

A program célja:

A programban előre rögzített repülőgép járatokra lehet jegyet foglalni, foglalást törölni, menüt választani, valamint lehet járatokat keresni és meg tudja jeleníteni a foglaltsági térképet is. A járatokról pedig egy összesítést is meg tud jeleníteni.

A program felépítése:

Main függvény:

A program az indulást követően kiolvassa az eddigi foglalásokat (ha léteznek) a foglalások.txt fájlból. Később ugyanebben a fájlban fogja tárolni a program futása során bekövetkezett változásokat (új/törölt foglalás). A kiolvasott adatokat a program eltárolja.

A foglalható járatok a jaratok.txt -ben vannak rögzítve a következőképpen:

járatazonosító mikor(ÉÉÉÉ/HH/NN), honnan, hová, férőhely.

Ezeket is szintén beolvassa a program induláskor és ideiglenesen eltárolja az adatokat, majd a végén frissíti a fájlt. A járatok adatainak ilyen módon történő tárolására a férőhelyek változása miatt van szükség.

A program egy konzolos, menüvezérelt program.

A program a külső adatok beolvasása után kilistázza a menüpontokat, majd a felhasználótól bekéri az általa választott menüpont sorszámát. A sorszám egy egész szám, a program képes felismerni ha a felhasználó hibás adatot vagy nem létező sorszámot ad meg. A menü programrész egy ciklusban van, ami addig ismétlődik, ameddig a kilépő sorszámot nem adja meg a felhasználó. Amennyiben a felhasználó a kilépést választja, a ciklus nem fut le többször és a program folytatódik tovább.

Elérhető menüpontok:

1.: járat keresése: a menüpont választása után a program bekér két várost a felhasználótól(honnan-hová, a városok neveit ékezet nélkül kell megadni), majd egy időintervallumot(ÉÉÉÉ/HH/NN formában) és meghívja a jaratKeres() függvényt.

2.: repülőjegy foglalása: ebben a menüpontban a program egy járatszámot(egy adott járat számát a járat keresése funkciót használva lehet megtudni).

A program megpróbálja megkeresni a járatot a járatszám alapján. Ha a járat szerepel a rögzített adatok között és van férőhely, akkor bekéri a felhasználó nevét, majd megjeleníti a foglaltsági térképet. (Ellenkező esetben a program közli a felhasználóval a hibát: „nem létező járat”, illetve „nincs több férőhely”).

A foglaltsági térkép alapján a felhasználó el tudja dönteni, hogy melyik szabad helyet szeretné választani. Ezt a program szintén bekéri a felhasználótól. Ha olyan helyre szeretne foglalni, amelyre már létezik foglalás, akkor hibaüzenetet kap, ha a hely még szabad, akkor a foglalás tovább folytatódik.

A következő adat, amit a felhasználónak meg kell adnia az a járaton fogyasztani kívánt étel. Három menü közül lehet választani: normál, vega, laktózmentes. Ezeket a program a megszokott módon kilistázza és a bekért adat egy sorszám.

3.: foglalás törlése: ebben a menüpontban a program bekéri a felhasználó nevét, majd meghívja a jaratTorol() függvényt.

4.: összesítés: meghívja az Osszesit() függvényt.

5.: kilépés: ez a menüpont csak megszakítja a ciklust, a ciklusfeltétel miatt a menü már nem fut le többször.

A program a kilépés választása után még elvégzi a foglalások és a járatok adatainak mentését.

Ehhez megnyitja a már említett „foglalások.txt” és „jaratok.txt” fájlokat és beleírja az eltárolt adatokat, követve a formai követelményeket.

Formai követelmények:

foglalások.txt: minden foglalás külön sorban, minden adat szóközzel elválasztva ebben a sorrendben: járatszám, név, ülőhely, választott étel.

járatok.txt: minden járat külön sorban, minden adat szóközzel elválasztva ebben a sorrendben: járatszám, honnan, hová, indulás időpontja(ÉÉ/HH/NN formátumban), max férőhely, szabad férőhely.

Struktúrák:

A program háromféle struktúrát és egy enumot használ.

Datum: a megadott időpontok az egész számokra való feldarabolás után ilyen struktúrában tárolódnak el.

3 elem: év(int), hónap(int), nap(int)

Jarat: 6 elem: járatszám(string), Datum struktúra, induló város(string), megérkezés helye(string), férőhely(int), foglalt ülések(stringek tömbje)

Foglalas: 4 elem: járatszám(string), név(string), ülőhely(string), menü(int)

enum Menu: normal, vega, laktozmentes

Egyéb függvények:

jaratKeres(): 6 bemenet: járatok listája(pointer), járatlista mérete, indulási hely(string), megérkezés helye(string), kezdő időpont(Datum struktúra), végső időpont(Datum struktúra)

A függvény void típusú, nincs visszatérési értéke.

A függvény kilistázza azokat a járatokat, amelyek megfelelnek az induló városnak, az úticélnek és indulásuk a megadott időintervallumba esik.

jaratFoglal(): 5 bemenet: járatok listája(pointer), foglalások listája(pointer), a foglalás(Foglalas struktúra), járat indexe a járatlistában(int), foglalások listájának mérete(pointer)

A függvény a foglalások listájával tér vissza, az új foglalással kibővítve.

A függvény kibővíti a foglalások listáját egyel(azért kapja meg a méretet pointerként, hogy ezen egyúttal módosítani is tudjon), majd belehelyezi a foglalást.

jaratTorol(): 3 bemenet: a foglalások listája(pointer), a foglalások listájának mérete(pointer), a felhasználó neve(string)

A függvény a foglalások listájával tér vissza, a kitörölt foglalás nélkül.

A függvény a felhasználó neve alapján megkeresi a foglalását a foglalások[] tömbben és kitörli azt. Amennyiben nem létezik a megadott néven foglalás, akkor hibaüzenetet ír ki.

datumBeolvas(): nincs bemenet

A függvény egy Datum struktúrával tér vissza.

A függvény stringként beolvas egy dátumot a standard inputról, majd átalakítja az időpontot olyan formába, hogy az év, a hónap és a nap három külön egész számként legyen felhasználható, majd bele helyezi ezeket egy Datum struktúrába.

datumOsszehasonlit(): 2 bemenet: datum1(Datum struktúra), datum2(Datum struktúra)

A függvény bool típusú, igaz/hamis értékkel tér vissza.

Ez a függvény összehasonlítja a két megadott időpontot és igaz értékkel tér vissza, ha datum1 későbbi időpont, mint datum2. Ellenkező esetben, amikor datum1 korábbi időpont, mint datum2 akkor pedig hamis értékkel tér vissza.

hibaKeres(): 1 bemenet: datum(Datum struktúra)

A függvény bool típusú, igaz/hamis értékkel tér vissza.

Ez a függvény egy előre meghatározott időpont előtti időpontokat hibának érzékel, ezzel feltárhatóak azok az esetek, amikor a felhasználó a múltba szeretne jegyet foglalni. Igaz értékkel tér vissza ha hibát talált, hamissal ha nem.

foglaltsagiTerkep(): 1 bemenet: járatszám

A függvény void típusú, nincs visszatérési értéke.

Ez a függvény a járatszám alapján megjeleníti a foglaltsági térképet és más karakterrel jelöli a foglalt, illetve a szabad üléseket.

Osszesit(): 4 bemenet: járatlista(pointer), foglalások listája(pointer), járatlista mérete(int), foglalások listájának mérete(int)

A függvény void típusú, nincs visszatérési értéke.

Ez a függvény elvégzi az összesítést, kilistázza a járatokat járatszám alapján és megszámlolja, majd megjeleníti, hogy melyik menüből hányat kell felvinni a fedélzetre.