```
/*--- hw25.txt ---*/
/*
#include <stdio.h>
#include <stdlib.h>


//Prototypes
char** read_letter(FILE* readfrom, int height, int length);
void print_letter(char** letterArray, int height, int length);


//Main
int main() {
  FILE* fin = fopen("letters.txt", "r");

  int numletters = 4;
  int height = 4;
  int length = 7;
  char*** word = calloc(numletters, sizeof(char**));

  //read letters
  for (int i = 0; i < numletters; i++) {
    word[i] = read_letter(fin, height, length);
  }

  //swap first and second
  char** temp = word[0];
  word[0] = word[1];
  word[1] = temp;

  //print letters and free the head-based arrays
  for (int i = 0; i < numletters; i++) {
    print_letter(word[i], height, length);
    for (int j = 0; j < height; j++) {
      free(word[i][j]);
    }
    free(word[i]);
  }
  free(word);

  return 0;
}


//Definitions
char** read_letter(FILE* readfrom, int height, int length) {
  char** letter = calloc(height, sizeof(char*));
  for(int i = 0; i < height; i++) {
    letter[i] = calloc(length, sizeof(char));
  }
  for(int i = 0; i < height;i++) {
    for(int j = 0; j < length; j++) {
      letter[i][j] = fgetc(readfrom);
    }
    while(fgetc(readfrom) != '\n') { }
  }

  return letter;
}


void print_letter(char** letterArray, int height, int length) {
  for(int i = 0; i < height; i++) {
    for(int j = 0; j < length; j++) {
      printf("%c", letterArray[i][j]);
    }
    printf("\n");
  }
}
\n*/

/*--- oddfirst.c ---*/
//Mike Hanling
//oddfirst.c

#include <stdio.h>
#include <stdlib.h>


int* read_nums(int length);
void selectionSort(int* data, int size);
int* splitodds(int* fullarray, int size, int* count);
int* splitevens(int* fullarray, int size, int* count);
void print_nums(int* first, int size1, int* second, int size2);


int main() {
  int len = 10;

  //reads ints
  int* nums = read_nums(len);

  //sorts ints
  selectionSort(nums, len);

  //splits odds and evens
  int oddcount = 0;
  int* odds = splitodds(nums, len, &oddcount);
  int evencount = 0;
  int* evens = splitevens(nums, len, &evencount);

  //prints odds then evens
  print_nums(odds, oddcount, evens, evencount);

  free(nums);
  free(odds);
  free(evens);

  return 0;
}


int* read_nums(int length) {
  int* nums = calloc(length, sizeof(int));

  for (int i = 0; i < length; i++) {
    scanf(" %i", &nums[i]);
  }

  return nums;
}

void selectionSort(int* data, int size) {
  for(int length = size; length > 1; --length) {
    // Find imax, the index of the largest
    int imax = 0;
    for(int i = 1; i < length; ++i) {
      if (data[imax] < data[i]) {
        imax = i;
      }
    }
    // Swap data[imax] & the last element
    int temp = data[imax];
    data[imax] = data[length - 1];
    data[length - 1] = temp;
  }
```

```
}


int* splitodds(int* fullarray, int size, int* count) {
  int* odds = calloc(size, sizeof(int));

  for (int i = 0; i < size; i++) {
    if (fullarray[i] % 2 == 1) {
      odds[*count] = fullarray[i];
      *count += 1;
    }
  }

  return odds;
}


int* splitevens(int* fullarray, int size, int* count) {
  int* evens = calloc(size, sizeof(int));

  for (int i = 0; i < size; i++) {
    if (fullarray[i] % 2 == 0) {
      evens[*count] = fullarray[i];
      *count += 1;
    }
  }

  return evens;
}


void print_nums(int* first, int size1, int* second, int size2) {
  for (int i = 0; i < size1; i++) {
    printf("%i ", first[i]);
  }
  for (int i = 0; i < size2; i++) {
    printf("%i ", second[i]);
  }
}
```