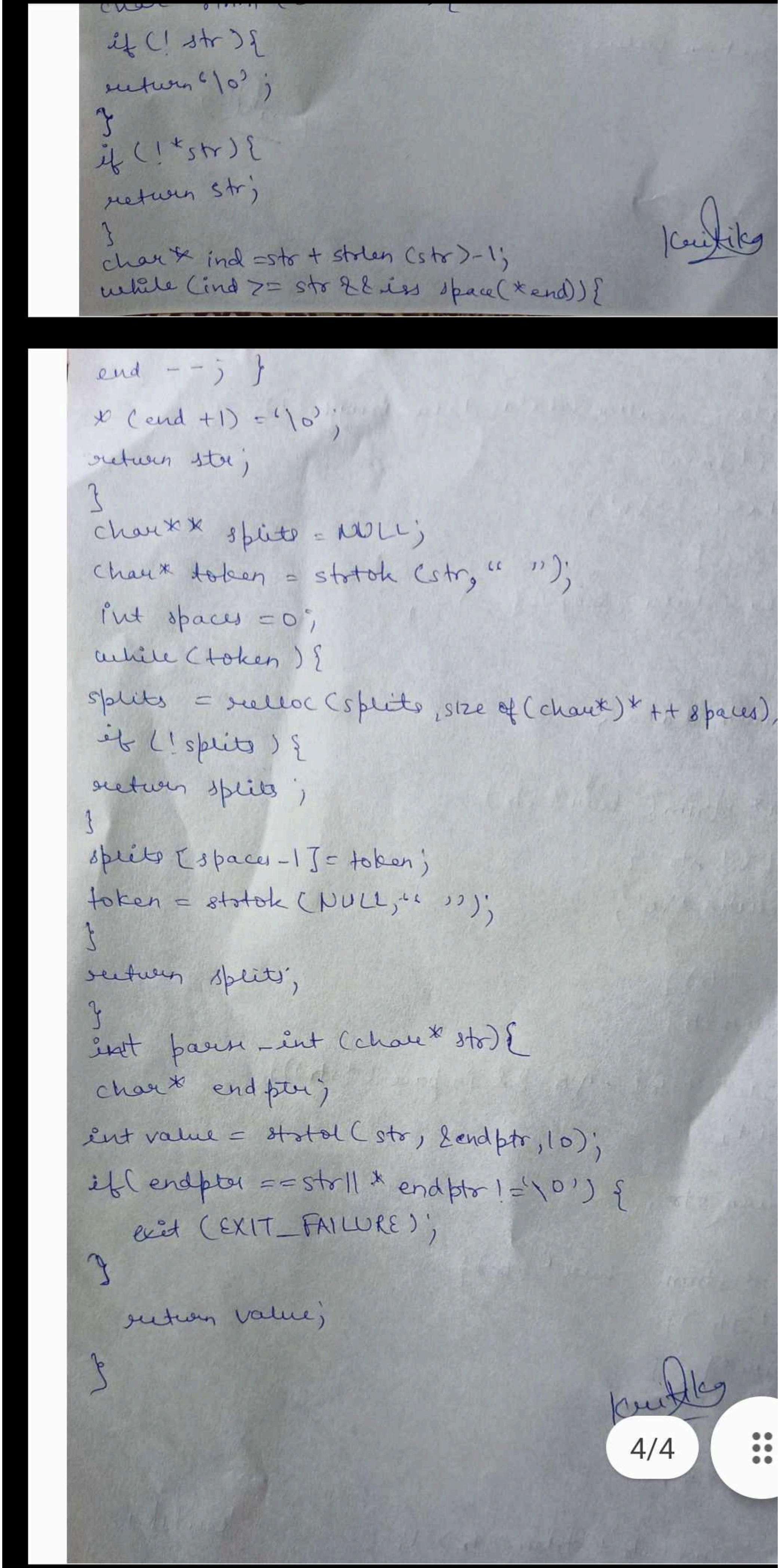


```
if (! str){
return '\0';
}
if (!*str){
return str;
}
char* ind =str + strlen (str)-1;
while (ind >= str && iss space(*end)){
```

Kritika

```
end --; }
*(end +1) = '\0';
return str;
}
char** splits = NULL;
char* token = strtok (str, " ");
int spaces =0;
while (token){
splits = realloc (splits, size of (char*) * ++ spaces),
if (! splits){
return splits;
}
splits [spaces -1] = token;
token = strtok (NULL, " ");
}
return splits;
}
int parse_int (char* str){
char* endptr;
int value = strtol ( str, &endptr, 10);
if( endptr == str|| * endptr != '\0') {
    exit (EXIT_FAILURE);
}
    return value;
}
```

Kritika

Name - Akanksha Dabral
Roll no.- 2023017

Operating System

Akanksha

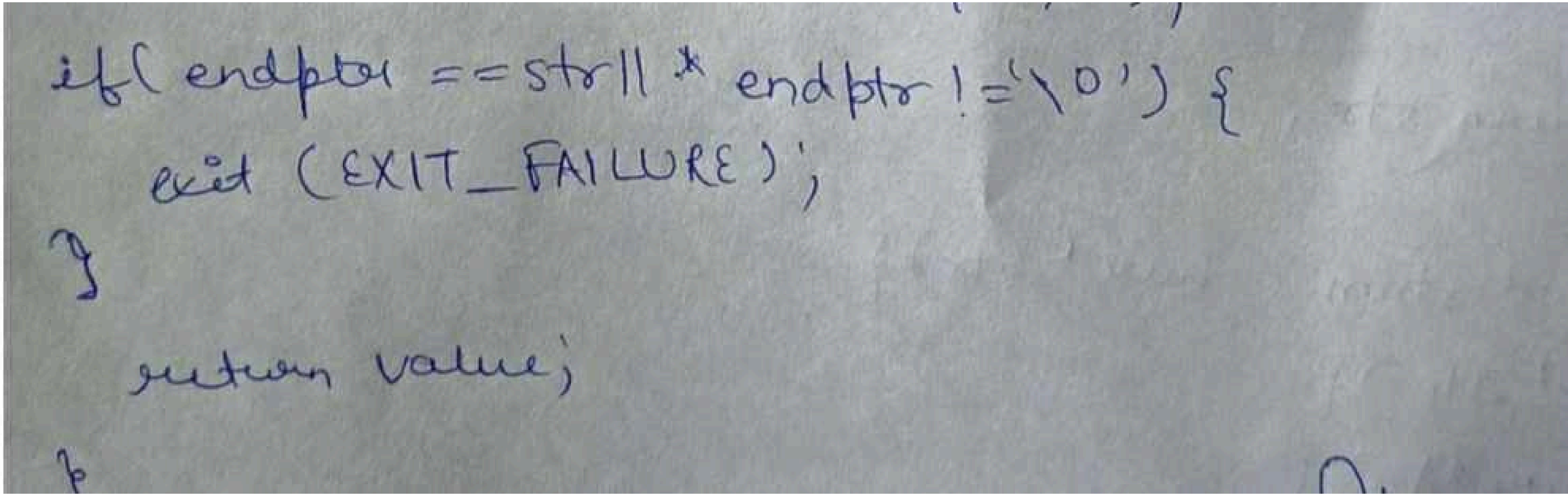
Ans1 Code

```
# include <stdio.h>
# include < limits.h>
# include < ctype>
# include< math.h>
# include< stddef.h>
# include < stdlib.h>
# include< stdin.h>
# include<string.h>
Char * readline ();
Char* ltrim(char*);
Char* rtrim(char*)
int parse_int (char*);
 int main ()
{
  file* fptr = foper(getens ("output_path"), "w");
  int n = parse_int ( rtrim ( readline () ));
  put ** customers = malloc (n* size of int*));
   for (int i=0 ; i<n, i++)
   {
     * (customers + i) = malloc (2* size of(int)));
   char* * customers_item_temp = split _string (strium (readers)
     for(int j=0, j<2, j++)
    {
     int customers_item = pas_ilat (* customers_ item_temp+j'));
```

```
((customers + i) +j) = customers_item; } }
 int result = minimum Average (n,2, customers);
  fprintf (fptr, "%d\n", result);
   fclose (fptr);
  return 0;
 }

char* readline () {
size_t alloc_length =1204;
 size_t data length =0;
char* data = malloc( alloc_length);
 While ( true)
{
 char* cursor = data + data_length;
 char* line = fgets (cursor, alloc-length_data_length, stdin),
   if (!line)
 {
    break; }
if (data_length < alloc_length - 1 // data [data_length _1]='\n')
 {
  break;
 }
 alloc_length [ < = 1;
data = realloc ( data , alloc_length);
 if (!data)
   data = '\0' ;
{ break;
if ( data ( data_length-1] == '\n')
   data [data_length - 1] = '\0'; } }
  else
 {
  data = realloc( data , data.length + 1 );
    if(!data )
  {
    data = '\0';
   }
     else {
   data [ data_length] = '\0' ;
   }
     }
   return data ;
   }
  char*  trim (char* str)
    {
     if (!str) {
        return '\0' ;
    }
      if (!*str) {
     return str;
    }
     while(*str != '\0' && inspace (*str)) {
      str ++; }
    return str ;
     }
     char*  rtrim (char * str) {
       if (!str){
      return '\0';
     }
```

```
if (!*str){
return str;
}
 char* ind = str + strlen(str)-1;
while (ind>=str && iss space(*end) {
end--; }
 * (end+1 = '\0';
  return str;
 }
 char** splits=NULL;
 char* token= strtok(str," ");
  int spaces=0;
 while (token) {
  Splits = realloc(splits, size of (char*) * ++spaces);
   if (!splits) {
   return splits;
  }
   Splits[spaces-1] = token;
   token = strtok(NULL," ");
  }
   return splits
 }
  int parse-int(char* str){
  char* end ptr;}
  int value = strtol(str, &end ptr, 10);
```



```
if( endptr == str || *endptr != '\0') {
    exit (EXIT_FAILURE);
}

   return value;

}
```