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

/*Orfilia Qiu*/


int main(int count, char **items){

	FILE *output = fopen("Fibonacci.txt", "w+");
	FILE *FibEven = fopen("Fibonacci.even.txt", "w+");
	FILE *FibOdd = fopen("Fibonacci.odd.txt", "w+");
	unsigned long int medium = 0;

	int i =0;
	printf("Your command line: ");
	printf("Count");
	while(i<count){
			printf(" %s", *items);
			items++;
			i++;
	}

	int p = count;
	while(p--){
			items--;
	}
	printf("\n\n");



	//checking input stream
	if(count>4 || count<4){
			printf("Too many or too few arguments\n");

	}else if(count==4){

			//checking for non numeric
			int containsChar = 0, m=0;
			while(m<strlen(items[1])){
					char *toSec = items[1];

					if(((toSec[m])!= 0) &&(((toSec[m])<48) ||
((toSec[m]) >57))){
						containsChar =1;
						break;
					}
					m++;
					toSec++;
			}
			if(!containsChar){
					int v =0;
					while(v< strlen(items[3])){
						char *toForth = items[3];
```

```
                        if(((toForth[v])!= 0) &&(((toForth[v])<48) ||
((toForth[v]) >57))){
                              containsChar =1;
                              break;
                        }
                        v++;
                        toForth++;
                  }
                  }
                  if(containsChar){printf("Illegal argument\n");

                  }else{
                        if((atoi(items[1])+ atoi(items[3]))>2){
                        unsigned long int x = atoi(items[1]), y =
atoi(items[3]);

                        unsigned long int totEle = (x+y);
                        printf("Calculating %lu Fibonacci
numbers\n\n", totEle);
                        fprintf(output, "%lu ", 1);
                        printf("1.File Fibonacci content: \n%d ", 1);
                        unsigned long int j =1, pre = 0, next=1;
                        unsigned long int num =0;
                        while(j<totEle){
                              num= next+pre;
                              fprintf(output, "%lu ", num);
                              printf("%lu ", num);
                              pre = next;
                              next =num;
                              if(num<pre){
                              printf("\n\nAddition overflow. Stop
generating next number.");
                              j++;
                              break;
                        }
                              j++;
                              if(j%4 ==0){
                                    fprintf(output, "\n");
                                    printf("\n");
                        }

                  }
                  printf("\n\n  Total: %d numbers in Fibonacci \n\n", j);

                  rewind(output);
                  fflush(output);
                  //computing Fibonacci medium
                  unsigned long int mediumIdx = j/2, counted=0;
                  unsigned long int temp=0;

                  while(fscanf(output, "%lu", &temp)!=EOF){
                        counted++;
                        unsigned long int temp2=0;
```

```
                              if((counted==mediumIdx) && (j%2==0)){
                              fscanf(output, "%lu", &temp2);
                              medium = (temp+temp2)/2;

                              break;}
                              else if((counted == mediumIdx) && (j%2==1)){
                              fscanf(output, "%lu", &temp2);
                              medium = temp2;
                              break;}
                  }

                  printf("2.The Fibonacci medium: %lu\n\n", medium);
                  fflush(output);
                  rewind(output);

                  unsigned long int oddCount=0, evenCount=0;
                  unsigned long int seg=0;
                  while(fscanf(output, "%lu", &seg)!=EOF){
                              if((seg%2)==0){
                                    fprintf(FibEven, " %lu ", seg);
                                    evenCount++;
                                    if(evenCount%4==0){fprintf(FibEven,
"\n");}
                              }else{
                                    fprintf(FibOdd, " %lu ", seg);
                                    oddCount++;
                                    if(oddCount%4==0){fprintf(FibOdd,
"\n");}
                              }

                  }
      fclose(output); //done with Fibonacci Number file
      fflush(FibEven);
      rewind(FibEven);
      fflush(FibOdd);
      rewind(FibOdd);
                  //computing even numbers medium
                  unsigned long int evenMid = evenCount/2, evenLoopCount=0,
trueEven;
                  unsigned long int evenSeg=0;
                  while(fscanf(FibEven, "%lu", &evenSeg)!=EOF){
                              //base case of 1 even number
                              if(evenCount==1){
                                    trueEven=evenSeg;
                                    break;
                                    }
                              evenLoopCount++;
                              unsigned long int evenSeg2 =0;

      if((evenLoopCount==evenMid)&&(evenCount%2==0)){
                                    fscanf(FibEven, "%lu", &evenSeg2);
                                    trueEven = (evenSeg+evenSeg2)/2;
                                    break;
                              }
```

```
                              else if((evenLoopCount == evenMid) &&
(evenCount%2==1)){
                                    fscanf(FibEven, "%lu", &evenSeg2);
                                    trueEven=evenSeg2;
                                    break;
                              }
                  }

                  //computing odd numbers medium
                  unsigned long int oddMid = oddCount/2, oddLoopCount =0,
trueOdd;
                  unsigned long int oddSeg =0;
                  while(fscanf(FibOdd, "%lu", &oddSeg)!=EOF){
                              oddLoopCount++;
                              unsigned long int oddSeg2=0;
                              if((oddLoopCount==oddMid) &&
(oddCount%2==0)){
                                    fscanf(FibOdd, "%lu", &oddSeg2);
                                    trueOdd = (oddSeg+oddSeg2)/2;
                                    break;
                              }
                              else if((oddLoopCount == oddMid) &&
(oddCount%2==1)){
                                    fscanf(FibOdd, "%lu", &oddSeg2);
                                    trueOdd=oddSeg2;
                                    break;
                              }
                  }

                  printf("3.Total: %lu numbers in Fibonacci.even
\n",evenCount);
                  printf("  Total: %lu numbers in Fibonacci.odd
\n",oddCount);
                  printf("  Fibonacci.even medium: %lu \n",trueEven);
                  printf("  Fibonacci.odd medium: %lu \n",trueOdd);

      fflush(FibEven);
      rewind(FibEven);
      fflush(FibOdd);
      rewind(FibOdd);

                  //pointing to the end of odd file
                  unsigned long int pointToLast=0, oddFinalCount;
                  while(fscanf(FibOdd, "%d",
&pointToLast)!=EOF){oddFinalCount++;}

                  //appending even numbers to odd file
                  unsigned long int finalCount =oddFinalCount;
                  unsigned long int finalSeg=0;
                  while(fscanf(FibEven, "%lu", &finalSeg)!=EOF){
                              finalCount++;
                              fprintf(FibOdd, " %lu", finalSeg);
                              if(finalCount%6==0){fprintf(FibOdd,"\n");}
                  }
```

```
                //print on screen
                printf("\n4.Total count in Fibonacci.odd is now: %d\n",
(finalCount));
                fclose(FibEven);
                fflush(FibOdd);
                rewind(FibOdd);
                unsigned long finalOddSeg=0;
                int printCount=0;
                while(fscanf(FibOdd, "%lu", &finalOddSeg)!=EOF){
                            printCount++;
                            printf("%lu ", finalOddSeg);
                            if(printCount%5==0){printf("\n");}
                }

      fflush(FibOdd);
      rewind(FibOdd);

                //removing \n in odd File
                /*int posInFile =0;
                char ch = ' ';
                while((ch=fgetc(FibOdd)) != EOF){
                            if(ch == 10){
                                  ch=fgetc(FibOdd);
                                  if(ch==10){
                                  fseek(FibOdd, ftell(FibOdd)-1,
SEEK_SET);
                                  fprintf(FibOdd, "%c", ' ');}else{
                                  fseek(FibOdd, ftell(FibOdd)-1,
SEEK_SET);
                                  fprintf(FibOdd, "%c". );
                                  }
                            }
                }
      fflush(FibOdd);
      rewind(FibOdd);

                int cnt = 0;
                while((ch=fgetc(FibOdd)) != EOF){
                            if(ch == 32){cnt++;}

                            if((cnt%6==0) && (cnt!=0)){
                            fseek(FibOdd, ftell(FibOdd)-1, SEEK_SET);
                            fprintf(FibOdd, "%c", 10);

                            }
                }*/
                printf("\n\n");

      }else{
                printf("N1 + N2 is less than 2. Program terminated\n");
      }//end of if inside else

      }//end of else
```

```
        }
        fclose(FibOdd);
        return 0;
}
```