Error ID	points	Error description
COM	80	compilation error on a VM
PRM	5	permission problems
SBM	5	wrong submission file name
FOL	5	folders inside zip file
ZIP	10	submission file isn't zip
SH	5	wrong name for script file
CIP	5	wrong name for c file
XTR	10	extra files
A	2	there should be exactly 2 cp commands - only the first two files should be copied
B	2	there should be exactly 1 mv command - only the third file
C	2	wrong folder names
D	2	the flags in ls commands should be -la
E	1	missing one of descriptive text
F	2	didn't exit and/or deleted the temp folder
G	2	didn't reverse the file names
H	2	didn't remove the 2 first files separately before removing the entire folder
I	2	used extra commands
J	5	wrong functionality
K	2	didn't set the first directory (with the ID) as the current directory
L	1	the file names shouldn't have any extensions (like .txt)
M	2	didn't echo the contens directly (for example first echoed into a file and then used cat file)
N	2	displayed first the contents of the file with the first name inside it (the name of the file as shown in the echo message should be reversed with the contents)
O	5	used absolute address
P	2	created the files (with touch, for example) before echoing the contents into them.
Q	2	the reversal was performed separately from the cp command
R	2	removed the third file separately before removing the entire dir
S	5	The content of the files is never printed.
T	10	doesn't run properly (at all)
a	5	once or twice: didn't check return values of ALL system calls except close() and closedir().
b	10	repeated: didn't check return values of ALL system calls except close() and closedir().
c	5	once or twice: On an error: print a descriptive error message (string + result of strerror(errno)) and exit.
d	10	repeated: On an error: print a descriptive error message (string + result of strerror(errno)) and exit.
e	5	Should open output files with O_TRUNC, so that if they exist their content is truncated
f	5	when transferring data - should use a static buffer and not malloc() it
g	5	Use only system calls shown in class. These are: open, read, write, close, opendir, readdir, closedir, stat, lseek. No C functions except printf() for output, strcmp() for comparing filename to "." or "..", and sprintf() for concatenating the f
h	20	shouldn't read files to memory!
i	5	shouldn't use file size from stat call. instead, read both files until EOF.
j	5	didn't set a buffer of constant size to read into
k	5	didn't set variable to actual number of bytes read - might be less than 1024
l	5	didn't repeat reading of key file until reaching the right number of bytes
m	0	didn't repeat until everything is written (you need to check write's return value in the code beyond just making sure it's not negative!)
n	5	didn't free all allocated memory
o	5	reading too few bytes at a time (less than 100)
p	40	code crashes
q	10	doesn't work with large input file (the encryption is not correct when tested with an input file larger than the key file or it takes too long to finish)
r	10	doesn't work with large key file
s	5	didn't close properly all opened files
t	10	doesn't work when input is smaller than the key
u	10	doesn't work when input and key have read only permissions
v	10	doesn't work when output folder has only write permission (usually because the existense of the output folder is checked using opendir)
w	1	don't use perror
x	10	Memory usage should be at most 1GB, and no more than 1MB consecutive
y	20	encryption doesn't pass diff test for any size

z	5	permissions error when there is already existing output file in the output folder.
empty	10	says that key file is empty when it's not
c99	10	can only be compiled with flag -std=c99
run	20	failed to run the cipher code