Ha egy adatszerkezetben az elemek számára lefoglalt tárrész konstans, és amelyben az elemek változtatják sorrendjüket azt egy félstatikus adatszerkezetnek nevezzük. A tárhely lefoglalása statikus a struktúra szintjén és dinamikus az elemek szintjén. Amikor egy elemre már nincs szükség az eltávolítjuk a memóriából, az elem által elfoglalt hely újra elérhetővé válik.

A félstatikus adatszerkezet a program folyamán bővül, új elemeket táról, azokat az elemeket amelyeket már nem használ kizár magából. Egy félstatikus adatszerkezet a verem is. A verem adatszerkezet egy LIFO lista, működésé hasonló egy földbe ásott vereméhez, melynek csak a tetején lehet terméseket be tenni illetve kivenni. A veremből fordított sorrendbe vesszük ki az elemeket.

A verem adatszerkezetben muszáj tárolni egy mutatót(stack pointer/verem mutató) mely mindig a legutoljára betett elemre mutat, ezen kívül tárolni kell magát „ a vermet” is,

A verem adatszerkezetnek különböző műveletei vannak, ahhoz, hogy új elemet tegyünk bele használjuk a beszúr(push) utasítást, ahhoz ,hogy a legfelső elemet eltávolítsuk használjuk a töröl(pop) utasítást. Ha csak megszeretnénk tekinteni a legfelső elemet és nem szeretnénk azt törölni, akkor használjuk a teteje(top) utasítást.

Mielőtt törölnénk a veremből a pop utasítással ellenőriznünk kell ,hogy nem üres-e, az Üres (isempty) utasítással, ha hozzá szeretnénk adni egy elemet akkor ellenőriznünk kell, hogy nincs-e tele a, Tele(isfull) utasítással.

Utasítások:

1. Létrehoz(): Létre hozza a vermet. Nincs előfeltétele, az utófeltétel: létre jön egy új verem, a verem mutató nem mutat sehova.
2. Ürít(): Törli a vermet, és a benne lévő elemeket. Nincs előfeltétel, utófeltétel a verem kiürítése, és a lefoglalt memória felszabadítása, ha nem statikus veremről van szó.
3. Beszúr() a verembe új elemet teszünk a segítségével, ez a verem tetejére kerül. Előfeltétel: a verem nincs tele, az utófeltétel: a verem tetején megtalálható az új elem.
4. Töröl(): a legfelső elemet vesszük ki a veremből ennek a műveletnek a segítségével. Előfeltétele: a verem nem üres, az utófeltétel: a verem tetején a tőrölt elem előtt álló elem lesz megtalálható, ha létezik.
5. Teteje(): ez a művelet vissza téríti a legutoljára betett elemet annak törlése nélkül. Előfeltétele: nem üres a verem, az utófeltétel: visszatéríti az utoljára betett elemet.
6. Tele(): eldönti a veremről, hogy megtelt-e? Előfeltétel: a verem létre kell legyen hozva. Utófeltétel: ha a verem tele van akkor igazat térítünk, ha nincs tele akkor hamist.
7. Üres(): eldönti az adott veremről, hogy üres-e. Előfeltétel a verem létre van hozva, utófeltétel igazat térit vissza ha a verem üres, hamist ha a verem nem üres.
8. Kiír(): kiírja az verem tartalmát. Előfeltétel a verem létre van hozva, utófeltétel kiírja a verem tartalmát.