PFI lecture other built-in type

- Similar to **int (4 bytes)**
  - **short int** or **short (2 bytes)**
  - **long int** or **long (4 bytes)**
  - **long long (8 bytes)**
  - **unsigned int (no negative value)**
- Similar to **double (8 bytes)**
  - **Float (4 bytes)**
  - **long double (8 bytes)**
- Similar to **char (1 byte)**
  - **wchar_t (2 or 4 bytes)**
  - **char16_t (2 bytes)**

PFI lecture other operators

- postfix ++, a unary operator
- postfix --
- prefix ++
- prefix –
- The operand must be a lvalue (anything we put on the left hand size of assignment) such as a variable.
- Example:
  - `int i = 2; cout << i++ << endl; cout << i<< endl;`
  - `int i = 2; cout << ++i << endl; cout << i<< endl;`

PFI lecture other operators

- Postfit ++,-- has higher precedence, left to right
- Prefixt ++,-- right to left
- The expression with postfix++ or -- is a rvalue, value not location
- The expression with prefix ++ or -- is a lvalue, a location.
- Example (complex expressions)
  - int i = 2; i++++; // same as (i++)++, compiler error
  - int i = 2; ++++i; // same as ++(++i), ok
  - int i = 2; ++i++; // same as ++(i++), compiler error
  - int i = 2; (++i)++; // ok

PFI lecture other operators

- compound assignments, same precedence as assignment (=), the same association rules as assignment, and the same semantics
  - +=, -=,*=,/=,%=,
- x += y; // the same as x = x + y;
- x -= y; // the same as x = x - y;
- x += y -= x; // what is this?

PFI lecture other statements

- **do while loop**, the body will be executed at least once.

```
char c;
do {
  cout << "please type y or n for yes or no\n";
  cin >> c;
} while (c != 'y' && c != 'n');
cout << "you typed " << c << endl;
```

PFI lecture other statements

- **break**, used inside loop statements or switch statements

```
char c;
while ( true ){
  cout << "Type q to quit and others continue:";
  cin >> c;
  if (c == 'q')
        break;
  cout << "Hello!" << endl;
}
```

PFI lecture other statements

- **continue**, used inside loop statements only. Stop the current iteration (skip all the statements until the end of the loop) and start the next iteration (loop beginning)

```
char c;
for ( ; ; ){ // infinite loop?
  cout << "Type q to quit and others to continue:";
  cin >> c;
  if (c == 'q')
        break;
  if ( ! (c >= '1' && c <= '9') )
        continue;
  for (int i = 0; i < c - '0'; i++)
        cout << "Hello!" << endl;
}
```

PFI lecture other statements

- **switch**, specialized if else if statement

```
char c;
cin >> c;
switch (c) { // expression in () must be integral
  case 'a':
  case 'A':
        cout << "compute gpa\n";
        break;
  case 'b':
        cout << "listing\n";
        break;
  default:
        cout << "for the future\n";
  }
```