1. Dan je seznam, ki vsebuje premi pregled dvojiškega drevesa. Sestavi algoritem, ki ugotovi, če je to drevo lahko iskalno. Npr. za sezname [8, 4, 2, 1, 6, 5, 10, 9, 30, 11], [1, 2, 3], [8, 10, 9, 30, 11], [3, 2, 1], [3, 1, 2], vrne True, za sezname [8, 4, 10, 2], [8, 4, 2, 1, 6, 5, 10, 7, 30, 11], [1, 2, 4, 1] pa False.
2. Dan je naraščajoče urejen seznam (dolžine , v katerem hranimo **cela** števila. Poišči algoritem, katerega časovna zahtevnost **v najboljšem primeru** je in **v najslabšem primeru** , ki ugotovi, ali v seznamu obstaja tak , da je . Algoritem podrobno opiši. Za sezam [-12, -3, -1, 2, 3, 4, 6, 8] je odgovor True (za i = 6), za [-12, 14, 15, 16, 27] pa False.

1. Miha se je brezupno zaljubil v Alenko, ki pa mu njegovih pozornosti ne vrača. Zato se je odločil, da si bo pridobil novih spretnosti in jo tako očaral. Naredil si je seznam spretnosti, za katere misli, da se jih lahko nauči. Poleg vsake je zapisal tudi predviden čas učenja spretnosti (v dnevih) ter oceno, koliko naklonjenosti si bo z njo pridobil pri Alenki. Pomagaj mu in določi, katerih spretnosti naj se uči, če ima le **še 24 dni** časa do konca šolskega leta (potem Alenke lep čas spet ne bo videl). Upoštevaj tudi, da se spretnosti ne da naučiti le 'na pol': naučiti se mora cele ali pa ne bo učinka. Ker Miha ni polovičar, ga seveda učenje posamezne spretnosti zasede za cel dan (na posamezen dan se torej uči le ene spretnosti).

|  |  |  |
| --- | --- | --- |
| Ime spretnosti | Čas učenja (v dnevih) | Število točk |
| Igranje orglic | 12 | 11 |
| Igranje kitare | 20 | 15 |
| Ples | 8 | 15 |
| Mešanje koktajlov | 9 | 16 |
| Astronomija | 15 | 10 |
| Igranje košarke | 6 | 8 |
| Kuhanje hrenovk | 8 | 15 |
| Dokaz Einstein-Pitagorovega izreka | 12 | 15 |

Koliko točk lahko Miha doseže v tem času? Ker pa ima Miha rad izbiro, mu razloži, katere vse možne izbire ima za doseganje maksimalnega učinka. **Prepričaj ga, da so to res vse**!

Kaj pa, če ima na voljo **le 14 dni**? Koliko točk lahko doseže takrat? Katere »koktajle« spretnosti pa se lahko za največji učinek nauči takrat? Tudi tu **ga prepričaj, da so to res vse**!

Da bo lažje:

Z7 = {(8, 15), (14, 23), (16, 30), (23, 32) …}

S7 = {(0, 0), (6, 8), (8, 15), (9, 16), (14, 23), (15, 24), (16, 30), (17, 31), (23, 39)…}

Z8 = {(12, 15), (18, 23), (20, 30), (21, 31)…}

1. V podatkovni strukturi vrsta kot podatke lahko hranimo cela števila. Imamo vrsto, kjer so vsi podatki pozitivni. Sestavi algoritem prostorske zahtevnosti O(1), ki to vrsto spremeni tako, da bodo v njej samo še liha števila v istem relativnem vrstnem redu kot prej. Postopek vrsto 12, 7, 4, 5, 32, 1, 1, 7 torej spremeni v 7, 5, 1, 1.
2. Vas LeviDesni je prav posebna. Od pošte se da priti do vsake hiše na enoličen način tako, da se na nekaj križiščih odločiš, ali boš na njih zavil levo ali desno (prvo križišče je tik za pošto). Zato lahko vse poti do vseh hiš zakodiramo v dvojiško drevo, kjer velja:

* v korenu drevesa se nahaja pošta
* notranja vozlišča v drevesu predstavljajo križišča, na katerih lahko zavijemo levo oz. desno
* listi v drevesu predstavljajo hiše v vasi
* pot od korena do lista ustreza poti od pošte do hiše (če na poti proti hiši v križišču zavijemo levo, bo list, ki predstavlja to hišo v levem poddrevesu tega križišča, če pa desno, bo v desnem poddrevesu).

Novi poštar NePrevečLevi ima hude težave s privajanjem na leve zavoje, zato zagotovo ne zmore več kot dva zaporedna. Pomagaj mu in sestavi algoritem, ki kot vhod dobi zgoraj opisano opis vasi ter izpiši tiste hiše, do katerih se da od pošte priti tako, da nikoli ne gremo več kot dvakrat zaporedoma levo.



Primer: za drevo na zgornji sliki bi moral algoritem izpisati H2, H3, H7, H8, H6 ter H5. Vrstni red izpisanih hiš ni pomemben.