

Lecture 3 : 자료구조 추상화(abstraction) : 배열(array)의 경우

추상화(abstraction)는 모든 과학의 보편적인 표현양식이다. 추상화의 가장 대표적인 학문은 수학(mathematics)이다. 추상화를 통하여 개념을 확장하고 복잡한 문제를 체계적으로 이해할 수 있도록 도와준다. 컴퓨터 과학에서도 마찬가지이다. 특히 현대 SW개발의 경우 여러 사람의 공동작업이 필수적인데 이런 추상화가 잘 정립되지 않으면 전체 작업에 큰 혼란이 발생하게 된다. 추상화를 정확하게 이해하기는 쉽지 않지만 일반적으로 큰 규모의 SW개발에 참여하여 실제 작업을 하게 되면 궁극에는 왜 추상화가 필요한지 이해하게 될 것이다.

Object Oriented Language에서 제공하는 Class는 이 데이터 추상화를 가장 성공적으로 지원해주는 매우 중요한 feature이다. 따라서 Class의 활용여부가 추상화 기반의 SW 설계의 가장 중요한 요소가 된다.

3.1 추상 데이터 (Abstract Data)

- 설계(design)와 구현 (implementation)의 차이를 이해해야 한다.
e.g) 건축가와 목공의 차이, 디즈니 만화영화의 비밀, Rendering Farm
- 자료구조= <Physical Data, 연산 Operation>
- 설계(design)와 구현(implementation)의 분리

Q) a1, a2, a3 동작이 가능한 자료구조를 만들려고 한다면 ?

Q) Data Structure Augmentation

- 조작적 정의 (operational definition)
핑을 잡으면 '매'다.
산소는 밀폐된 공간에서 발화가 증가시키는 기체이다.
마시고 취하면 술이다.

시계(watch)의 추상화	시계의 구현
시간을 알 수 있어야 한다. 분(min.)을 알 수 있어야 한다. 시계를 끌 수 있어야 한다.	해시계 물시계, 모래시계, 배꼽시계 기계식 시계 휴대폰 시계 수정자 시계
초 단위로 알 수 있어야 한다. 날짜를 알 수 있어야 한다. 시각을 수정할 수 있어야 한다.	

3.2 windows 환경에서 작성한 C++ 코드가 UNIX에서도 잘 수행되는 현상을 추상화를 이용해서 설명해보시오.

3.3 우리나라 사람의 주민등록 번호가 신규로 분당 30-40명씩 들어온다. 각 순간에 어떤 사람이 있는지 검사할 수 있어야 한다. IS_NUMBER(this), 어떤 특정번호 A부터 B번까지의 모든 번호를 출력할 수 있어야 한다. SCAN(from, this)... 가장 좋은 자료구조는 어떤 것이 있을까 ?

- 다시 보지 않을 물건이라면 정리할 필요는 없다. 사람도 마찬가지.

3.3 객체지향 프로그래밍(Object Oriented Programming)과 캡슐화(encapsulation)

- 왜 객체지향 프로그래밍인가 ?
- 공동 작업을 할 때 생기는 문제를 해결
- 왜 C에서는 int, float와 같이 서로 다른 type을 만들어 두었을까요?
e.g.) 초기 assembly language에는 이런 type이 없음.
- 극단적인 encapsulation INTEGER X{2-100} ;

- “오른손이 짤 프로그램을 왼손이 알지 못하게 하라.”

3.4 배열의 추상화 타입

- Generate_Array(), Retrieve(), Store()
- 추가 연산자 = { Copy(A, B),
Indexing(i)
Get_Size(),
Compact(),
Sample(X, k) }

3.5 일반적인 C, C++에서 마련된 배열자료구조에서 잘되는, 잘 안되는 연산

- Order가 분명한 자료, 앞뒤 찾아보기,
- Compaction, Sort, Insertion, Size Manipulate,

3.6 주어진 프로그래밍 언어에서 살펴보아야 할 내용

- 제공해주는 기본 자료구조는 어떤 것이 있는지 ?
- 각 자료구조에서 bound되는 범위는 ?
`int x[10000000000] ;` compile 가능한가 ? access 가능한가 ?
- Exceptional case handling 은 어떤 것이 되는지 ?

Q) $x[-3] = a$; 은 가능한가 ?

만일 그러하다면 왜 대부분의 언어에서는 이 기능을 불가능하게 설정했을까 ?

```
int a[100], b[100] ;
for( i= 1 ; i< 200 ; i++ )    a[i] = function(int i) ;
for( i= -99 ; i< 100 ; i++ )    b[i] = function(int i) ;
```

3.7 “ 단일-변수 다항식을 배열로 자료구조를 잡아라.” 라고 한다면 어떻게 해야 하나 ?

$$- f(x) = 4x^5 - 3x^3 - 2x + 10, \quad g(x) = x^6 - 4x^5 + x^2 + x$$

$$- f(x) + g(x), \quad f(x) - g(x) ,$$

$$\frac{d}{dx} f(x)$$

Evaluate $f(2.34)$

3.8 배열로 표현하면 좋을 작업과 그에 필요한 동작(Operations)을 함께 나열해 보시오.

자료형	동작 1	동작 2	동작 3	이유
58명 학생의 성적관리				
카톡 사용자 관리 내역				
페이스 북 친구 관계				
패션 코디 (입고 나갈 옷)				
회사 조직도				
오버워치 게임 시간				
여름 29도 이상 days				

- 돌아만 가면 되는 것이 아니라, 재빨리 돌아가야 한다.

3.9 저장된 “숫자(number)”에서 정렬했을 때 오름차순으로 k번째 원소 찾아내기

숫자 1 = { 45, 67, 89, 345, 45 }

숫자 2 = { 13.4675, 111.5667, 0.08989, 3.45, 138.8245 }

숫자 3 = { $84 + \frac{23423}{12414321}$, $34 + \frac{8933423}{414676}$, $\frac{802899345623}{7199246767}$ }

- floating point 숫자를 믿지 마라.

$$\frac{80289934895623}{7199241943216767} \neq \frac{80289934895622}{7199241943216766} \quad \text{그러나 현실은....}$$

3.10 floating point 숫자의 문제

$a = 3/10,000,000$ 의 값을 천만번 더하면 과연 무엇이 나올까 ?

$a = t / N$

$x = a + a + a + \dots \dots \dots a$

$y = (t/N) * N$

3.11 최대한 실수 연산(floating point computation)을 줄여야 한다.

```
if  $\sqrt{45^2 + 31^2 + 101^2} < 114$  then
    print "A"  $\implies 45^2 + 31^2 + 101^2 < 114^2$ 
else
    print "B"
```

3.12 SW 개발 과정에서 자료구조 추상화를 하는 단계

- 어떻게 C++를 사용하여 짤 것인가를 생각하지 말고
- 그 프로그램이 어떤 기능을 수행해야 할 것인가를 먼저 규정해야 한다.
(사용자 설명서를 먼저 만들어야 한다.)

3.13 사례 연구

- N명의 학생의 성적 정보가 있다.
- 특정 게임 사이트에 많은 사용자들이 login하고 logout을 한다.

3.14 Component Software를 사용해야 하는 이유를 설명하시오.

- 수제(Craft) 4 Giga 크기의 USB 만들기
- 가정용 원자력 발전기
- 휴대용 원자력 보온병
- 캠핑가서 먹은 돼지 불고기, 김치찌개를 위해서 준비해야 할 것은 무엇 ?