The obstacles that I met in this program is that the algorithm of this project. So I decided to develop incrementally. The second obstacle is to debug. So I used debugger to debug step by step. The third obstacle is writing all test cases. I used a random string generator from a websit e to test my program.

**Pseudo code**

Initializing writable char arrays for crib, ciphertext and output

Initializing counters

Initializing hashtable to -1

clearArray ()

reapeatedly

if isalpha

write to crib

deal with counters

else

deal with ocunters

clean up spaces in the beginning and the back

//clearCrib

repeatedly

if index more than 90

return false

if isalpha

write to crib

deal with counters

else

deal with ocunters

clean up spaces in the beginning and the back

repeatedly

if both alpha or both non-alpha

keep index

repeatedly

match with hash table

if match

break

if not match

reset hash table

continue

repeatedly

if ciphertext string is alpha, not match in hash table

change to lower case

else

match with hash table and change to upper case

**test cases:**

// testing 2 words crib

runtest("Hirdd ejsy zu drvtry od.\nO'z fodvtrry.\n", "my secret");

// testing false return

runtest("Hirdd ejsy zu drvtry od.\nO'z fodvtrry.\n", "shadow");

// testing lots of useless characters

runtest("F gspt fe! zyxZYXzyx--Abca abCa bdefg## $$dsptrqtj6437 wvuWVUwvu\n\n8 9\n", " hush???hUSh--- --- until NovemBER !! ");

//testing inputs with multiple newline characters

runtest("kvbz pqzzyfq bz abcdef zqxjqk\n", "secret");

runtest("kvbz pqzzyfq bz gqbjw\n", "secret");

runtest("Rswjo qgx Tmeuo sgjsy jds vqgf vo jds vqzf xbby. \nUdbyjo iqcju cg wybgj cg jds esjqiqo zqy \nXbg'j rsj jds jsrrsy jycn jds ucrsgj qrqyt. \nZU 31 cu zdqrrsgecge!", "silent alarm");

// testing spcaes

runtest(" ", " ");

// testing empty string

runtest("","");

// testing with newline character between match crib

runtest("adedf ef\n abc\n", "my xyz");

runtest("iUIZcy2EvwQ0SdbGyjFePQ1IVKHQJPVFhPmiSEwtcg9o4lrVvAWqxuXyJTRXPnPuUmo4EZ\n"

"ZavsLVhrHY1WezTFEVws6NaXJYA4ql9RzO5srJ5ssSoUkCDGPZLOnBK14lcQ6cVJ4w8oTC\n"

"SoUTtQ7rl5pAQ5bMzpuNuxKfOg7isnzBLGpHjriixPnatE0BViGopTK1pD7rYg94LZTt31\n"

"HaLZdE3oLJuZqI6KwY86Bg4dDlZPXLuMUukR8eDXJUVtgJv8uWuRYP9XRW338npvyCcztJ\n"

"S3btgRlkPmmxa8Zdr9SPUuNzGrEHLF1cBMswCENBW89r3Zj87fP8B6sza0d3znuvIkjvnf\n"

"IDdffuQdTsVGro7ygNDl7MbYX632DpXUWJXvQ0xIZoaI9N4OyB95fRIIBlCxX2HNw4HP6w\n"

"D4ClDN73axoNzu16DGHmzMXJ5zSwFjpNDcgyQz5TYVehckLLkSl7Nj3S5zlPTkjVWs7Kw2\n"

"xJUvIJr36p1ntKOV0Dq7L6im2AXucCPsm7G8pdsL9a0BG0UI3NZYuzZZtIoTE65RjGLF0y\n"

"fTbXIbadu7YJEMYZd3kM382OfkzP3rsUjkAn4ph4Da28DkLeZUvOV4IZtLecrPrf7QYmF8\n"

"aYBwCt53T1wodk7ZEsgnJxJSOlQlcl90EHWidJ4xdSdVnWGjIdmuSHnJEpxufi8h7Bmy0f\n"

"7A2YGiP5UG6elscSKYVCvOOCM6Dm3iGVCIXoQGexdd7F9zoieMmtzshbmmr0Ew50AgvTzQ\n"

"7Cr8v7qtL1REFxgYviIpW4chQAfRdvH0mQC1fyW8g0B6xe3E6dK4dxAgM1HteOPpthlD9P\n"

"KuAmv5xXEGq3fsq3xNZLbl4qD1EXEE3KR6YFndFafOp3YTC0lb4JOVXXTU5R6ad9vOSra8\n"

"h6A6AAc8oxYMxveyQch8dIKkZk8c3KzcWrD3OgQ7Mm2NJNnJvjNlaEEPgVnzy65Y1PMqLC\n"

"e9xAHWTWFomyfBs2kBLquK3R0xpH4q4bEt446mBm8sN7lDDTiz0Zp1uC74l5BDeYNOK05u\n"

"qPwtnXuwj5ycpdjPONONVeKht33woBHebD2yej2YCPkctrThZY0QsxerkcjbTdYTTa77KY\n"

"EkUbPPqJbaPhQMXIZR814OpZ2H8xiw0eENm0YpWOmfwmmiGgWkVEJqAK8MYjybvRmoKaRh\n"

"SNRJpe0pzU7kTcKLBwXKVJpcOKAscAHiNjPzxsXr2DhXsfZyY9wVhBzGyuQtbCpGBCbspG\n"

"sQ8CX5pBUXdhpgv3YVjdft45dJKmFogNvMq93dVwXGMetuLnFuxQ9jHKpXzsQo6kv6fuCt\n"

"oLelZzscOXFjJizhoxqKKCNm9pNJnLffgFkWnOgCtjqf5niQfYNiugn4KQ025c38ePEeba\n"

"189NwqTzVBOeJuEyLRTQf3tlpszuV2k72a417oMFrNs84vUl522ILn4vjC9bgQGlLPdZd6\n"

"IxGODQcY3c6RWrC2Tvz9tL430GHMuhjxjiCyLu5YOznnfnnbhoJtmNUIuOZ8N1qqbtYXuX\n"

"EQiWRvNOhhxrNinYon6ABH2MRA6WxRRm788izKPNrEqfYmev2AlxbRJylOwHt3I52kpl09\n"

"7ev2mPZnTOV8fDHlguOAHZ0nPIEuueIWHHLKvHqUXnsWBXCBDtxZTraylDZn5wtXrgGigy\n"

"JuAZhYawjvACgeZE83V7dsnCfjKysotPfq2deIpxNLuyBzH2m26fObXam4299IgT431svw\n"

"MKvtRXXOhaFcWdQ4Dp9PMWNf4TFuBXwYxHqyYzA4dpPGG1Dc2yNca1ZSARmMjuTbqI2Kzk\n"

"L90TOKOIvdcnNRAsAj4PXIOBerydGbVllBu6U1HsBN9T5TdSujNLghJFHRhkuWt1A3vWN3\n"

"XZfsduC0pruyDpyRyZBVvLJSf8Dc3mBjNsOxo2ISIKg0CymMvJ1yWewYeg42icWfGnfHvC\n"

"s2fja2wLsinV6Oj7lW3sqN5uq5DXMHA3hzdRwtgXbtMSGBAJU2ubFJjPr9wccZ4hcH9VVt\n"

"lMN1Kis7Ux43rISXr7fB8rRIZuIsM8Yq8zPqrD50RBP5oGNuH1U4NPaO3ZsqIIN16PdcCl\n"

"8rWhv2DZ0BFVqLGrxXV9nG6Cqn4V8YdoeCUz5l62rnOOTz9EuARUaplO0DZwIAaR9nQkOg\n"

"r90A8ooHzTvE0Rmy6lv4iIY4yysBvPCPIOM1iMseSm7Ql2hfSERxTQIPjQ8EjUHVUAJ5TP\n"

"eiyG3bJaeLBYNjiCLersNTio8du7181PXseT9qhFaTOw1RojFTnfAIbm4c3uUalDeU4OOr\n"

"p16EU4vhvobukqq96i9UqxF9BIP7xXu5IBV6VKQuB3QzKmMWjAFFy3GnXGEw8QKNYGw2Uc\n"

"mhdCU7LdCyoMsavZHTHytclJxYrKBYMfVR14QW2agXP1CTMFxTSq0CfAG6FNYSvNjfdedG\n"

"dAlAvL8AIROMRgB7PTVGO1h5wcVCy0pC6gMhwH3RTg9v3eWfrOMKoXVq79L7H9YN6ZrhQA\n"

"EQoxQeLQRnJ9fKAPPDz9K0YtOva5sdsYPazSJItNDNU8YAV8d6TosoBcFS5552gATP5w0v\n"

"TeokrmxsdXlGbR7q0ESvlaa3h1NumNsqTF10rMIqIuZIkkU97qCEpSMb4BtTXqElUtPa3s\n"

"GMFKHeM09OgRDTWNV1hh0cKCpLB5Qc07NdBlB8TWKjbllNfSXw9Z7x7mRuZpFTuHM2Mxy4\n"

"6n6cDUoND2bD8pgwBEfnrYzZDiuWxmSd3zj44BsQQPeyHlTtajzKKIV1GjCobLysNTogXC\n"

"cirxR1EuerkgoRzZuF5CsPEdOFtJBJml8TV4Qey65lywTG6Wg4m2TN11ngmUaIaG94iuND\n"

"utIW8PXTUPflMMYVzLy5H1qezliZlklDoqzQgGE5Ab9CVNDgAz222WzNXTS2Oyc0apHQng\n"

"pudURefqUkSKW6dYVPnesjaRW8fezaEOBREGr799D8v1GIZCEXHyfoZQtcZOPScplwtJZF\n"

"gVARhbrC4XItpczzNLWH6US2LUBb0XDlUB8mesKj9kumx3RrWZqGSNyFwT06pYsRQNaolI\n"

"1htWxXaxXqGbn2FOlGgWH398i10LzqsYajHokJbxbcO7bq3iolRJ3OveKYgR7eDgSEJhf6\n"

"UBSSk9NHGMa0m2JRo7XYlMIuAGpObnPSSfqED3g1rCIpC97May0tPhPVHfOgZbYDld3mr9\n"

"gUIImOpwIgnIvmNtU4pRZDZaOK29WuojBmo9jlIkeLp0YonxriZN0d7WyBQNjzXR2j2jWl\n"

"S4sNrn6ZbKYz1TKQRbhiCgJWjUuWtf2BrxCLwtZo4mTX8lhCPYlEWk2JuwMnaGXHpB4JPw\n"

"OePFYL4pyUTRxciRU5UYoIRRcox35WfvjRzWheXJBuIN6jC0jVwbhhMyw7a1ewdF992Zjf\n"

"Kf2KLU9mH8M6zslWCdixTxVDWNJKUDKhdTvlPNYJyrnGsLFCQP4pU9ZZGrDDw5KLcVtg9D\n"

"dyzBBJygl8VZElPelujnv5Sw405mYJssye0i82QbHyCf9H0Kbxi74zdjmEvcCixlQ5T1s2\n"

"NIreKvTlvkD8ikth1ztB3G0jvphKh7rNRs4GoVSwqzYix35iTF10NULHwVBgE6fqpirOMb\n"

"3EBHL6N6USCJwPwZhAaToGdZFj0NCeukOzBSBAJ5rRVM6Ggamzmif8zjGosdKcuRNUh45O\n"

"AlkYqaYEQlPyMUbhxbTVku3OzhMnU5gtIaBtisDufXKVXwYWCRdfzADrfJNTl9hskxBzdL\n"

"OoQTJyPJz4M3t6eQZYbEGCVbxZymSAPryGXAbZM1TUk11nf1RUS5dhStQfrJvlWf59jRVr\n"

"GOuByymOlTOKSdIJJLHXVsIdfzj6owmkPvWH9oPMtClIsIe9luNMXTM6mflEfHrgrDeCuD\n"

"0AZvb1UMFOMb9S3KH4ZfcGx3N8LHLv0O9tycMseyBngbDRJa3dyx27RtgskwKEpYuEZ4Xq\n"

"ZGfnTJYdfLSdTuyq0eGF1V8ZYQnOOvQOBxbQvXwzA71poNGp6tXgI6NmoY7qGIL4lVuRj5\n"

"8OUvfRFnkkBgY7Vn9VSWow2eMFYay6TEc4MSGTuSWRk4eKIfPlpOvFdjYrg6tpWg9hA3SF\n"

"WBOeN4PI00D4ddRqwHE5SNF9vuAE9Y5bmGHar0leb5GdYiF1vc5KLZz1iE08PBGEV6T3SG\n"

"Ao1aRjW7u94iHvImdfiA92wnhJcsWhu3gJk56HjGQrOkn7X9AMGzW9oOgYLm3Hzm2ohtLK\n"

"ZaVUg2NyyAjFxq78FDw6P6LW94MvDQoFQZ0Msa1tpzwGGVeAcnaFcwhhQQmkdnJZVKS3Ku\n"

"r2EmIXxuAJYp18GqOOx050wWMPSnsoMBeyo7JCmMN8GETx4a9LNLDYGbfEn9t8uDWvjZb0\n"

"skO70dzk3VHdYkhNfhmQZQgAAKMH7fiD43nNmEFRpS44xUQkYkYsPz4y3tToNeFQEZzkZY\n"

"oQNfHLpIcztBZgPwkjyIDAxkU2hLdRpRGYY2jAXUXfNnNeHJoQ4CecaweaH51hhi3nCWcX\n"

"GUZ5o9WI2uK6CyrVc4CNRGWAchVIDMRovkBaxWABOzGOdnTOQct7BaexmmDjIcjVZCihfd\n"

"qOcIsokuxV1oxm9fedFNPcdfhbplqkkYJqQ5JMflwDCAgXqXzqbXHNqmQpNEQMAyuxriS1\n"

"RgF6l24SDawEIX8ExoBD92wZkxWA8ayfD5Quwmt97cIblY0P4hm1NhJvmA62uMEqNkptE4\n"

"Ekxgw0TejYYDv99zJtQ90pObpXwd6WP6ROu85XkkR3SWXHotZ9b5rGbCU34Ma7rqM9fGqg\n"

"MYgQ6PC5QKVtkm8ABuM4xJXX9K1ZEsl6eAc4xOMNtpw4UTEdyy8BuEwxtmGaSKR5frTiHR\n", "BuEwxtmGaSKR5frTiHR");

//testing exactly 90 characters in crib after wash

runtest("abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm abcdef kckvm \n", "silent alarm silent alarm silent alarm silent alarm silent alarm silent alarm silent alarm 0827");

// testing multiple newline characters in crib

runtest("silent alarm\n", " \n\n\nsilent \n\n\nalarm \n\n\n");