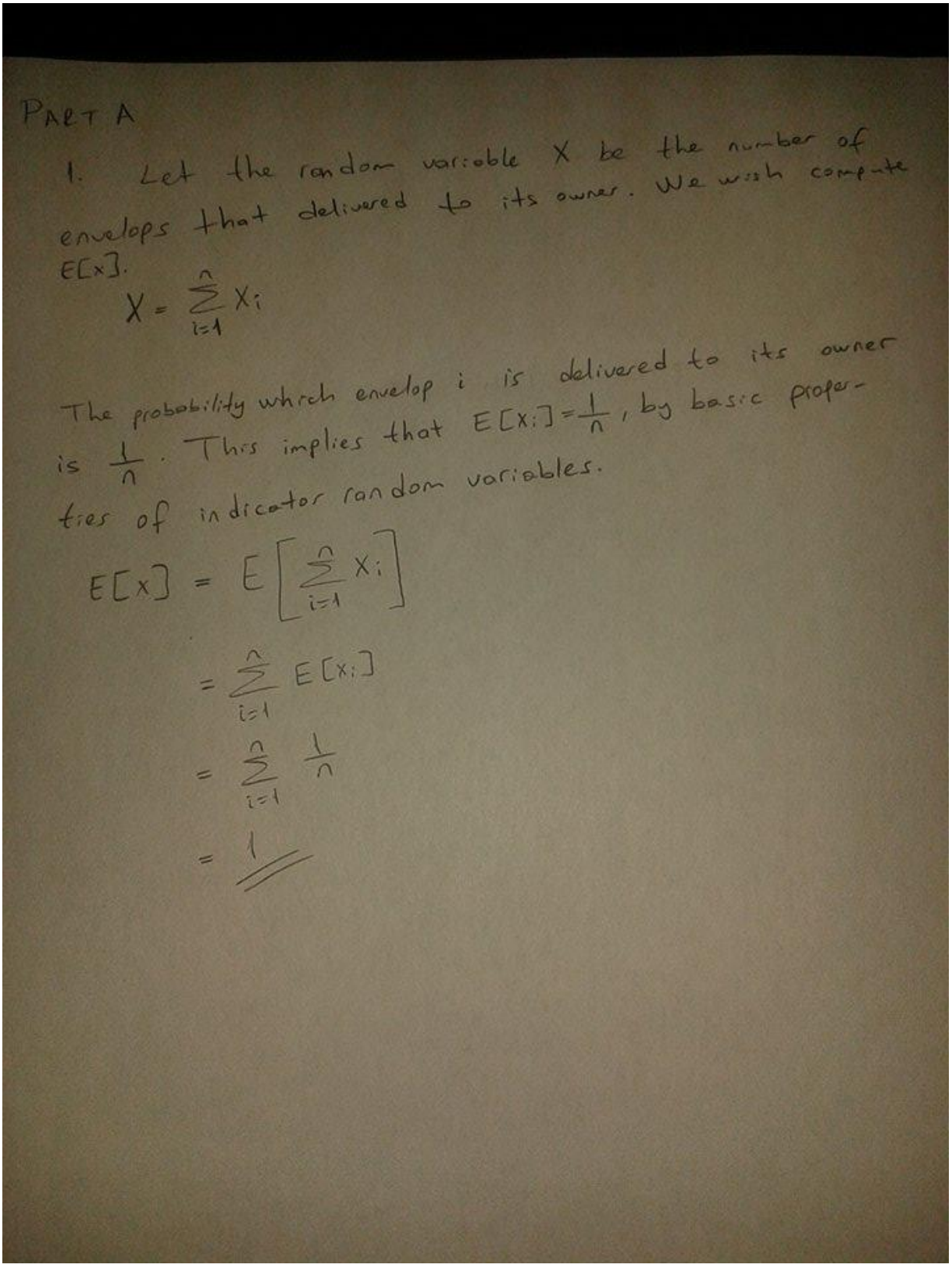
Part A.

1.



PART A

1. Let the random variable X be the number of envelops that delivered to its owner. We wish compute E[X].

$$X = \sum_{i=1}^{n} X_i$$

The probability which envelop i is delivered to its owner is $\frac{1}{n}$. This implies that $E[X_i] = \frac{1}{n}$, by basic properties of indicator random variables.

$$E[X] = E\left[\sum_{i=1}^{n} X_i\right]$$

$$= \sum_{i=1}^{n} E[X_i]$$

$$= \sum_{i=1}^{n} \frac{1}{n}$$

$$= 1$$

2.

Part B.

1. In this part, we sorted line number in mobydick.txt file by using quick sort algorithm. Then, in M(th) operation, N(th) element of array is print out. For example, for M=100 and N=500 we printed out **353.**

```
[budakf@ssh oop_algo]$ g++ Source.cpp
[budakf@ssh oop_algo]$ ./a.out 1 100 500
353
```

2. In this part, we stored words and word numbers in vector and associate each other line by line with mapping operation. Then word numbers are sorted by using random quick sort algoritm. Finally, all sentences are printed to novel.txt in order.

Unsorted of line 551 is that:

551 {how_Ishmael._mind_Call_#_ago-_Some_years_never_precisely-_#_me_long} {10_4_9_2_1_7_5_6_8_12_0_3_11}

First Step:

{how_Ishmael._mind_Call_#_ago-_Some_years_never_precisely-_#_me_long} and **{10_4_9_2_1_7_5_6_8_12_0_3_11}** are stored in **vector<string>** variables

Second Step: each elements of **{10_4_9_2_1_7_5_6_8_12_0_3_11}** are converted to integer and are sorted using random quick sort algorithm. As a result, We got **0 1 2 3 4 5 6 7 8 9 10 11 12.**

Final Step: We printed out this sentence in sequence**. # # Call me Ishmael. Some years ago- never mind how long precisely-**

Novel.txt file in nano editor.

```
[budakf@ssh oop_algo]$ nano novel.txt
  GNU nano 2.3.1              File: novel.txt

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # 18$
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # MOBY DIC$
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # OR THE WHA$
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # by Herman Melvil$
ETYMOLOGY
# # ETYMOLOGY
# # (Supplied by a Late Consumptive Usher to a Grammar School)
#
# # The pale Usher- threadbare in coat, heart, body, and brain; I see
him now. He was ever dusting his old lexicons and grammars, with a
queer handkerchief, mockingly embellished with all the gay flags of
all the known nations of the world. He loved to dust his old grammars;
it somehow mildly reminded him of his mortality.
#
# # "While you take in hand to school others, and to teach them by
what name a whale-fish is to be called in our tongue leaving out,
through ignorance, the letter H, which almost alone maketh the
signification of the word, you deliver that which is not true."
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # $
#
# # "WHALE. * * * Sw. and Dan. hval. This animal is named from roundness
                          [ Read 19932 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
```

3. Running times of program for some K values:

```
root@conquerrorr:~/Masaüstü# g++ x.cpp
root@conquerrorr:~/Masaüstü# ./a.out 2 100
It took me 34 clicks (0.000034 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 1000
It took me 472 clicks (0.000472 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 10000
It took me 3624 clicks (0.003624 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 100000
It took me 46186 clicks (0.046186 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 1000000
It took me 462689 clicks (0.462689 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 2000000
It took me 974882 clicks (0.974882 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 3000000
It took me 1491198 clicks (1.491198 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 4000000
It took me 2057088 clicks (2.057088 seconds).
root@conquerrorr:~/Masaüstü# ./a.out 2 5000000
It took me 2656401 clicks (2.656401 seconds).
root@conquerrorr:~/Masaüstü#
```

Fatih Budak
150130006

Analysis of Algorithms 1
Project 2 Report

Screenshot of numbers.txt for K=50000.

```
[budakf@ssh oop_algo]$ ./a.out 2 50000
[budakf@ssh oop_algo]$ nano numbers.txt
  GNU nano 2.3.1            File: numbers.txt

450
550
660
835
1095
1148
1220
1578
1612
1709
2013
2039
2243
2383
2461
2566
2646
2755
2761
3070
3157
                          [ Read 50000 lines ]
```