

Osoitinmuuttuja

- Tietokoneen RAM-muisti koostuu peräkkäin numeroiduista muistipaikoista.
https://en.wikipedia.org/wiki/Random-access_memory
- Jokainen muuttuja sijaitsee omassa kohdassa muistia.
- Osoitinmuuttuja (*pointer*) on muuttuja, jonka arvo on **muistiosoite**.
- Sen tyyppi on sama kuin osoitetulla kohteella.
- Kun osoitinmuuttuja esitellään/määritellään, laitetaan muuttujan eteen merkki *

short *osoitinmuuttuja = nullptr; // asetetaan alkuarvoksi puuttuva arvo tai arvo 0.

- Kun haluamme päästä käsiksi muistiosoitteessa sijaitsevaan **tietoon**, tulee myös käyttää osoitus-operaattoria *
- Osoitinmuuttuja käsite tulee olio-ohjelmoinnissa vastaan, kun käsitellään dynaamista muistinhallintaa.

Null-osoitin

Osoittimelle voidaan antaa arvoksi null, tyhjä. Kyseistä osoitinta kutsutaan silloin null-osoittimeksi (toisinaan nollaosoitin). Se on erikoisarvo, joka tarkoittaa että osoitin ei osoita minnekään. Sisäisesti null-arvo on usein luku 0, sillä useissa käyttöjärjestelmissä ensimmäisen muistisivun käyttö on vikatilanne.

C++11-standardissa C++-kieleen lisättiin **nullptr**-osoiteliteraali, jonka voidaan antaa arvoksi osoitinmuuttujille, jotta osoittimet eivät jää alustamatta.

IN00BQ93 Laite- ja tuotesuunnittelun syventävät opinnot: ohjelmoinnin jatkokurssi

Osoitinmuuttuja

- Esimerkki: Muistiosoitteen sijoittaminen osoittimeen

```
short kokonaislukumuuttuja=10;  
short *osoitinmuuttuja = nullptr;           // nullptr arvo voidaan antaa arvoksi osoitinmuuttujille, jotta osoittimet eivät jää alustamatta.  
osoitinmuuttuja = &kokonaislukumuuttuja // kokonaislukumuuttujan muistiosoite osoitinmuuttujaan & operaattorin avulla
```

- Aluksi luodaan *kokonaislukumuuttuja* tyyppiä **short** ja annetaan sille alkuarvo 10.
- Sitten luodaan *osoitinmuuttuja* **short**-tyyppiseen tietoon ja alustetaan se siten, että osoitinmuuttuja ei siis osoita mihinkään muistissa.
- Osoite-operaattorilla & saadaan kokonaislukumuuttujan osoite esille ja tämä osoite tallennetaan sitten osoitinmuuttujaan
- Itse osoitinmuuttujan arvo on **muistiosoite** (esim. 0x229f00)
- Jos ohjelmassa kirjoitetaan alla oleva rivi, niin ohjelmaikkunaan tulostetaan arvo 10, koska * operaattorilla haetaan tieto siitä osoitteesta, johon osoitinmuuttuja osoittaa

```
cout << *osoitinmuuttuja << endl;
```

Muuttuja	Arvo	Muistiosoite
kokonaislukumuuttuja	10	0x229a00
...
osoitinmuuttuja	0x229a00	0x229f00

Osoitinmuuttuja, esimerkki 1

- Luo uusi projekti nimeltä **OsoitinJaViittaus (Non-Qt Project->Plain C++ Application)**. Poista main.cpp tiedostossa oleva koodi ja kirjoita alla oleva koodi **main.cpp** tiedostoon. Suorita ohjelmaa ja käy sitä läpi.

```
#include <iostream>
using namespace std;

int main()
{
    short kokonaislukuYksi;
    // Kun osoitinmuuttuja esitellään/määritellään, laitetaan muuttujan eteen merkki *
    short *osoitinmuuttuja = nullptr; // nullptr arvo voidaan antaa arvoksi osoitinmuuttujille, jotta osoittimet eivät jää alustamatta.

    kokonaislukuYksi = 10;
    osoitinmuuttuja = &kokonaislukuYksi; // osoitinmuuttuja saa muistiosoitteen arvokseen.

    cout << "kokonaislukuYksi muistiosoite: " << &kokonaislukuYksi << endl;
    cout << "kokonaislukuYksi arvo: " << kokonaislukuYksi << endl;
    cout << "osoitinmuuttuja muistiosoite: " << &osoitinmuuttuja << endl;
    cout << "osoitinmuuttuja arvo on aina muistiosoite: " << osoitinmuuttuja << endl;
    cout << "osoitinmuuttuja osoittaa arvoon: " << *osoitinmuuttuja << endl;
    return 0;
}
```

Osoitinmuuttuja, esimerkki 2

- Kommentoi **main.cpp** tiedostossa oleva koodi, ja kirjoita alla oleva ohjelma **main.cpp** tiedostoon. Tutustu ohjelman toimintaan.

```
#include <iostream>
using namespace std;

int main()
{
    short kokonaislukuYksi, kokonaislukuKaksi;
    short *osoitinmuuttuja = nullptr;

    kokonaislukuYksi =1;
    kokonaislukuKaksi = 2;

    osoitinmuuttuja = &kokonaislukuYksi;
    kokonaislukuKaksi = *osoitinmuuttuja; // Kun halutaan käsitellä muistiosoitteessa olevaa tietoa, käytetään osoitus-operaattoria *

    cout << "kokonaislukuYksi arvo: " << kokonaislukuYksi << endl;
    cout << "kokonaislukuKaksi arvo: " << kokonaislukuKaksi << endl;
    cout << "osoitinmuuttuja osoittaa arvoon: " << *osoitinmuuttuja << endl;

    return 0;
}
```

Tehtävä 1: Tee seuraava ohjelma käyttäen osoittimia

- Luo uusi projekti **OsoitinVaihda (Non-Qt Project->Plain C++ Application)**. Kommentoi **main.cpp** tiedossa oleva koodi.
- Määrittele **main()** funktioon kaksi **short** tyyppistä muuttujaa ja anna niille alkuarvot.
- Lisää ohjelmaan tyypitön (void) funktio nimeltä **vaihda**, jolla on parametrina kaksi **short** tyyppistä *osoitinta*.
- Kutsu **main()** funktiosta tyypitöntä (void) funktiota **vaihda**, anna parametreiksi aiemmin luodut **short** muuttujat.
- Koodaa **vaihda** –funktio siten, että se muuttaa **main()** funktiossa määriteltyjen **short** muuttujien arvot keskenään.
- Tulosta **main()** funktiossa muuttujien vaihdetut arvot.

Viittausmuuttuja

- Viittaus on kuin peitenimi
- Kun viittaus luodaan, se alustetaan toisen muuttujan tai olion nimellä
- Alustuksen jälkeen viittausta voidaan käyttää aivan kuin se olisi alustuksessa annettu muuttuja, mutta vain eri niminen
- Kaikki mitä viittaukselle tehdään tapahtuu varsinaisen viittauksen kohteelle
- Jos ohjelmassa on kokonaislukumuuttuja **int minunLuku**, siihen tehdään viittaus alla olevan mukaisesti

int &minunViittaus = minunLuku;

Viittausmuuttuja

- Kommentoi **main.cpp** tiedostossa oleva koodi, ja kirjoita alla oleva ohjelma **main.cpp** tiedostoon. Tutustu ohjelman toimintaan.

```
#include <iostream>
using namespace std;

int main()
{
    short kokonaislukuYksi;
    short &viittausMuuttuja = kokonaislukuYksi;

    kokonaislukuYksi = 10;

    cout << "kokonaislukuYksi: " << kokonaislukuYksi << endl;
    cout << "viittausMuuttuja : " << viittausMuuttuja << endl;

    viittausMuuttuja = 15;

    cout << "kokonaislukuYksi: " << kokonaislukuYksi << endl;
    cout << "viittausMuuttuja : " << viittausMuuttuja << endl;

    return 0;
}
```

Tehtävä 2: Tee seuraava ohjelma käyttäen viittauksia

- Luo uusi projekti **ViittausVaihda (Non-Qt Project->Plain C++ Application)**. Kommentoi **main.cpp** tiedossa oleva koodi.
- Määrittele **main()** funktioon kaksi **short** tyyppistä muuttujaa ja anna niille alkuarvot.
- Lisää ohjelmaan tyypitön (void) funktio nimeltä **vaihda**, jolla on parametrina kaksi **short** tyyppistä *viittausta*.
- Kutsu **main()** funktiosta tyypitöntä (void) funktiota **vaihda**, anna parametreiksi aiemmin luodut **short** muuttujat.
- Koodaa **vaihda** –funktio siten, että se muuttaa **main()** funktiossa määriteltyjen **short** muuttujien arvot keskenään.
- Tulosta **main()** funktiossa muuttujien vaihdetut arvot.