

시험공부

2022년 3월 10일 목요일 오후 6:27

헝가리안 표기법 (Hungarian notation) 또는 헝가리 표기법 은 컴퓨터 프로그래밍에서 변수나 함수의 이름에 그 종류, 곧 흔히 데이터 타입 따위를 명시하는 표기법으로, 명명규칙의 일종이다.

헝가리안 표기법 시험 출제 : 타입(자료형), 작성규칙, 변수명

정답 : 변수명 작성시 자료의 타입(자료형)을 쉽게 알 수 있도록 자료타입의 첫 글자를 혼용해서 변수명을 작성하는 규칙이다.

Name : 문자형(Character)

Age : 숫자형, 정수형(Integer)

성별 : 논리형(Boolean), 참(true,1) 거짓(false,0)

c_name : 문자형 타입의 이름 변수(필드)

i_age : 정수형 타입의 나이 변수(필드)

b_sex : 논리형 타입의 성별 변수(필드)

=====

↳ 프로그래밍 언어에서 변수 및 함수의 인자 이름 앞에 데이터 타입을 명시하는 코딩 규칙이다.

선형 구조 : 배열, 큐, 스택, 선형리스트(연속리스트, 연결리스트), 데크(큐+스택 장점)-양쪽에서 삽입 삭제 가능

비선형 구조 : 트리, 그래프

변수 작성 규칙

언더바_(밑줄) 사용가능, 하이픈- 사용불가, 자바에서는 \$로 시작가능

첫글자는 영문 대소문자, 특수문자_로 시작가능, 숫자로 시작 불가

예약어는 사용불가(if, else, for, while 등등)

Boolean Type(불린 타입) : 참(true), 거짓(false) 만 가진다.

정수 타입 구분 문제(자바 <>c언어)

★ 연산자 우선순위 > (괄호) > 단항 연산자(증감 연산자) > 산술 연산자 (*,/,+,-) > 시프트 연산자(>>, <<) >

관계 연산자(==,>,<= 등등) > 비트 연산자(&, |, ^) > 논리 연산자(&&, ||) > 조건 연산자(?, 삼항 연산자) >

대입 연산자(=, +=, -=)

++ > * > + > (<<) > == > & > && > +=

(괄호)

단항 연산자(증감 연산자)

++ -> 증가 연산자 = a=a+1
-- -> 감소 연산자 = a=a-1
! = 논리 NOT
~ = 비트 NOT

산술 연산자

+, -, *, / , % 순서대로 적용

시프트 연산자

a=10 b=a<<2 b= 40

a=10 b=a>>2 b=2

2비트 오른쪽으로 가라(>>) 예) 10/2의 2승=2.5 소숫점 버리고 2

2비트 왼쪽으로 가라(<<) 예) 10*2의2승 = 40

Ex)

a=16

b=a>>2

c=a<<1

->

b=4

c=32

관계 연산자(==, >=, <= 등등)

맞으면 1 아니면 0

비트 연산자

```
Or 2 | 4    0 0 1 0
           0 1 0 0
           =   0 1 1 0
And 2 & 4    0 0 1 0
           0 1 0 0
           =   0 0 0 0
```

xor 배타적 논리합 두 값이 같으면 0 다르면 1

```
2 ^ 10      0 0 1 0
           1 0 1 0
           =   1 0 0 0
```

논리 연산자

a=2, b=0

a&&b => 0

a||b =>1

4 &&0 = false

3||0= true

-1 &&6=true

4 & 10 0100

조건연산자(?, 삼항연산자)

예시

문제2) 정수형 변수 a=1, b=2 c=3 d=4 일 때 b*=a>b ? a : b; 답은 b*= 2 >> b=2*b=4

문제3) 정수형 변수 a=1, b=2 c=3 d=4 일 때 c-= a<b ? a-b : b-a

물음표 앞은 조건 / : 기준으로 참이면 앞 거짓이면 뒤로 간다

대입 연산자

a = a+1
a += 1
a++ -> 증가연산자
a = a-1
a -= 1
a-- -> 감소연산자

switch문 : 해당조건 하나만 처리한다.

다음 중 성격이 다른 하나는??

1. while 문
2. for문
3. do 문
4. switch 문

자료구조 연산 : $A + B$ / 연산자 : +, 피연산자 : A,B

수식의 표기법 :

- 전위 표기법(prefix) : 연산자를 연속된 피연산자 앞에 둠. +AB
- 중위 표기법(infix) : 연산자를 양쪽 피 연산자 중앙에 둠. A+B(인간)
- 후위 표기법(postfix) : 스택 사용. 연산자를 연속된 피연산자 뒤에 둠 AB+

문제) 다음 Postfix 연산식에 대한 연산 결과를 쓰시오

3 4 * 5 6 * + postfix => infix 변환해서 계산

(3 4 *) (5 6 *) +

{{(3 4 *) (5 6 *) +} 해당 괄호에서 연산자를 중앙으로 이동

{{(3 * 4)(5 * 6) +}

(3 * 4) + (5 * 6)

12 + 30 = 42

문제) $X = A + (B + C / D) * E - F$ 다음과 같은 중위식을 후위식으로 올바르게 표현한 것은?

$X = (A + (((B + (C / D)) * E) - F))$

$X = (A + (((B + C D /) * E) - F))$

$XABCD/+E*+F-=$

스택 사용 예)

1. 서브루틴 호출
2. 인터럽트
3. postfix(후위표기식)
4. 함수 복귀

객체 지향 프로그래밍 언어(O.O.P) - class를 만들 수 있다.

과거엔 구조적 프로그래밍 - C언어

현실 세계의 개체(Entity)를 기계의 부품처럼 하나의 객체로 만들어, 기계적인 부품들을 조립하여 제품 만들 듯이 소프트웨어를 개발 할 때도 객체들을 조립해서 프로그램을 작성할 수 있도록 한 프로그래밍 기법이다.

장점 : 1. 상속을 통한 재사용과 시스템의 확장이 용이하다.

2. 이해를 쉽게 해준다. 코드의 재활용성이 높다.

3. 개발 및 유지보수가 용이하다.

종류 : JAVA, C++, Smalltalk 등

*객체지향 프로그래밍 언어의 구성 요소

1. 객체(Object) : 하나의 클래스로부터 생성된 데이터(속성)와 이를 처리하기 위한 연산(메소드)을 결합시킨 실체이다.

* 속성 (Attribute) : 한 클래스 내에 속한 객체들이 가지고 있는 데이터 값들의 성질

* 메소드 (Method) : 함수, 연산, 오퍼레이션, 행위, 동작

객체가 메시지를 받아 실행해야 할 객체의 구체적인 연산을 정의한 것

2. 클래스(Class) : 공통된 특성(속성(데이터))과 행위(메소드,함수)를 갖는 객체의 집합이다. 두 개 이상의 유사한 객체들을 묶어서 하나의 공통된 특성을 표현하는 요소이다. 객체들을 생성해주는 하나의 설계도(틀)이다.

클래스에 대해 약술하라

정답 : 하나 이상의 유사한 객체들을 묶어서 하나의 공통된 특성을 표현한 것으로 데이터 추상화의 개념이다.

3. 메시지(Message) : 객체들 간에 상호작용을 하는데 사용되는 수단

4. 추상화 : 객체지향분석에서 불필요한 부분을 생략하고 객체의 속성 중 가장 중요한 것 에만 중점을 두어 개략 화 시킨 것
데이터의 공통된 성질을 추출하여 슈퍼 클래스를 선정하는 개념이다.

5. 상속(Inheritance) : 객체지향개념에서 이미 정의되어 있는 상위 클래스(슈퍼 클래스 혹은 부모 클래스)의 메소드를 비롯한 모든 속성을 하위 클래스가 물려 받는 것,
상속을 통해 하위 클래스는 상위 클래스의 모든 속성과 연산을 자신의 클래스 내에서 다시 정의하지 않고서도 즉시 자신이 속성으로 사용 할 수 있다.

-단일 상속 : 하나의 상위 클래스로부터 속성과 메소드 등을 물려 받는 것

-다중 상속 : 두 개 이상의 상위 클래스로부터 속성과 메소드 등을 물려 받는 것

6. 캡슐화 : 데이터(속성)와 데이터를 처리하는 함수(메소드) 를 하나로 묶는 것

캡슐화 된 객체의 세부 내용이 외부에 정보 은닉(은폐)되어, 변경이 발생할 때 오류의 파급효과가 적다. 캡슐화 된 객체들은 재사용이 용이하다.

7. 정보 은닉(information Hiding) : 캡슐화 에서 가장 중요한 개념으로, 다른 객체에서 자신의 정보를 숨기고 자신의 연산 만을 통하여 접근을 허용하는 것

8. 다형성(Polymorphism) 메시지에 의해 객체(클래스)가 연산을 수행하게 될 때 하나의 메시지에 대해 각 객체(클래스)가 가지고 있는 고유한 방법(특성)으로 응답 할 수 있는 능력 을 의미한다.

front-end : html(정적인 콘텐츠),css(스타일 시트), 자바스크립트(페이지에 동적인 효과) j-query(제이쿼리-데이터베이스 역할)
(폼, UI)

back-end : jsp, 스프링, 오라클(db), 서버 관련 기술=>연동

스크립트 언어: html내에 소스를 삽입해서 효과를 내어주는 기법

저급 언어 : 기계 중심의 언어

ex) 기계어 어셈블리어

(번역기=> 기계어인 어셈블리어로 프로그래밍 한 소스를 번역해주는 번역기 역할)

고급 언어: 인간중심의 언어, 쉽게 이해할 수 있다.

ex) C언어 , Python, java 등등