Assignment2b.c

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

void mergeSort(int arr[], int l, int r);
void merge(int arr[], int l, int m, int r);

void merge(int arr[], int l, int m, int r) {

  int i, j, k;
  int n1 = m - l + 1;
  int n2 = r - m;

  int L[n1], R[n2];

  for (i = 0; i < n1; i++) {
    L[i] = arr[l + i];
  }
  for (j = 0; j < n2; j++) {
    R[j] = arr[m + 1 + j];
  }

  i = 0;
  j = 0;
  k = l;

  while (i < n1 && j < n2) {
    if (L[i] <= R[j]) {
      arr[k] = L[i];
      i++;
    }
    else {
      arr[k] = R[j];
      j++;
    }
    k++;
  }

  while (i < n1) {
    arr[k] = L[i];
    i++;
    k++;
  }

  while (j < n2) {
    arr[k] = R[j];
    j++;
    k++;
```

```
  }
}

void mergeSort(int arr[], int l, int r) {

  if (l < r) {

    int m = l + (r - l) / 2;

    mergeSort(arr, l, m);
    mergeSort(arr, m + 1, r);
    merge(arr, l, m, r);
  }
}

int main() {

  pid_t pid; //stores the pid

  int *arr,size,ch;

  printf("\nEnter size: ");
  scanf("%d",&size);
  arr=(int*)malloc(sizeof(int)*size);

  printf("\nEnter the numbers: "); //accepting elements
  for(int i=0;i<size;i++) {
    scanf("%d",&arr[i]);
  }

  printf("\nArray: ");  //printing original array
  for(int i=0;i<size;i++) {
    printf("%d ",arr[i]);
  }
  printf("\n\n");

  mergeSort(arr,0,size-1); //sorting

  printf("Sorted: ");
  for(int i=0;i<size;i++) {
    printf("%d ",arr[i]);
  }
  printf("\n\n");

  pid = fork(); //creating a new process

  if(pid==0) { //child

    printf("\n======In Child======\n\n");
    printf("I am child %d, my parent is %d\n\n",getpid(),getppid());

    char *buffer[size+1];
```

```
    buffer[0]="./binarysearch";

    for(int i=1;i<size+1;i++) {

      buffer[i]=malloc(sizeof(int));
      snprintf(buffer[i],sizeof(int),"%d",arr[i-1]);
    }
    buffer[size+1]="NULL";
          execv(buffer[0],buffer);
  }

  else if(pid>=1) {  //parent

    printf("\n======In Parent======\n\n");

    printf("ID: %d\n\n",getpid());

    int stat_val;
    pid_t child_pid;

    child_pid = wait(&stat_val); //waiting for child

    printf("\n=======Back in Parent======\n\n");
    printf("Child has finished: PID = %d\n\n", child_pid);
    printf("ID: %d\n\n",getpid());
  }
  exit(0);
}
```

binarysearch.c

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

int binarysearch(int arr[],int data,int left,int right) {

  while(left<right) {

    int mid=(left+right)/2;
    if(arr[mid]==data) {
      return mid;
    }

    if(arr[mid]>data) {
      return binarysearch(arr,data,left,mid);
    }

    else {
        return binarysearch(arr,data,mid+1,right);
```

```
      }
    }
    return -1;
}

int main(int argc, char *argv[]) {

  printf("\n======In Binary Search======\n\n");
  printf("ID: %d\n",getpid());

  int arr[argc-1],size=argc-3,data,ch;

        for(int i=1;i<size+1;i++) {
    sscanf(argv[i],"%d",&arr[i-1]);
  }

  do {
    printf("\nEnter the value to be searched: ");
        scanf("%d",&data);

        int pos=binarysearch(arr,data,0,size);

    if(pos!=-1) {
      printf("\nValue Found at position: %d\n\n",pos);
    }
    else {
      printf("\nNot found\n\n");
    }
    printf("====================\n\n");
    printf("Continue (0/1): ");
    scanf("%d",&ch);
  }
  while(ch!=0);

  return 0;
}
```

Output:

```
Someshwars-MacBook-Pro:Assignments someshwargaikwad$ ./Assignment2b
Enter size: 7
Enter the numbers: 21 12 17 84 34 25 8
Array: 21 12 17 84 34 25 8
Sorted: 8 12 17 21 25 34 84

======In Parent======
ID: 2975

======In Child======
I am child 2980, my parent is 2975

======In Binary Search======
ID: 2980
Enter the value to be searched: 5
Not found

====================
Continue (0/1): 1
Enter the value to be searched: 34
Value Found at position: 5

====================
Continue (0/1): 1
Enter the value to be searched: 84
Value Found at position: 6

====================

Continue (0/1): 1
Enter the value to be searched: 56
Not found

====================
Continue (0/1): 0
=======Back in Parent======
Child has finished: PID = 2980
ID: 2975
Someshwars-MacBook-Pro:Assignments someshwargaikwad$
```