Lecture 2

Wednesday, February 12, 2020 11:27 PM

- passing argument
  - by value
    - create copies
  - by reference
    - int& refx = x;
    - refx = 20;
    - void DoubleValue(int& n){n = n * 2;} // this can actually change the value because the reference is passed
    - const int& refx = x;
    - void MatrixOperation(const BigMatrix& m1,
      const BigMatrix& m2) --> avoid using more memory but also does not allow the function to change the matrix
- pointer
  - Physical memory is arranged in blocks of bytes (8 bits)
  - Each memory location has an address
  - use * to declare a pointer
  - & : address of operator
  - * : dereference operator
  - used to access other variables/objects indirectly.
  - int* pnum = #
  - *pnum = 20;
  - int* pnum2 = nullptr; //good practice
  - int i = 10;
    - int* const ip = &i; //constant pointer
    - pointer ip can only point to i, but i_ can change
  - Lec2 p20-23 has an example
- arrays
  - int myarray[5];
    - myarray[0] = 3;
    - myarray[1] = 2;
  - int myarray[] {2, 3, 1, 14, 8}
  - address of the first element
    - &myarray[0]
    - myarray
    - int elem_1 = *myarray;
    - int elem_n = *(myarray + n - 1);
- strings
  - const char* greeting = "Hello";
  - '\0' is known as the string terminate character.
  - std::string
- control structure
  - if/else

    if (a==b)
    { … }
    else if (a==c)

```
{… }
else
{…}
```

- switch

```
cout << "Enter number: ";
int value = 0;
cin >> value;
switch(value)
{
    case 0: cout << "input: zero" << endl;
    break;
    case 1: cout << "input: one" << endl;
    break;
    case 2: cout << "input: two" << endl;
    break;
    default: cout << "input is not 0, 1 or 2";
}
```

- while

```
while (a==true)
{ … }
```

- do/while

```
do
{
    cout << "Enter number (0 to end): ";
    cin >> n;
    cout << "You entered: " << n << endl;
} while (n != 0);
```

- for

```
for (unsigned int i=0, j=0; i<10 && j<10; ++i, j+=2)
{
    cout << "i : " << i << ", j : " << j << endl;
}
```

- create projects for factorial, check if prime, Fibonacci, square root using Babylonian method, bubble sort
- enum:
  - enum CurrencyType { USD=0, EUR=1, GBP=2, CAD=3, AUD=4};
  - cout <<"Enter foreign currency (USD=0;EUR=1;GBP=2;CAD=3;AUD=4):
    int foreignCurrency;
    cin >> foreignCurrency;
    switch (foreignCurrency) {…}