

Забележка:

Всички задачи се слагат в папка със софтуни името ви и се изпращат като домашна!

Задача 1:

В терминала на програмата ви се подават 255 стойности от 0 до 255 в неопределен ред разделни само със ' '(space). Целта на програмата ви е да приемете тези 255 стойности, и да изпратите стойността, която се повтаря най-много пъти. Ако са няколко, изпращате най-голямата по бройна стойност (примерно ако се повтарят по равен брой пъти 7 , 190, 240 трябва програмата ви да върне 240).

Пример 1:

Input = output – формат

61 60 153 63 11 164 230 75 21 119 68 11 68 128 51 86 227 7 89 184 150 58 72 120 48 91  
194 130 42 1 58 105 206 183 55 22 197 20 15 72 225 127 231 125 114 112 103 62 75 127  
138 76 58 120 179 246 234 132 125 155 154 41 157 149 116 112 100 215 83 191 154 173  
7 89 149 72 185 127 207 192 252 240 64 148 63 144 112 65 103 43 153 61 93 142 69 90  
113 48 180 114 12 244 169 90 112 139 18 96 207 168 62 33 121 253 202 39 75 82 131 140  
93 169 115 153 229 165 3 203 11 206 26 157 220 34 139 203 148 255 31 215 160 247 131  
157 53 233 163 243 147 241 227 203 167 49 133 145 160 111 21 148 207 173 2 2 203 102  
255 153 228 191 116 158 135 83 33 8 229 111 212 110 123 8 87 192 175 106 110 233 65  
229 113 111 164 227 8 17 183 22 180 66 60 229 75 56 218 136 181 62 254 215 122 26 37  
78 99 219 79 223 95 218 198 223 147 55 248 97 2 209 188 138 145 154 88 50 196 159 189  
30 226 202 59 177 169 203 2 214 176 138 166 43 101 245 3 187 100 = 203

19 213 155 17 137 136 122 245 138 110 158 115 5 118 223 202 175 32 174 46 154 83 28  
131 180 25 35 137 53 40 182 87 115 213 12 189 126 249 211 58 249 98 109 7 230 225 98  
97 35 168 27 76 138 179 9 130 219 202 128 109 71 0 191 38 193 109 66 13 230 92 201 15  
166 107 174 82 205 96 173 191 169 252 97 255 0 255 93 178 83 249 11 107 192 66 39 161  
246 184 25 246 137 169 248 88 20 18 253 224 203 105 81 92 130 218 151 155 202 11 186  
121 198 184 178 70 174 162 13 160 185 112 42 171 78 38 115 201 164 208 122 219 103  
96 166 9 35 218 158 13 203 75 184 159 141 157 224 153 208 116 93 243 164 214 203 22  
152 10 168 148 78 120 54 241 229 110 70 2 213 120 167 132 126 210 50 145 70 55 166 9  
50 197 153 170 53 107 240 52 46 198 96 239 133 161 80 121 166 142 245 237 4 114 136  
252 244 131 207 38 244 179 177 132 133 18 153 117 146 178 184 44 236 206 85 163 217  
253 28 61 129 115 170 34 54 198 15 165 202 128 91 58 71 117 138 216 213 147 97 = 213

160 207 192 120 29 201 61 118 72 133 177 32 138 220 242 184 100 234 182 191 56 114  
172 21 148 54 37 150 9 123 142 1 2 5 8 28 178 73 11 205 85 224 216 184 236 21 20 168  
145 25 93 46 138 177 32 41 132 205 58 181 217 68 166 221 239 65 84 228 102 67 85 178  
70 9 122 209 254 182 22 157 15 156 217 125 189 30 226 159 216 45 111 125 249 44 93 79  
135 191 88 51 21 107 32 22 206 46 45 159 79 119 120 36 172 137 158 143 79 167 164 26  
158 250 87 185 147 156 46 254 49 188 196 189 142 88 252 118 176 252 64 26 203 240  
233 193 194 28 144 204 5 158 3 221 128 141 201 19 27 189 151 164 155 115 204 223 188  
209 154 150 42 66 85 99 206 206 92 62 201 105 75 93 71 198 10 246 122 185 99 185 230  
176 72 92 215 135 149 182 69 153 143 150 150 135 102 123 242 172 75 114 116 183 191  
104 55 85 8 177 190 0 53 67 9 125 73 57 6 31 46 20 177 54 211 112 226 91 117 102 175  
64 187 33 223 1 1 252 49 182 227 219 217 97 3 230 127 29 26 = 182

20 143 161 44 55 48 185 19 58 100 165 109 53 197 75 79 114 206 149 240 199 191 21 163  
 62 157 122 82 190 42 227 243 38 77 252 47 169 57 250 2 181 238 81 109 252 2 125 134  
 78 8 100 36 157 83 224 245 139 18 45 53 226 164 199 242 45 72 119 207 24 73 130 135  
 199 164 202 228 240 170 59 4 228 218 250 205 125 18 111 95 119 17 120 67 53 151 125  
 145 126 192 17 105 201 86 127 168 83 174 78 250 185 100 77 129 214 151 241 103 242  
 241 134 100 190 193 12 134 207 42 83 138 172 73 40 207 110 109 39 99 54 107 211 24  
 205 77 36 139 142 146 228 130 1 34 110 135 79 16 238 66 181 242 87 136 82 145 14 175  
 247 103 11 38 236 153 251 51 13 65 109 147 99 118 145 231 43 214 15 179 237 14 6 44  
 181 182 170 148 167 76 46 10 131 97 26 57 30 132 110 20 53 39 97 44 191 183 235 58  
 233 78 252 176 121 33 218 154 1 46 212 20 15 172 98 134 90 176 173 148 150 205 57 174  
 71 134 255 152 238 208 157 240 105 116 103 78 35 33 87 18 92 74 102 = 134

11 6 22 254 230 183 181 19 9 226 204 184 204 174 35 19 174 135 105 144 231 200 237  
 240 88 193 235 111 0 145 165 28 107 107 37 224 153 43 156 217 155 9 204 63 107 132 67  
 213 165 36 37 227 142 113 76 54 41 110 93 82 210 2 244 224 161 135 0 233 78 159 9 47  
 92 184 16 219 109 144 252 121 154 14 69 167 125 121 143 26 50 84 252 245 133 51 120  
 176 50 61 141 107 63 4 143 62 141 162 168 217 196 51 236 254 195 36 239 238 176 190  
 57 42 164 34 114 200 1 181 125 9 53 176 119 67 148 151 53 109 1 175 104 57 127 34 42  
 181 191 63 59 171 195 254 220 88 142 66 108 29 143 245 206 229 121 201 164 209 254  
 96 182 55 39 83 205 112 251 123 152 70 151 97 159 131 81 175 207 243 84 29 134 117 83  
 102 63 103 206 250 156 254 31 139 140 45 9 46 239 222 173 153 252 191 4 149 49 132  
 188 72 240 186 119 210 252 123 69 216 223 230 22 153 151 156 227 143 43 108 6 35 156  
 207 80 233 62 20 176 203 145 18 255 221 244 136 87 181 200 229 158 209 133 = 254

96 186 15 203 3 72 21 192 106 76 115 96 175 199 201 126 199 153 204 134 61 112 115  
 183 9 184 64 36 50 102 174 91 36 185 222 91 24 177 6 156 139 0 36 140 174 51 77 242  
 113 233 9 248 69 124 210 24 145 123 229 101 132 107 249 12 139 211 41 185 228 184  
 168 209 10 204 11 120 59 220 164 175 40 159 227 131 147 59 4 58 228 169 231 220 137  
 52 133 24 4 16 255 21 37 54 132 21 99 2 203 102 187 12 152 130 163 165 31 27 241 74  
 146 70 222 176 157 139 200 96 223 112 53 79 95 181 231 145 251 68 141 231 199 78 103  
 177 150 178 81 73 53 233 110 216 243 37 10 47 47 239 38 242 107 244 65 97 117 200 198  
 221 57 223 220 36 81 73 218 141 67 1 253 177 128 90 18 226 62 96 55 210 116 77 144  
 179 106 177 153 192 179 90 48 94 153 248 13 216 166 191 169 173 34 19 70 220 119 170  
 23 95 53 202 61 188 55 223 154 36 171 130 24 188 152 52 113 81 0 91 122 249 44 168  
 129 17 147 57 58 10 91 204 242 225 104 84 233 18 212 152 152 198 5 = 36

## Задача 2:

Непослушните ученици от 9 клас в Механотехникума в Бургас редовно не посещават часовете си по програмиране и математика в резултат на това те не са подготвени за предстоящото контролно по математика на тема квадратни уравнения. Вие, като хора, които редовно са посещавали часовете по математика и курсовете по C++ в СофтУни трябва да бъдете добри и да им напишете програма за смятане на квадратни уравнения. Като вход в програмата ще получавате числа 3 числа –  $a, b, c$ , които са коефициентите на квадратното уравнение (т.е. квадратното уравнение ще изглежда като  $ax^2 + bx + c$ , като  $a, b, c$  могат да имат знак минус при въвеждането от клавиатурата). Числата на вход са разделени със ' ' (space) и винаги са 3 на брой. В отговора вие трябва да кажете, кои са корените, които удовлетворяват уравнението с точност до втория знак след десетичната запетая. Ако уравнението има един корен, този корен се изписва 2 пъти. Ако уравнението няма решение се изписва nan.

## Пример 2:

-9 16 5 = -0.27,2.05  
18 -11 15 = nan,nan  
-42 -32 -32 = nan,nan  
31 11 -33 = 0.87,-1.22  
-33 -17 46 = -1.47,0.95  
-33 21 1 = -0.04,0.68  
-50 21 -37 = nan,nan  
-37 -38 1 = -1.05,0.03  
-37 -35 24 = -1.41,0.46  
-3 -35 4 = -11.78,0.11  
43 -35 -36 = 1.41,-0.59  
-25 11 -5 = nan,nan  
-12 16 13 = -0.57,1.90  
-29 -43 -44 = nan,nan  
-16 36 -39 = nan,nan

### Задача 3:

В края на лятото и по точно през месец август повечето хора ходят на почивки по морета, океани, басейни или в локвите образували се в незапълнените неравности на пътя пред техните домове. В качеството си на бъдещи програмисти преминали успешно курса по програмиране при Мартин Куванджиев, вие сте помолени да помогнете на всички тези хора да намерят перфектното за тях място за почивка, за целта ще бъде нужно да направите програма, която да описва местата за почивка в няколко отделни държави.

Опишете в клас различните отделни места за почивка, като класа ви трябва да съдържа следните параметри:

- Име на мястото
- Координати
- Тип на терена (пясък, скали, плочки, асфалт(или каквото е останалот от него),смесен тип, друго.
- Тип на водния басейн (море, океан, басейн, локва, няма, друг).
- Оценка на чистотата на терена (от 0 до 10)

- Оценка на чистотата на водния басейн (от 0 до 10)
- Брой посетители годишно
- Средни цени за ден престой (в местна валута)
- Брой барове около мястото
- Цена на малиновото дайкири (в местна валута)
- Цена на мохитото (в местна валута)
- Цена на портокаловия сок (в местна валута)

Направете клас - държава, в който да се съдържат всички плажове, които са налични за дадена държава. Класа трябва да съдържа следните параметри:

- Име на страната
- Плажове
- Валута на страната
- Разменен курс на валутата спрямо еврото

И да има следните функции:

- Добавяне на ново място за почивка
- Премахване на вече налично място за почивка на база име на мястото (по желание)
- Търсене на всички места на база на поне 3 параметъра от параметрите на класа място за почивка (вие си избирате на база на кои параметри ще направите търсенето, като пример може да търсите само по най-важните – цената на портокаловия сок, оценка на чистотата на водата, средни цени за ден престой).

Изисквания за задачата:

- Търсенето на места за почивка трябва да се извършва асинхронно(едновременно) в различните държави. Трябва да бъде невъзможно да се добавя или премахва ново място за почивка, по време на търсене в държавите. Когато бъде намерено съвпадение по критериите за търсене се извежда пълната информация за мястото на почивка (всичките членове на класовете).