

1

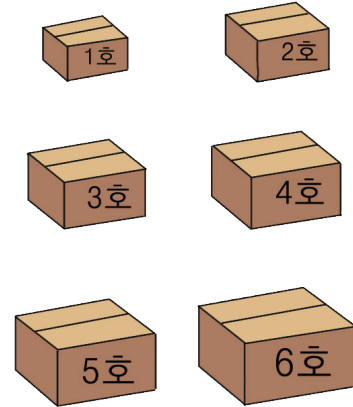
변수와 자료형

학습목표

- 변수의 개념을 이해하고 자료형에 따라 적절한 변수를 선언하여 사용할 수 있다.
- 자료형의 특성과 기억 장소의 크기를 이해하고 효율적인 기억 장소를 지정할 수 있다.

생각 펼치기

변수란 데이터를 저장할 수 있는 공간을 의미한다. 데이터를 저장하는 변수의 크기는 다양하다. 공간마다 자료를 저장할 수 있는 범위를 두어 적절한 데이터가 들어갈 수 있도록 지정하는 것을 자료형이라고 한다. 택배로 책 한 권을 보낼 때 작은 상자에 넣어 보낼 수 있지만, 큰 상자에도 넣을 수 있다. 또, 큰 곰인형을 택배로 보낼 때 큰 상자에는 들어가지만 작은 상자에 넣어 보낼 수 없다. 이때 상자를 변수라고 하며, 상자의 크기는 변수의 크기를 나타낸다.



핵심 질문

자료형에 맞게 변수를 선언하고 사용하는 방법은 무엇일까?

미션

이 단원을 학습하면서 해결해 보자.

2015학년도 대학수학능력시험 성적통지표

수험 번호	성명	생년월일	성별	출신 고교(반 또는 졸업 년도)		
10000001	홍길동	96.10.9.	남	평가고등학교(0001)		
구분	국어 영역	수학 영역	영어 영역	사회 탐구 영역		제2외국어 /한문 영역
	B형	A형		한국사	법과 정치	중국어1
표준 점수	102	133	127	61	56	59
백분위	89	89	90	83	62	85
등급	2	2	2	3	4	3

2014. 12. 3.
한국교육과정평가원장

대학수학능력시험의 성적을 저장하고 관리하는 프로그램을 작성하려고 한다. 필요한 자료가 아래 표와 같을 때, 프로그램에서 필요한 각 항목에 대한 자료형과 변수명을 지정해 보자.

국어 점수	수학 점수	영어 점수	한국사 점수	법과 정치 점수	중국어 점수
수험 번호	등급	백분위	표준 점수	전 과목 총점	전 과목 평균

1 C 프로그램의 구조를 알아보기

C 언어는 1970년대 초반 벨 연구소의 데니스 리치가 개발한 프로그래밍 언어이다. C 언어는 하드웨어를 제어할 수 있으며, 이식성이 뛰어나다. 또한 함수 단위의 모듈을 구현할 수 있으며, 구조적 프로그래밍이 가능하다.

1 C 프로그램의 기본 구조

일반적으로 C 프로그램의 기본적인 구조는 크게 헤더 파일을 포함(include)하는 부분, 프로그램의 시작(main 함수)을 나타내는 부분, 그리고 구체적인 프로그램 코드가 들어가는 몸체 부분(중괄호 안)으로 구성되어 있다.

프로그램

```
01 #include <stdio.h>
02 int main( )
03 {
04     printf("Hello, World\n");
05     return 0;
06 }
```

선언부로 프로그램 실행 전 필요한 준비 작업을 위해 사용

중괄호 { } 안에 처리할 명령문을 나열

프로그램의 몸체 부분

문장 끝에 세미콜론(;)을
쓰지 않으면 프로그램에서
오류가 나서 실행이 안 돼.



실행 결과

Hello, World

2 C 프로그램의 구성 요소

- ① **include 문과 헤더 파일**: include는 '포함하다'라는 의미를 나타내며 < > 안의 헤더 파일을 현재 프로그램에 포함시킨다. 헤더 파일은 확장자가 h인 파일로, 프로그램 작성에 사용되는 함수들을 미리 정해 놓은 파일이다.
- ② **main() 함수**: 가장 기본이 되는 main() 함수는 프로그램의 실행 시작과 마무리가 되는 곳으로 하나의 프로그램에는 반드시 하나의 main() 함수가 존재한다.
- ③ **시작과 종료**: C 언어에서 한 블록의 시작과 끝은 중괄호 { }로 표시한다. 한 문장의 끝은 세미콜론(;)을 반드시 붙여야 한다.

stdio.h

- stand Input/Ouput의 약자로 표준 입출력과 관련된 정의가 들어 있다.
- 프로그램에서 입출력 함수를 사용하기 위해서는 반드시 선언해야 한다.

3 C 프로그램을 작성할 때 주의 사항

- ① 하나의 프로그램에는 반드시 하나의 main() 함수가 존재해야 한다.
- ② C 언어 표준에 따라 main 함수의 리턴 타입은 int로 설정하고, 몸체의 가장 마지막 부분은 return 0;을 넣어준다.
- ③ 몸체 부분의 문장들의 가독성을 위해서 들여쓰기를 한다.

Q & A

Q return 0;을 넣어 주는 이유는 무엇일까?

A main 함수의 return문은 운영체제로 값을 반환한다. 0을 반환하면 에러가 없이 정상적으로 프로그램이 종료되었다는 뜻이다.

2 자료형과 변수에 대해 알아보자

컴퓨터를 통해 처리해야 할 데이터에는 숫자나 문자와 같이 다양한 자료형이 있다. 이러한 자료형은 어떻게 구분하며, C 언어에서 제공되는 기본 자료형에는 어떤 것이 있는지 알아보자.

1 자료형의 이해

컴퓨터가 처리하는 데이터는 크게 숫자와 문자가 있다. 숫자와 문자를 처리하기 위해서는 컴퓨터 내부의 기억 장치에 저장시켜야 하는데 효율적인 공간 활용을 위해서 C 언어에서는 기본 자료형을 제공한다.

기본 자료형은 크게 정수형과 실수형, 문자형으로 나뉘며, 자료형의 크기에 따라서 각각 갖는 값의 범위가 달라진다. 따라서 기억 장소를 효율적으로 사용하기 위해서는 처리할 값에 알맞은 범위를 갖고 있는 자료형을 선택하여 사용해야 한다.

[표 II - 1] C 언어의 기본 자료형

구분	자료형	크기	범위	비고
문자형	char	1Byte	$-2^7 \sim 2^7 - 1$	
정수형	short	2Byte	$-2^{15} \sim 2^{15} - 1$	
	int	4Byte	$-2^{31} \sim 2^{31} - 1$	정수형 기본 타입
	long	4Byte	$-2^{31} \sim 2^{31} - 1$	
실수형	float	4Byte	$-3.4 \times 10^{-38} \sim 3.4 \times 10^{38}$	
	double	8Byte	$-1.7 \times 10^{-308} \sim 1.7 \times 10^{308}$	실수형 기본 타입

ANSI 표준

- 실제로 C 언어의 표준을 정하는 ANSI에서는 추상적으로 자료형의 크기를 표준화하고 있다.
- short와 int는 최소 2Byte로 하며, int는 short보다 크기가 크거나 같아야 한다.



컴파일러마다 각 자료형의 크기는 고정되어 있지 않고, 자신의 컴퓨터 환경에서 sizeof 연산자를 통해 메모리 크기를 확인할 수 있다. sizeof 연산자는 단항 연산자로서 피연산자의 메모리의 크기를 반환한다.

sizeof 연산자

C 언어에서 자료형의 크기를 계산하는 데 사용하는 연산자이다. 프로그램을 실행하면 메모리 공간에서 차지하는 메모리의 크기를 바이트 단위로 계산한 값을 보여 준다.

예제 sizeof 연산을 이용하여 정수형 및 실수형 기본 자료형의 크기를 알아보는 프로그램을 작성해 보자.

프로그램

```
01 #include <stdio.h>
02 int main( )
03 {
04     printf("%d %d %d %d\n", sizeof(char), sizeof(short),
05                                     sizeof(int), sizeof(long));
06     printf("%d %d", sizeof(float), sizeof(double));
07     return 0;
08 }
```

실행 결과

```
1 2 4 4
4 8
```

더 알아보기 int가 아닌 기본 자료형의 연산

일반적으로 CPU는 처리하기에 가장 적합한 자료형을 int로 정의한다. 따라서 int형보다 작은 크기의 데이터형의 연산을 할 경우 결과가 int형으로 변환한다. 다음 프로그램을 통해 int가 아닌 다른 정수형 자료일 경우 어떤 결과가 나타나는지 확인해 보자.

프로그램

```
01 #include <stdio.h>
02 int main( )
03 {
04     char num1 = 1, num2 = 2;
05     printf("%d", sizeof(num1+num2));
06     return 0;
07 }
```

int가 아닌 다른 정수형 기본 자료

실행 결과

```
4
```

2 변수의 선언 및 할당

컴퓨터 프로그래밍에서 **변수**는 프로그램에 사용할 자료를 담아 두는 기억 장소의 이름을 의미한다. 변수란 '변할 수 있는 수'라는 의미로 프로그램이 실행되는 동안 변수의 값은 계속 바뀔 수 있다. 마치 음식을 담을 적당한 그릇을 선택하고 음식을 냉장고에 넣는 것과 유사하다.

① 변수의 선언

변수명은 변수명 규약에 맞게 만들 수 있다. 개인적으로 프로그램을 작성하거나 프로젝트를 진행할 때에는 일정한 규칙을 정하여, 변수의 목적에 맞는 이름을 선언하여 가독성 및 일관성을 높인다.

예약어

이미 정의된 뜻이 있어 사용자가 마음대로 사용할 수 없는 단어를 말하며, 주로 명령문에 해당한다.

- 반복문: for, while, do
- 조건문: if, else, switch, case, default
- 자료형: char, int, short, long, unsigned, float, double, struct, union, typedef
- 기타: sizeof 등

- 변수명에는 영문자, 숫자, _(밑줄)을 사용할 수 있다.
- 영문자의 대문자와 소문자는 다르게 구분한다.
- 예약어는 변수명으로 사용할 수 없다.
- 첫글자는 반드시 영문자 또는 _(밑줄)로 시작해야 한다.
- 변수명에 공백이나 한글을 사용할 수 없다.

C 언어의 변수는 자료형과 함께 선언한 후 사용한다. 변수에 넣을 자료의 형태에 따라 자료형을 저장한다. 같은 자료형으로 여러 개의 변수를 지정할 경우에는 변수와 변수를 콤마(,)로 구분하여 나열한다. 변수를 선언하는 형식은 다음과 같다.

변수의 선언 형식

자료형 변수명1, 변수명2, 변수명3, ... ;

다음은 여러 자료형에 대한 변수의 선언 예시 프로그램이다.

프로그램

```
01 #include <stdio.h>
02 int main( )
03 {
04     char myChar;
05     int count, resultNumber;
06     double weight;
07     return 0;
08 }
```

문자형 변수 myChar 선언

정수형 변수 count, resultNumber 선언

실수형 변수 weight 선언

실행 결과

출력없음

② 변수의 할당

변수의 할당이란 변수에 값을 넣는 것을 의미한다. C 언어에서는 '=' 기호를 이용하여 값을 할당하는데, 우리가 일반적인 실생활에서 쓰는 '같다' 라는 의미와는 다르다. 예를 들어, 'a=5'라고하면, '변수 a에 5를 저장한다.'라는 의미로, 변수의 할당 형식은 다음과 같다.

'==' 연산자

수학적 기호로 '='은 '같다'의 의미를 갖고 있다. C 언어에서는 '같다'는 '=='를 사용한다.

변수의 선언과 할당을 분리

```
int a;  
a = 5;
```

변수의 선언과 할당을 한번에

```
int a = 5;
```

변수의 값을 할당할 때에는 그 변수의 자료형에 맞는 값을 할당해야 한다.

예제 변수를 선언하고 자료형에 맞는 값을 할당하는 프로그램을 작성해 보자.

프로그램

```
01 #include <stdio.h>  
02 int main( )  
03 {  
04     char myBloodType = 'A';  
05     int age, schoolNumber;  
06     double weight;  
07     age = 18; schoolNumber = 12;  
08     weight = 173.4;  
09     return 0;  
10 }
```

char형 변수를 선언하고 문자 'A'를 할당

double형 변수 weight를 선언

weight에 실수값 173.4를 할당

실행 결과

출력없음



1. 성적 관리 프로그램에서 필요한 각 항목에 대한 자료형과 변수명을 지정해 보자.

구분	국어 점수	수학 점수	영어 점수	한국사 점수	법과 정치 점수	중국어 점수
자료형	int	int	int	int	int	int
변수명(예시)	korean	math	english	history	society	chinese
구분	수험 번호	등급	백분위	표준 점수	전 과목 총점	전 과목 평균
자료형	int	int	float or double	float or double	int	float or double
변수명(예시)	number	grade	percent	standard	sum	average

2. 선언한 자료형과 변수명을 가지고 C 언어로 선언하고 임의의 값을 초기화해 보자.

구분	변수의 선언 및 초기화
국어 점수	int korean = 89;
수학 점수	int math = 89;
영어 점수	int english = 90;
한국사 점수	int history = 83;
법과 정치 점수	int society = 62;
중국어 점수	int chinese = 85;
수험 번호	int number = 10010001;
등급	int grade = 2;
백분위	float percent = 8.75f;
표준 점수	float standard = 91.1f;
전 과목 총점	int sum = 498;
전 과목 평균	float average = 83f;

스스로
해결하기

다음과 같은 프로그램을 작성하였을 때 프로그램의 실행 결과는 어떻게 나타나는지 쓰시오.

프로그램

```

01  #include <stdio.h>
02  int main( )
03  {
04      int i = 2147483647;
05      printf("%d", i+1);
06      return 0;
07  }
```

실행 결과





오버플로우는 메모리가 표현 범위를 초과하는 수의 값을 저장하는 경우이며, 언더플로우는 메모리가 표현할 수 있는 수보다 작은 수를 저장하는 경우이다. 다음 프로그램을 통해 오버플로우와 언더플로우가 어떻게 쓰여지는지 확인해 보자.

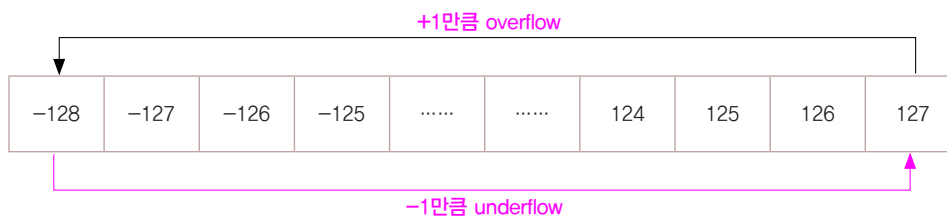
프로그램

```
01 #include <stdio.h>
02 #include <limits.h>
03 int main( )
04 {
05     printf("char의 최솟값: %d, char의 최댓값: %d\n", CHAR_MIN, CHAR_MAX);
06     printf("short의 최솟값: %d, short의 최댓값: %d\n", SHRT_MIN, SHRT_MAX);
07     char cA = 128, cB = -129;
08     short sA = 32768, sB = -32769;
09     printf("%d %d\n", cA, cB);
10     printf("%d %d\n", sA, sB);
11     return 0;
12 }
```

정수형 범위를 나타내는 상수를 정의함.

실행 결과

```
char의 최솟값:   -128, char의 최댓값:   127
short의 최솟값: -32768, short의 최댓값: 32767
-128 127
-32768 32767
```



char 변수 cA값 128은 char자료형의 최댓값(127)을 넘어 오버플로우된다.
char 변수 cB값 -129는 char자료형의 최솟값(-128)을 넘어 언더플로우된다.



수행 평가

변수의 할당을 이용한 값의 교환

활동 목표 변수의 할당을 통해 값을 교환할 수 있다.

두 개의 변수값을 교환하는 프로그램을 이용하여 다음 문제를 풀어 보자.

프로그램

```
01  #include <stdio.h>
02  int main( )
03  {
04      int a = 1, b = 2;
05      int temp;
06      temp = a;
07      a = b;
08      (가)
09      return 0;
10  }
```

초급 a와 b 변수값을 교환하기 위해 빈칸 (가)에 들어갈 코드를 작성해 보자.

중급 a 와 b 변수값을 교환하기 위해 다음과 같은 프로그램에서 빈칸 (나)에 들어갈 코드를 작성해 보자.

```
01  #include <stdio.h>
02  int main( )
03  {
04      int a = 1, b = 2;
05      a = a + b;
06      b = a - b;
07      (나)
08      return 0;
09  }
```

스스로 평가하기

평가 항목	구분		
	그렇다	보통이다	그렇지 않다
• 변수에 값을 할당할 수 있다.			
• 문제 해결을 위해 변수에 값을 할당하여 결과를 얻을 수 있다.			



내 실력 확인하기

내용을 이해했나요?

- **C 프로그래밍의 기본 구조:** 헤더 파일을 포함하는 부분, 프로그램의 시작을 나타내는 부분, 구체적인 프로그램 코드가 들어가는 몸체 부분으로 구성되어 있다.
- **자료형:** 숫자와 문자를 처리하기 위해서는 컴퓨터 내부의 기억 장치에 저장시켜야 하는데 효율적인 공간 활용을 위해서 C 언어에서는 기본 자료형을 제공한다. 기본 자료형은 크게 정수형과 실수형, 문자형으로 나뉘며, 자료형의 크기에 따라서 각각 갖는 값의 범위가 달라진다.
- **변수와 변수명:** 프로그래밍할 때 필요한 자료를 저장하는 기억 장소를 변수라 하고, 그 곳에 붙이는 이름을 변수명이라고 한다.
- **변수의 할당:** 변수에 값을 넣는 것을 의미하며, C 언어에서 '=' 기호를 이용하여 값을 할당한다.

문제로 확인할까요?

1. 다음과 같은 프로그램을 작성하였을 때, 프로그램의 실행 결과를 작성해 보자.

```

01  #include <stdio.h>
02  int main( )
03  {
04      int a = 1, b = 2, c = 3;
05      int temp;
06      temp = a;
07      a = b;
08      b = c;
09      c = temp;
10      printf("%d %d %d", a, b, c);
11      return 0;
12  }
```

2. 1번 문제에서 5번 라인부터 9번 라인까지의 코드를 수정하여, 실행 결과가 '3 1 2'가 되도록 프로그램을 수정해 보자.

평가해 볼까요?

★ 다음 평가 항목에 따라 자신의 성취 척도를 스스로 점검해 보자.

영역	평가 항목	척도				
		1	2	3	4	5
이해	자료형과 변수의 개념을 이해하고 있는가?					
적용	변수를 프로그램에 적용할 수 있는가?					