

프로그래밍을 통한 논리적사유연습





6 장. 표준 입출력

표준 입출력 함수의 종류

- 프로그래밍
 - 사용자가 어떤 값을 컴퓨터에 넣고(입력) 처리한 결과를 사용자에게 다시 제공하는 것(출력)
- 입출력 함수
 - 표준 입출력 함수: 키보드와 모니터를 통해 자료를 입출력
 - 파일 입출력 함수: 특정한 파일을 통해 자료를 입출력
 - 저급 입출력 함수: 운영체제 내, 시스템 호출을 통해 자료를 입출력

구분	종류
표준 입출력 함수	scanf(), printf(), getchar(), putchar(), gets(), puts()
파일 입출력 함수	fscanf(), fprintf()
저급 입출력 함수	read(), write()

표준 입출력 함수의 종류

- 표준 입출력 함수

구분	종류
형식화된 입출력 함수	scanf(), printf()
문자 입출력 함수	getchar(), putchar()
문자열 입출력 함수	gets(), puts()

형식화된 입출력

- **printf() 함수**
 - 표준 출력 장치인 모니터로 자료를 출력하는 함수
 - 도스창에 실행결과를 표시할 때 사용함
- 기본 형식

```
printf("문자열");  
printf("제어 문자열(Format-String)", 변수);  
printf("제어 문자열(Format-String)", 표현식);  
printf("제어 문자열(Format-String)", 변수나 표현식);
```

printf() 함수

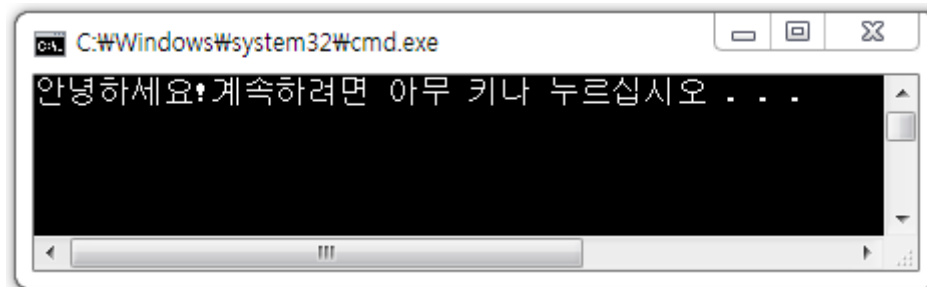
- 제어 문자열
 - 아래와 같은 변환기호(명세)를 추가하여, 이에 대응하는 값을 해당 형식에 맞게 출력 가능함

종류	의미
%d	인자의 내용을 부호 있는 10진 정수로 변환하여 출력한다.
%o	인자의 내용을 부호 없는 8진 정수로 변환하여 출력한다.
%x	인자의 내용을 부호 없는 16진 정수로 변환하여 출력한다.
%u	인자의 내용을 부호 없는 10진 정수로 변환하여 출력한다.
%e	인자의 내용을 지수 형태로 출력한다(부동소수점 형태).
%f	인자의 내용을 실수 형태로 출력한다(부동소수점 형태).
%g	인자의 내용을 %e 또는 %f의 변환 중 문자수가 적은 쪽으로 출력시키는데, 이때 유효하지 않는 0은 출력되지 않는다.
%c	인자의 내용을 문자 1개로 인식하여 출력한다.
%s	인자의 내용을 문자열로 인식하여 출력한다.
%n	%n이 나타날 때까지 출력한 문자수를 해당 인자에 저장한다.
%ld	인자를 long형 10진수로 출력한다.
%lu	인자를 long형 부호 없는 10진 정수로 변환하여 출력한다.

printf() 함수

- 예제) "안녕하세요!"를 출력

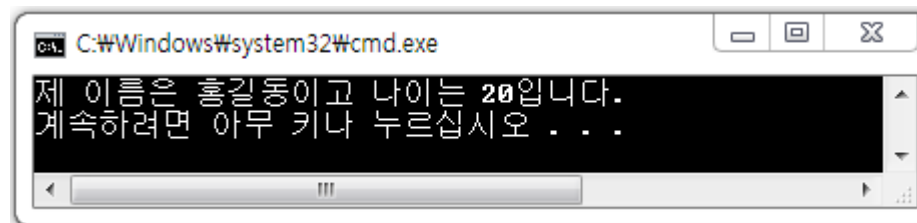
```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     printf("안녕하세요! ");
06
07     return 0;
08 }
```



printf() 함수

- 예제) 이름과 나이를 출력

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int age = 20;
06     printf("제 이름은 홍길동이고 나이는 %d입니다.\n", age);
07
08     return 0;
09 }
```

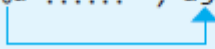


A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The command prompt displays the output of the program: '제 이름은 홍길동이고 나이는 20입니다.' followed by a new line and '계속하려면 아무 키나 누르십시오 . . .'. The text is in Korean and matches the example code provided.

printf() 함수

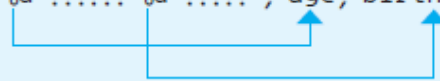
- printf(" ... %d ...", age);
 - 변환기호 %d의 출력형식에 맞춰 age의 값이 출력
 - 변환기호 %d와 변수 age는 서로 연결됨

```
int age = 20;  
printf(" ..... %d ..... ", age);
```



- 변환기호가 여러 개 있을 때
 - 순서대로 변환기호와 변수 또는 표현식 연결

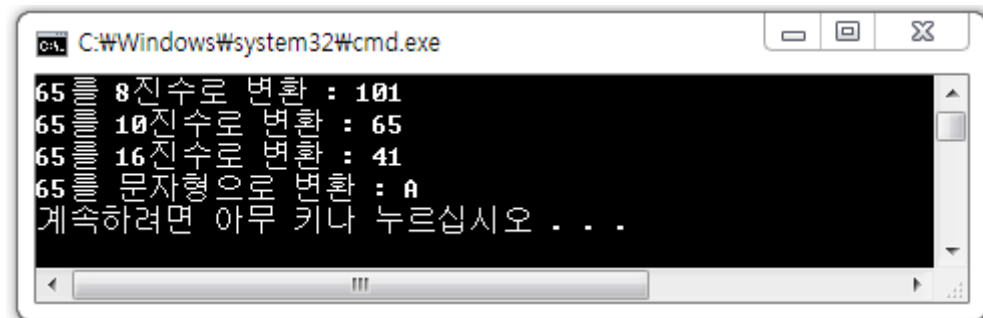
```
int age = 20;  
int birthyear = 1994;  
printf(" ..... %d ..... %d ..... ", age, birthyear);
```



printf() 함수

- 예제) 8진수, 10진수, 16진수, 문자 출력

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int Number = 65;
06
07     printf("65를 8진수로 변환 : %o\n", Number);
08     printf("65를 10진수로 변환 : %d\n", Number);
09     printf("65를 16진수로 변환 : %x\n", Number);
10     printf("65를 문자형으로 변환 : %c\n", Number);
11
12     return 0;
13 }
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe". The output of the program is displayed as follows:

```
65를 8진수로 변환 : 101
65를 10진수로 변환 : 65
65를 16진수로 변환 : 41
65를 문자형으로 변환 : A
계속하려면 아무 키나 누르십시오 . . .
```

scanf() 함수

- scanf() 함수
 - 표준 입력 장치인 키보드를 통해 자료를 입력 받을 때

- 기본 형태

```
scanf("변환기호(Format-String)...", &변수1, &변수2, ....);
```

- 변환기호: " " 안에 포함되는 내용이 어떤 자료 형식으로 변수에 입력되는지 결정

```
scanf("%d", &age);
```

- 변수 앞에 변수의 주소를 의미하는 기호인 &를 반드시 넣어야 함

scanf() 함수

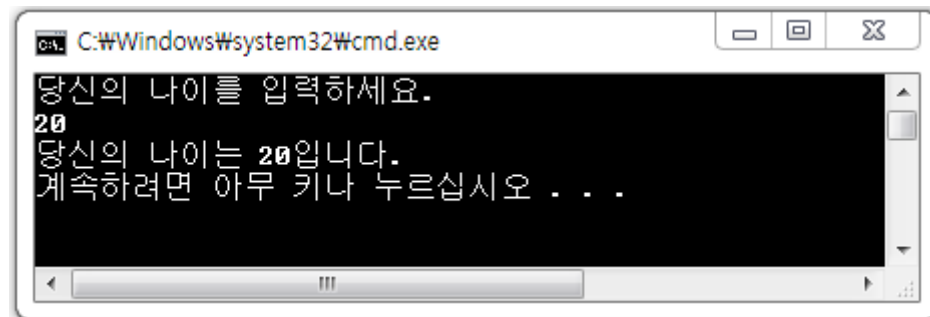
- 2개 이상 자료를 입력 받을 때
 - 쉼표를 넣어 변환기호와 변수를 순서대로 대응

```
Int age=0, year=0;  
scanf("%d %d", &age, &year);
```

scanf() 함수

- 예제) 나이를 입력받아 출력

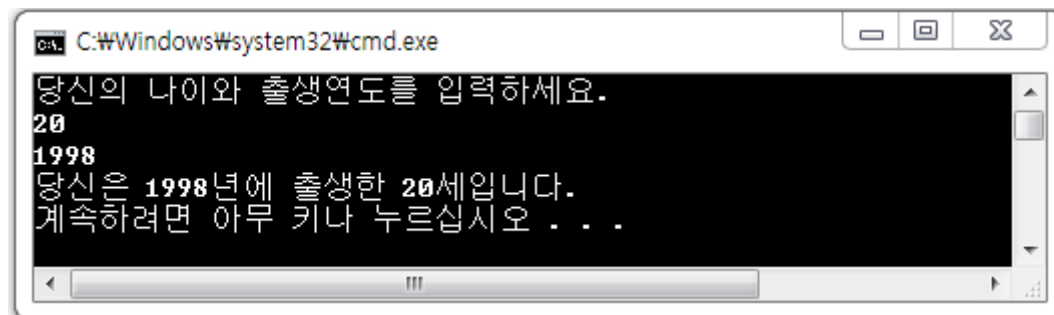
```
01  #include <stdio.h>
02
03  int main(void)
04  {
05      int age = 0;
06
07      printf("당신의 나이를 입력하세요. \n");
08      scanf("%d", &age);
09      printf("당신의 나이는 %d입니다. \n", age);
10
11      return 0;
12  }
```



scanf() 함수

- 예제) 나이와 출생연도를 입력받아 출력

```
01  #include <stdio.h>
02
03  int main(void)
04  {
05      int age = 0, year = 0;
06
07      printf("당신의 나이와 출생연도를 입력하세요. \n");
08      scanf("%d %d", &age, &year);
09      printf("당신은 %d년에 출생한 %d세입니다. \n", year, age);
10
11      return 0;
12  }
```



```
C:\Windows\system32\cmd.exe
당신의 나이와 출생연도를 입력하세요.
20
1998
당신은 1998년에 출생한 20세입니다.
계속하려면 아무 키나 누르십시오 . . .
```


scanf() 함수

```
C:\Windows\system32\cmd.exe
당신의 나이와 출생연도를 입력하세요.
20
1998
당신은 1998년에 출생한 20세입니다.
계속하려면 아무 키나 누르십시오 . . .
```

```
C:\Windows\system32\cmd.exe
당신의 나이와 출생연도를 입력하세요.
20 1998
당신은 1998년에 출생한 20세입니다.
계속하려면 아무 키나 누르십시오 . . .
```

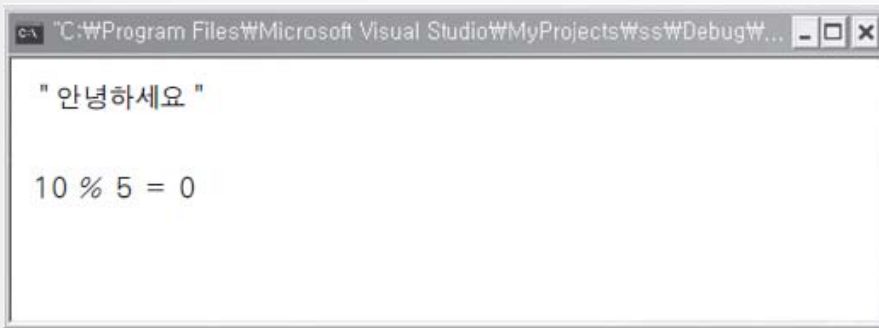
탈출 기법

- 탈출기법
 - 표현방법이 마땅히 없거나 특수문자 등을 입력하는 방법
 - 표의 프로그램 표현에 나와있는 내용을 printf() 함수와 같이 사용함

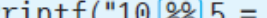
문자명	정수값(아스키 코드)	프로그램 표현	의미
경고	7	\a	경고음이 울린다.
수평탭	9	\t	수평으로 탭만큼 이동한다.
개행문자	10	\n	다음 줄로 이동한다.
캐리지 리턴	13	\r	현재 줄의 처음으로 이동한다.
큰따옴표	34	\"	"문자를 나타낸다.
작은따옴표	39	\'	'문자를 나타낸다.
역슬래시	92	\\	\ 문자를 나타낸다.
퍼센트	37	%%	% 문자를 나타낸다.

탈출 기법

- 탈출 기법을 이용한 출력 표현



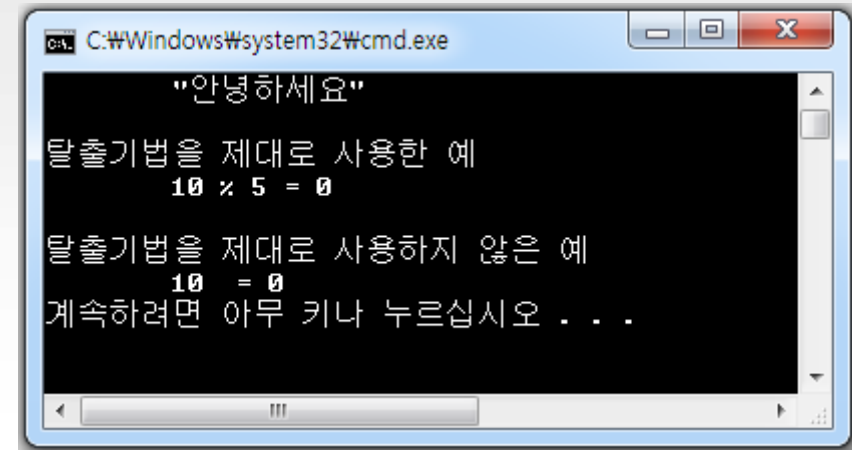
출력 : `printf("\안녕하세요\n");`
"안녕하세요"

– % 출력 : `Printf("10%% 5 = 0 \n");`


탈출 기법

- 예제) 탈출기법을 이용

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     printf("\t\" 안녕하세요\"\\n\\n");
06
07     printf("탈출기법을 제대로 사용한 예\\n");
08     printf("\t10 %% 5 = 0\\n\\n");
09
10     printf("탈출기법을 제대로 사용하지 않은 예\\n");
11     printf("\t10 % 5 = 0\\n");
12
13     return 0;
14 }
```



```
C:\Windows\system32\cmd.exe
"안녕하세요"
탈출기법을 제대로 사용한 예
10 % 5 = 0
탈출기법을 제대로 사용하지 않은 예
10 = 0
계속하려면 아무 키나 누르십시오 . . .
```

문자 입출력

- 문자 입출력 함수
 - 문자 1개를 키보드로 입력받아 출력하는 함수
 - getchar() 함수: 문자 입력 함수
 - putchar() 함수: 문자 출력 함수

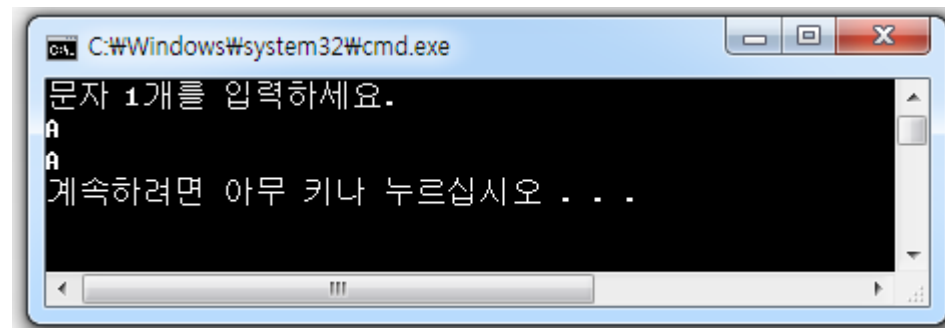
종류	의미
int getchar()	문자 1개를 키보드로 입력받으며 리턴값은 입력받은 문자에 해당하는 아스키 코드값이다.
int putchar(char 문자)	문자 1개를 모니터에 출력하며 리턴값은 출력하려는 문자에 해당하는 아스키 코드값이다.

```
char Input;  
Input = getchar();  
putchar(Input);
```

문자 입출력

- 예제) 문자 1개를 입력받아 출력하는 예제

```
01  #include <stdio.h>
02
03  int main(void)
04  {
05      char Input;
06      printf("문자 1개를 입력하세요. \n");
07
08      Input = getchar();
09      putchar(Input);
10
11      printf("\n");
12
13      return 0;
14  }
```



문자열 입출력

- 문자열 입출력 함수
 - 문자 여러 개를 한번에 입력받고 출력하는 함수
 - gets() 함수 : 공백 문자를 포함한 문자열을 입력할 때 사용
 - puts() 함수 : 표준출력 장치로 주어진 문자열을 출력할 때 사용

종류	의미
char *gets(char* str)	키보드에서 한 행을 입력받는다. 리턴값은 읽은 문자열이다.
int puts(char* str)	모니터에 한 행을 출력한다. 리턴값은 양수다.

문자열 입출력

- 예제) 이름을 문자열로 입력받아 출력

```
01  #include <stdio.h>
02
03  int main(void)
04  {
05      char Name[20];
06
07      printf("당신의 이름을 입력하세요 : ");
08
09      gets(Name);
10      puts(Name);
11
12      return 0;
13  }
```

