

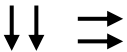
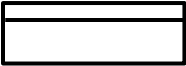

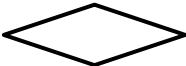


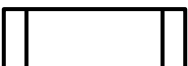
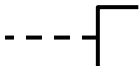








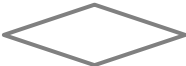



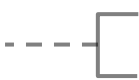

# 알고리즘 기초

2018-01학기  
홍명덕

# 순서도 기호 정리

기 호	이 름	의 미
	터미널	순서도의 시작과 끝을 표시
	준비	배열선언 및 초기 설정에 사용
	흐름선(Flow-line)	순서도 기호간 연결 및 흐름을 표시
	반복(Loop)	반복 수행
	입출력	데이터의 입출력시 사용
	비교, 판단	비교 및 판단에 의한 논리적 분기를 할 경우 사용
	결합	같은 페이지에서 순서도 흐름을 연결
	페이지 결합	순서도 흐름이 다른 페이지로 연결될 경우 사용
	서브루틴 (함수)	부 프로그램 처리
	주석	주석이나 설명을 표시
	처리	여러 가지 종류의 작업을 처리하는 것




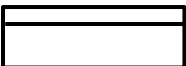

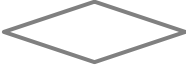





# 순서도 기호에 따른 C 코드

기 호	이 름	의 미
	터미널	<pre>#include &lt;stdio.h&gt;  void main() {     // to do. }  int sum(int a, int b) {     // to do. }</pre>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	






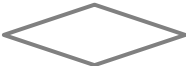





# 순서도 기호에 따른 C 코드

기 호	이 름	의 미
	터미널	<pre> int a;  int a = 0;  float b;  float b = 0.0;  int *p;  char *a;  int arr[10];  #define MAX 10 </pre>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	






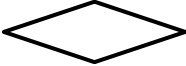





# 순서도 기호에 따른 C 코드

기 호	이 름	의 미
	터미널	<pre> for (int i=0; i&lt;10; i++) {     // to do. }  while (i &lt; 10) {     // to do.     i++; }  do {     // to do.     i++; } while(num &lt; 10 - 1) </pre>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	






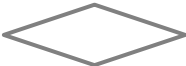


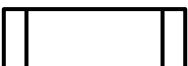


# 순서도 기호에 따른 C 코드

기 호	이 름	의 미
	터미널	<pre>printf("Hello world");</pre>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	

# 순서도 기호에 따른 C 코드



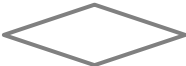


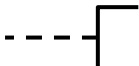

기 호	이 름	의 미
	터미널	<pre> if (a &lt; b) {     // true, to do. }  if (a &lt; b) {     // true, to do. } else {     // false, to do. }  if (a &lt; b) {     // first true, to do. } else if (a &gt; b) {     // second true, to do. } else {     // false, to do. }  switch (s) {     case 0:         // to do.         break;     case 1:         // to do.         break;     default:         // to do.         break; } </pre>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	

# 순서도 기호에 따른 C 코드






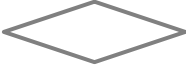





기 호	이 름	의 미
	터미널	<pre>func1();</pre> <pre>i = sum(10, 20);</pre>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	



# 순서도 기호에 따른 C 코드

기 호	이 름	의 미
	터미널	<div>// 한 줄 주석</div> <div>/*</div> <div>여러 줄 주석</div> <div>*/</div>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	

# 순서도 기호에 따른 C 코드

기 호	이 름	의 미
	터미널	<code>a = 1;</code>  <code>b = a + 1;</code>  <code>c = a &lt; b;</code>   <code>// 동적으로 메모리를 할당하는 함수</code> <code>int *ptr = (int *) malloc(sizeof(int));</code>  <code>// 힙 영역에 할당된 메모리를 해제하는 함수</code> <code>free(ptr);</code>  <code>// calloc함수는 malloc함수와 같은 기능</code> <code>int *ptr2 = (int*) calloc(5, sizeof(int));</code>  <code>// 이미 할당한 공간의 크기를 바꿀 때</code> <code>realloc(ptr2, sizeof(int) * 10);</code>
	준비	
	흐름선(Flow-line)	
	반복(Loop)	
	입출력	
	비교, 판단	
	결합	
	페이지 결합	
	서브루틴 (함수)	
	주석	
	처리	

# C Code

```
// 화면 지우기
```

```
#include <windows.h>
```

```
system("cls");
```

```
// 임의의 수 생성
```

```
#include<time.h>
```

```
#include<stdlib.h>
```

```
srand((unsigned)time(NULL)); // 시간을 이용한 seed값 사용
```

```
i = rand() % 10; // 0~9사이의 임의의 정수 생성
```