

SPRAWOZDANIE IV

Dotyczy : **Problemu mapowania**

1. Opis algorytmu:

Program wczytuje multizbiór, a następnie wyszukuje dwóch największych elementów w nim.

Następnie porównuje licznosc zbioru ze wzorem $(n(n-1))/2$ - jeżeli obie wartości są sobie równe, to program przechodzi do funkcji *szukaj()*, która do naszego rozwiązanie dodaje element multizbioru, następnie sprawdza wszystkie możliwe sumy elementów naszego obecnego rozwiązania, jeżeli sumy nie ma, program szuka innego rozwiązania, aż do wypełnienia tablicy z rozwiązaniem.

Jak zostało znalezione rozwiązanie to algorytm tworzy na podstawie rozwiązania osobny multizbiór, następnie sortuje oba multizbiory, jeżeli rozmiary multizbiorów są takie same to porównuje elementy, jeżeli wszystkie są takie same, znaleźliśmy rozwiązanie.

2. Zawartość instacji i wyniki testów:

1. Instacja 1.

A: 1 3 3 4 6 7 43 46 49 50 71 83 126 129 132 133 154 197 200 203 204

rozwiązanie : 1 3 3 43 83 71

czas trwania to: 0.001

2. Instacja 2.

A: 1 1 1 1 1 1 1 2 2 2 2 2 3 3 3 3 4 4 4 5 5 5 5 6 6 6 6 7 8 9 9 10 11 12 15 15 16 17 18 19 20 21 21 22 23 24 25 26
27 40 45 46 46 51 52 55 60 61 61 62 63 64 65 66 66 67 67 67 68 68 69 69 70 70 71 71 72 72 73

rozwiązanie : 1 1 1 1 1 1 6 9 6 40 5 1

czas trwania to: 0.95

3. Instacja 3.

A: 1 1 1 1 1 2 2 2 3 3 3 3 3 4 4 4 4 5 5 5 5 6 6 7 7 7 8 8 8 8 9 9 9 10 11 11 12 12 13 14 14 14 15 16 17
80 83 87 88 90 91 92 94 95 96 97 104 170 173 177 178 181 182 184 184 185 186 187 187 191 192
195 196 198 199 200 201

rozwiązanie : 1 1 1 2 1 3 1 4 3 80 90 14

czas trwania to: 11.255

4. Instancja ins-PDP-11a-asc.txt

A: 2 3 4 4 5 5 6 6 6 7 7 8 8 9 10 11 11 11 11 12 13 13 14 14 15 15 16 16 17 18 19 19 19 20 20 21 21 22
23 23 25 25 25 26 26 27 27 27 30 30 31 31 32 32 35 35 36 36 38 38 38 40 41 42 42 43 44 46 46 49 51
52 53 57 57 61 63 67

rozwiązanie : 4 6 5 8 3 9 5 2 4 7 8 6

czas trwania to: 25.891

5. Instancja ins-PDP-12a-asc.txt

A: 2 3 4 4 5 5 6 6 6 6 7 7 8 8 9 10 11 11 11 11 12 12 13 13 14 14 15 15 16 16 17 18 19 19 19 20 20 20
21 21 22 23 23 25 25 25 26 26 27 27 27 27 30 30 31 31 31 32 32 33 35 35 36 36 38 38 38 38 40 41 42
42 43 44 46 46 47 49 50 51 52 53 57 57 58 61 63 63 67 69 73

rozwiązanie : 4 6 5 8 3 9 5 2 4 7 8 6 6

czas trwania to: 376.68 s

6. Instancja ins-PDP-13a-asc.txt

A: 2 3 3 4 4 5 5 6 6 6 6 7 7 8 8 9 9 10 11 11 11 11 12 12 13 13 14 14 15 15 15 16 16 17 18 19 19 19 20
20 20 21 21 22 23 23 23 25 25 25 26 26 27 27 27 27 30 30 30 31 31 31 32 32 33 34 35 35 36 36 36 38
38 38 38 40 41 41 42 42 43 44 46 46 47 49 50 50 51 52 53 53 57 57 58 61 61 63 63 66 67 69 72 73 76
rozwiązanie : 3 6 6 8 7 4 2 5 9 3 8 5 6 4 4

czas trwania to: 2075.28

7. Instancja ins-PDP-11b-asc.txt

A: 12 13 15 25 27 35 38 42 47 48 54 57 60 66 66 74 79 82 89 93 95 101 104 107 108 112 114 123 135
136 139 146 150 151 161 167 171 184 186 193 196 199 205 209 211 212 224 228 239 241 247 250
253 259 262 265 286 294 307 307 313 319 321 334 346 351 360 373 395 398 400 408 420 433 446
458 474 512

rozwiązanie : 38 74 27 66 42 15 89 47 35 13 12 54

czas trwania to: 0.006

8. Instancja ins-PDP-11b-desc.txt

A: 512 474 458 446 433 420 408 400 398 395 373 360 351 346 334 321 319 313 307 307 294 286 265
262 259 253 250 247 241 239 228 224 212 211 209 205 199 196 193 186 184 171 167 161 151 150
146 139 136 135 123 114 112 108 107 104 101 95 93 89 82 79 74 66 66 60 57 54 48 47 42 38 35 27
25 15 13 12

rozwiązanie : 38 74 27 66 42 15 89 47 35 13 12 54

czas trwania to: 0.004

9. Instancja ins-PDP-14b-desc.txt

A: 673 619 607 594 594 559 540 537 528 515 512 512 483 480 474 471 458 458 446 433 433 423 423
420 408 408 400 398 395 376 373 366 360 351 346 344 334 329 321 319 313 307 307 300 294 287
287 286 273 272 265 262 259 253 250 247 241 239 230 228 224 221 212 211 209 205 199 199 196
194 193 186 184 171 167 164 161 161 151 150 146 139 137 136 136 135 123 120 114 112 108 107
104 101 95 93 89 82 82 79 79 74 66 66 63 60 57 57 54 48 47 42 38 35 27 25 25 15 13 12

rozwiązanie : 54 12 13 35 47 89 15 42 66 27 74 38 25 57 79

czas trwania to: 0.002

3. Wnioski:

- a. Im więcej miejsc restrykcyjnych tym wolniej program znajduje rozwiązanie - potrzebuje na to więcej czasu
- b. Jeśli dana instancja zawiera fragmenty o niskich wartościach fragmentów, wtedy program wyszukuje rozwiązanie w krótszym czasie.