

В дадения случай **Var** е променлива от тип *int.* Тя се намира на адрес 0х...7dd4. Това значи, че 4 последнователни байта след началото на този адрес са заети за тази променлива(нейната стойност е 10). Променливата **Ptr** е указател от тип *int\*.* Тя се намира на адрес 0х...99е8. Това значи, че 8 последнователни байта след началото на този адрес са заети за тази променлива(нейната стойност е 0х...7dd4). Един указател се държи като напълно нормална променлива с голямата разлика, че вместо да държи число, един указател държи в себе си, като данни, адрес от паметта.

Имаме две основни операции за работа със адреси и стойностите на тях:

- рефериране(&) - прилага се върху дадена **променлива**, за да разберем кой е първия от адресите, на които е записана тя(винаги върща адрес, нещо от сорта на 0x56A34DD)

- дерефериране(\*) - прилага се върху даден **адрес,** за да вземем стойността записана на него. Вече от това какъв е типа зависи колко последователни байта след този адрес ще се прочетат.(Например ако е \**int\** ще се прочетат 4, при \**char\** - 1, при \**float\** - sizeof(float) на брой байта и тн.)

Пояснения:

- Var = 10

- &Var = 0x7fffa0757dd4

- ptr = 0x7fffa0757dd4

- \*ptr = 10

- &ptr = 0x7fff98b499e8