

University of Central Punjab Faculty of Information Technology

Project:

A special camera is launched into Mars' orbit to identify and label "points of interest" (POI for short), which are essentially pattern files (smaller matrix), within the source image file for scientific research. The camera produces black and white images in varying resolutions in the form of large matrices with integer values between 0 and 255. You must create a program that is capable of reading a file containing such an image and search it for any POI provided by the user through a file as well. Once a POI has been identified, you must mark it by inverting its values (make them negative e.g. 155 becomes -155). Finally after all such POI's have been found and marked, output the entire matrix to a new file specified by user.

Your program must adhere to the following guidelines:-

- a) Must be dynamic to allow for files of any size to be processed
- b) Store matrices using appropriate structs (with appropriate initialization functions)
- c) No memory leakage
- d) Allow user to choose multiple POI files to mark for a single source image file
- e) You can assume number of columns of the matrix is provided at the start of the file (but no row count data)

Bonus functionality:

- a) POIs may be rotated either 90, 180, or 270 degrees
- b) POIs may overlap
- c) Neither row nor column data is provided in the file

Sample run:

Input file:

101		(122	-	70	101	202		240		40	254		
101	175	123	52	78	184	202	8	219	15	49	254	86	62
156	213	41	200	123	131	252	186	108	116	39	205	243	120
218	239	201	109	52	173	244	58	185	18	64	209	165	222
81	136	247	149	183	206	164	214	179	121	176	200	89	128
61	224	221	103	114	170	88	133	117	30	104	46	19	151
99	111	41	180	188	1	32	27	100	241	97	56	22	10
26	238	248	155	154	106	123	52	78	66	125	225	193	190
54	127	159	3	34	152	41	200	123	146	94	90	195	199
105	191	212	23	194	207	201	109	52	231	95	250	150	55
51	227	204	36	80	253	160	215	153	236	124	71	98	<u> 75</u>
249	47	168	13	211	59	87	240	228	68	63	218	156	101
148	141	91	115	40	38	197	137	83	123	158_	239	213	175
157	129	12	5	147	109	33	134	77	144	101	175	123	<u>123</u>
135	177	9	45	123	52	78	118	73	232	156	213	123	182
167	162	96	174	41	200	123	11	16	122	218	239	175	161
139	132	44	25	201	109	52	201	216	78	142	138	84	234
72	4	29	52	119	79	42	233	235	140	112	187	21	198

- Green boxes highlight non bonus example where submatrices are always oriented straight up and never overlap
- Blue boxes highlight bonus example where top blue matrix has been rotated by 90 degrees clockwise and bottom blue matrix is oriented straight up
- Yellow portion highlights overlap occurring between the two blue boxes

POI file (non-bonus):

POI file (with-bonus):

101	175	123
156	213	123
218	239	175
		_, _

Output file:

175	-123	-52	-78	184	202	8	219	15	49	254	86	62
213	-41	-200	-123	131	252	186	108	116	39	205	243	120
239	-201	-109	-52	173	244	58	185	18	64	209	165	222
136	247	149	183	206	164	214	179	121	176	200	89	128
224	221	103	114	170	88	133	117	30	104	46	19	151
111	41	180	188	1	32	27	100	241	97	56	22	10
238	248	155	154	106	-123	-52	-78	66	125	225	193	190
127	159	3	34	152	-41	-200	-123	146	94	90	195	199
191	212	23	194	207	-201	-109	-52	231	95	250	150	55
227	204	36	80	253	160	215	153	236	124	71	98	75
47	168	13	211	59	87	240	228	68	63	-218	-156	-101
141	91	115	40	38	197	137	83	123	158	-239	-213	-175
129	12	5	147	109	33	134	77	144	-101	-175	-123	-123
177	9	45	-123	-52	-78	118	73	232	-156	-213	-123	182
162	96	174	-41	-200	-123	11	16	122	-218	-239	-175	161
132	44	25	-201	-109	-52	201	216	78	142	138	84	234
4	29	52	119	79	42	233	235	140	112	187	21	198
	213 239 136 224 111 238 127 191 227 47 141 129 177 162 132	213 -41 239 -201 136 247 224 221 111 41 238 248 127 159 191 212 227 204 47 168 141 91 129 12 177 9 162 96 132 44	213	213 -41 -200 -123 239 -201 -109 -52 136 247 149 183 224 221 103 114 111 41 180 188 238 248 155 154 127 159 3 34 191 212 23 194 227 204 36 80 47 168 13 211 141 91 115 40 129 12 5 147 177 9 45 -123 162 96 174 -41 132 44 25 -201	213 -41 -200 -123 131 239 -201 -109 -52 173 136 247 149 183 206 224 221 103 114 170 111 41 180 188 1 238 248 155 154 106 127 159 3 34 152 191 212 23 194 207 227 204 36 80 253 47 168 13 211 59 141 91 115 40 38 129 12 5 147 109 177 9 45 -123 -52 162 96 174 -41 -200 132 44 25 -201 -109	213 -41 -200 -123 131 252 239 -201 -109 -52 173 244 136 247 149 183 206 164 224 221 103 114 170 88 111 41 180 188 1 32 238 248 155 154 106 -123 127 159 3 34 152 -41 191 212 23 194 207 -201 227 204 36 80 253 160 47 168 13 211 59 87 141 91 115 40 38 197 129 12 5 147 109 33 177 9 45 -123 -52 -78 162 96 174 -41 -200 -123 132 44 25 -201 -109 -52	213 -41 -200 -123 131 252 186 239 -201 -109 -52 173 244 58 136 247 149 183 206 164 214 224 221 103 114 170 88 133 111 41 180 188 1 32 27 238 248 155 154 106 -123 -52 127 159 3 34 152 -41 -200 191 212 23 194 207 -201 -109 227 204 36 80 253 160 215 47 168 13 211 59 87 240 141 91 115 40 38 197 137 129 12 5 147 109 33 134 177 9 45 -123 -52 -78 118 162 96 174 -	213 -41 -200 -123 131 252 186 108 239 -201 -109 -52 173 244 58 185 136 247 149 183 206 164 214 179 224 221 103 114 170 88 133 117 111 41 180 188 1 32 27 100 238 248 155 154 106 -123 -52 -78 127 159 3 34 152 -41 -200 -123 191 212 23 194 207 -201 -109 -52 227 204 36 80 253 160 215 153 47 168 13 211 59 87 240 228 141 91 115 40 38 197 137 83 129 12 5 147 109 33 134 77	213 -41 -200 -123 131 252 186 108 116 239 -201 -109 -52 173 244 58 185 18 136 247 149 183 206 164 214 179 121 224 221 103 114 170 88 133 117 30 111 41 180 188 1 32 27 100 241 238 248 155 154 106 -123 -52 -78 66 127 159 3 34 152 -41 -200 -123 146 191 212 23 194 207 -201 -109 -52 231 227 204 36 80 253 160 215 153 236 47 168 13 211 59 87 240 228 68 141 91 115 40 38 197 137 <	213 -41 -200 -123 131 252 186 108 116 39 239 -201 -109 -52 173 244 58 185 18 64 136 247 149 183 206 164 214 179 121 176 224 221 103 114 170 88 133 117 30 104 111 41 180 188 1 32 27 100 241 97 238 248 155 154 106 -123 -52 -78 66 125 127 159 3 34 152 -41 -200 -123 146 94 191 212 23 194 207 -201 -109 -52 231 95 227 204 36 80 253 160 215 153 236 124 47 168 13 211 59 87 240 228 <t< td=""><td>213 -41 -200 -123 131 252 186 108 116 39 205 239 -201 -109 -52 173 244 58 185 18 64 209 136 247 149 183 206 164 214 179 121 176 200 224 221 103 114 170 88 133 117 30 104 46 111 41 180 188 1 32 27 100 241 97 56 238 248 155 154 106 -123 -52 -78 66 125 225 127 159 3 34 152 -41 -200 -123 146 94 90 191 212 23 194 207 -201 -109 -52 231 95 250 227 204 36 80 253 160 215 153 236 124 71 <td>213 -41 -200 -123 131 252 186 108 116 39 205 243 239 -201 -109 -52 173 244 58 185 18 64 209 165 136 247 149 183 206 164 214 179 121 176 200 89 224 221 103 114 170 88 133 117 30 104 46 19 111 41 180 188 1 32 27 100 241 97 56 22 238 248 155 154 106 -123 -52 -78 66 125 225 193 127 159 3 34 152 -41 -200 -123 146 94 90 195 191 212 23 194 207 -201 -109 -52 231 95 250 150 227 204 36 <td< td=""></td<></td></td></t<>	213 -41 -200 -123 131 252 186 108 116 39 205 239 -201 -109 -52 173 244 58 185 18 64 209 136 247 149 183 206 164 214 179 121 176 200 224 221 103 114 170 88 133 117 30 104 46 111 41 180 188 1 32 27 100 241 97 56 238 248 155 154 106 -123 -52 -78 66 125 225 127 159 3 34 152 -41 -200 -123 146 94 90 191 212 23 194 207 -201 -109 -52 231 95 250 227 204 36 80 253 160 215 153 236 124 71 <td>213 -41 -200 -123 131 252 186 108 116 39 205 243 239 -201 -109 -52 173 244 58 185 18 64 209 165 136 247 149 183 206 164 214 179 121 176 200 89 224 221 103 114 170 88 133 117 30 104 46 19 111 41 180 188 1 32 27 100 241 97 56 22 238 248 155 154 106 -123 -52 -78 66 125 225 193 127 159 3 34 152 -41 -200 -123 146 94 90 195 191 212 23 194 207 -201 -109 -52 231 95 250 150 227 204 36 <td< td=""></td<></td>	213 -41 -200 -123 131 252 186 108 116 39 205 243 239 -201 -109 -52 173 244 58 185 18 64 209 165 136 247 149 183 206 164 214 179 121 176 200 89 224 221 103 114 170 88 133 117 30 104 46 19 111 41 180 188 1 32 27 100 241 97 56 22 238 248 155 154 106 -123 -52 -78 66 125 225 193 127 159 3 34 152 -41 -200 -123 146 94 90 195 191 212 23 194 207 -201 -109 -52 231 95 250 150 227 204 36 <td< td=""></td<>