ID is %d\n", getppid()); // Should be 1 if orphaned

      insertionsort(arr, n);
      printf("CHILD: Sorted elements using insertion sort: ");
      for (i = 0; i < n; i++) {
          printf("%d ", arr[i]);
      }
      printf("\nCHILD: Process finished and now exiting...\n");
      exit(0);
  }
  else {
      // PARENT process
      printf("\nPARENT: My process ID is %d\n", getpid());
      printf("PARENT: My child's ID is %d\n", pid);

      sleep(5);  // Parent exits before child
      printf("PARENT: Exiting now, child will become orphan...\n");
      exit(0);  // Parent exits here
  }

  return 0;
}
```

OUTPUT:-

```
pict@mplab-12:~/Desktop/33164$ gcc orphan.c -o orphan
pict@mplab-12:~/Desktop/33164$ ./orphan
Enter the number of values in array: 5
Enter the array elements: 8
7
5
7
4

--- Starting Fork ---

PARENT: My process ID is 8286
PARENT: My child's ID is 8343
PARENT: Exiting now, child will become orphan...
pict@mplab-12:~/Desktop/33164$
```

```
CHILD: My process ID is 8343
CHILD: My parent's ID is 2053
CHILD: Sorted elements using insertion sort: 4 5 7 7 8
CHILD: Process finished and now exiting...
```