Documentație aplicație

Note:

-inputul pentru lista trebuie dat cu o virgula între numere, fără spațiu la final

Rezultatul va fi afisat, dar funcția pentru fiecare problema va afișa indicele de unde începe secvența maxim + lungime.

Problema 9:

Aplicația pentru problema 9 caută secvența cu lungime maximă unde p=1 sau în oricare trei elemente consecutive există o valoare care se repetă.

|  |  |
| --- | --- |
| Input | Rezultat |
| 7 [1,2,1,2,3,4,3] | 0, 4 (4, [1,2,1,2]) |
| 10 [7,17,8,17,5,17,10,1,2,5] | 1, 3 (3, [17,8,17]) |
| 10 [7,17,8,17,5,17,17,17,17,17] | 5, 5 (5 [17,17,17,17,17]) |
| 7 [1,3,4,5,7,7,7] | 4, 3 (3 [7,7,7]) |
| 5 [0,0,0,0,0] | 0, 5 (5, [0,0,0,0,0]) |
| 2 [1,3] | -1 (Mesaj special) |

Pentru cazul 8 [1,2,1,2,3,2,4,3] rezultatul va fi același cu primul caz de testare deoarece conform cerinței în secvența maximă oricare trei numere consecutive trebuie să aibă o valoare care se repetă, deci secvența [1,2,1,2,3,2] nu este o soluție deoarece în subsecvența [1,2,3] nu se repetă nicio valoare.

Scenariu de rulale pentru secventa [1,2,1,2,3,4,3] de lungime 7:

* Start aplicatie principala
* Apasare buton problema 9
* Start aplicatie pentru problema 9
* Entry\_1 = 7 de la utilizator (entry label)
* Entry\_2 = 1,2,1,2,3,4,3 de la utilizator(entry label)
* Apasare buton executare
* Se incearca punerea valorilor date prin intermediul celor doua entry label-uri in 2 variabile
* Daca datele sunt introduse corect: nr = 7, sir\_numere = 1,2,1,2,3,4,3
* Se apeleaza functia sec\_3numere\_consec(nr, sir\_numere)
* Variabilele lungime, start\_sec, lungime\_aux, start\_sec\_aux sunt setate la 0
* i = 0 se verifica conditia data pentru [1,2,1] ce este indeplinita, atunci start\_sec\_aux = 0, lungime\_aux = 3
* lungime\_aux > lungime, deci lungime = 3 si start\_sec = 0
* i = 1 se verifica conditia pentru [2,1,2] ce este indeplinita, atunci lungime\_aux = 4
* lungime\_aux > lungime, deci lungime = 4 si start\_sec = 0
* i = 2 se verifica conditia pentru [1,2,3] ce nu este indeplinita, asa ca lungime\_aux, start\_sec\_aux sunt setate la 0
* i =3 se verifica conditia pentru [2,3,4] ce nu este indeplinita, asa ca lungime\_aux, start\_sec\_aux sunt setate la 0
* i =4 se verifica conditia pentru [3,4,3] ce este indeplinita, atunci start\_sec\_aux = 4, lungime\_aux = 3
* secventa maxima a fost gasita, sub forma a doi indici, inceputul secventei si lungimea
* indicii dati sunti pusi in alte doua variabile , text1 respectiv text2 pentru a fi afisate; text1 = „Lungimea secventei: 4” text2 = „Secventa: [1,2,1,2]”
* in labelul de output, numit output\_label sunt concatenate text1 si text2 ce vor aparea pe ecran
* apasare buton iesire din aplicatia pentru problema 9
* apasare buton iesire din aplicatia principala

Problema 10:

Aplicația pentru problema 10 caută secvența cu lungime maximă unde p=1 sau diferentele (x[j+1] - x[j]) si (x[j+2] - x[j+1]) au semne contrare, pentru j=i..i+p-2.

|  |  |
| --- | --- |
| Input | Rezultat |
| 7 [1,2,1,2,3,4,3] | 0, 4 (4, [1,2,1,2]) |
| 10 [1,2,1,2,5,3,6,5,8,4] | 3, 7 (7, [2,5,3,6,5,8,4]) |
| 5 [0,0,0,0,0] | -2 (Mesaj special) |
| 5 [-1,2,7,3,2] | 1, 3 (3 [2,7,3]) |
| 5 [-1,2,7,8,5] | 2, 3 (3 [7,8,5]) |
| 2 [1,3] | -1 (Mesaj special) |

Scenariu de rulare pentru secventa [-1,2,7,8,5] de lungime 5:

* start aplicatie principala
* apasare buton problema 10
* start aplicatie pentru problema 10
* entry\_1 = 5 de la utilizator (entry label)
* entry\_2 = -1,2,7,8,5 de la utilizator (entry label)
* datele introduse sunt puse in doua variabile nr si lista
* este apelata min\_max\_local(nr, lista)
* start\_sec, lungime, lungime\_aux, start\_sec\_aux sunt setate la 0
* i=0 se verifica daca lista[1] == 2 este minimul sau maximul local intre vecinii lui lista[0] == -1 sau lista[2] == 7. Conditia nu este indeplinita, asa ca start\_sec\_aux = 0, lungime\_aux = 0
* i=1 se verifica daca lista[2] == 7 este minimul sau maximul local intre vecinii lui lista[1] == 2 sau lista[3] == 8. Conditia nu este indeplinita, asa ca start\_sec\_aux = 0, lungime\_aux = 0
* i=2 se verifica daca lista[3] == 8 este minimul sau maximul local intre vecinii lui lista[2] == 7 sau lista[4] == 5. Conditia este indeplinita, asa ca start\_sec\_aux = 2 si lungime\_aux = 3
* lungime\_aux > lungime, asa ca lungime = 3 si start\_sec\_aux = 2
* text\_1 = „Lungimea secventei: 3”
* text\_2 = „Secventa: [7,8,5]”
* result\_label(output pentru label) = „Lungimea secventei: 3

Secventa: [7,8,5]”

* apasare buton iesire din aplicatia pentru problema 10
* apasare buton iesire din aplicatia principala