

# Control Flow Graph

<b>담당교수</b>	JUNBEOM YOO
<b>T11</b>	200412351 정백균
	200811468 강민석
	200913987 이승효

# Contents

- ◆ **Definition of CFG**
- ◆ **Algorithm : Create of CFG**
- ◆ **Statement of Purpose**

# ***DEFINITION OF CFG***

# Definition of CFG

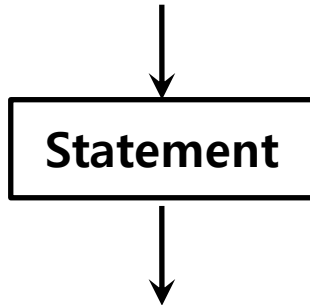
## ◆ CFG ?

- 프로그램이 실행되는 동안 프로그램이 traversal 할 수 있는 모든 경로를 표시한 그래프
- 단일 태스크에 의한 프로그램 순차흐름을 표현
- 함수 내의 소스코드 제어 흐름 구조 정보를 계층적 방향그래프 구조로 자동 생성
- 프로그램 최적화 정보를 얻기 위한 유용한 표현형태



# Definition of CFG

## ◆ Block



- ▶ events, activities, processes, functions, statement

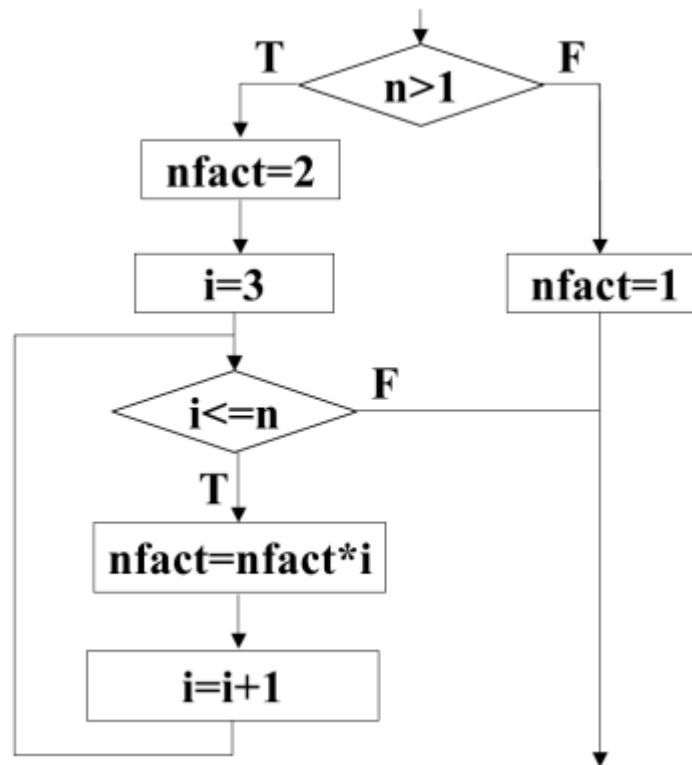


- ▶ branch conditions, outgoing edge

# Definition of CFG

## ◆ CFG of Factorial algorithm

```
int fact(n) { if (n>1)
  { nfact=2;
    int i=3;
    while(i<=n)
      { nfact=nfact*i;
        i=i+1;
      }
  }
  else
  { nfact=1;
  }
  return nfact;
}
```



```
.....
• 01  test n>1          •
• 02  if false goto 06  ---+
.....
•
.....
• 03  nfact=2          •
• 04  i=3              •
• 05  goto 07          -----+
.....
•
.....
• 06  nfact=1          -----+
.....
•
.....
+---07  test i<=n      -----+
| • 08  if false goto 12 ---+
| .....
| •
| .....
| • 09  nfact=nfact*i  •
| • 10  i=i+1          •
+---11  goto 07        •
| .....
| .....
| • 12  stop          -----+
| .....
| .....
```

# ***ALGORITHM : CREATE OF CFG***

- 1. GetBasicBlocks**
- 2. GetCFG**

# Algorithm : Create of CFG

## algorithm GetBasicBlocks

**Input.** A sequence of procedure statements.

**Output.** A list of basic blocks with each statement in exactly one basic block.

### **Method.**

1. Determine the set of *leaders*: the first statements of basic blocks.
  - (a) The first statement in the procedure is a leader.
  - (b) Any statement that is the target of an conditional or unconditional goto statement is a leader.
  - (c) Any statement that immediately follows a conditional or unconditional goto statement is a leader.
2. Construct the basic blocks using the leaders.

For each leader, its basic block consists of the leader and all statements.



# Algorithm : Create of CFG

## algorithm GetCFG

**Input.** A list of basic blocks for a procedure where the first block (B1) contains the first procedure statement.

**Output.** A list of CFG nodes and edges.

### **Method.**

1. Create entry and exit nodes
  - create edge (entry, B1); create edges (Bk, exit)
2. Traverse the list of basic blocks and add a CFG edge
  - (a) there is a conditional or unconditional goto statement from the last statement of  $B_i$  to the first statement of  $B_j$ , or
  - (b)  $B_j$  immediately follows  $B_i$  in the order of the procedure, and  $B_i$  does not end in an unconditional goto statement.
3. Label edges that represent conditional transfers of control as "T" or "F"; other edges are unlabeled.

# ***STATEMENT OF PURPOSE***

# Statement of Purpose

1. 입력은 C-language로 작성한 파일이다.
2. 출력은 콘솔 화면의 CFG이다.
3. 소스의 시작지점부터 종료까지 진행하는 모든 경우를 control flow로 나타낸다.
4. Control flow 로 출력되는 code는 block으로 나누어져 있다.
5. 모든 statement를 번호를 이용해 명시화하며 번호순서는 입력으로 받은 파일의 order와 같다.
6. block의 마지막 statement가 goto문일 경우 goto에 명시된 line과 연결된 모습을 출력한다.
7. block의 마지막 statement가 goto문이 아닐 경우 바로 아래 block과 순차적으로 연결된다.
8. 함수 호출, for문, while문, switch문에 대한 처리를 한다.

# Reference

## ◆ Definition of CFG

### - Wikidepia keyword : Control Flow Graph

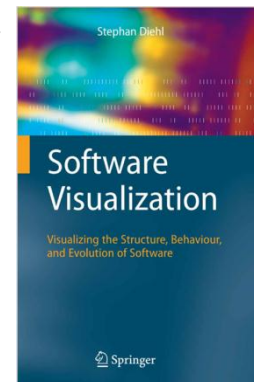
· [http://en.wikipedia.org/wiki/Basic\\_block](http://en.wikipedia.org/wiki/Basic_block)

### - [FREETECHEAMS.COM](http://www.freetecheams.com) : [Home](#) » [Computer Tips](#) » [Computer Tips](#) : CFG

· <http://www.freetecheams.com/computers-tips/computer-tips/control-flow-graph.html>

### - Software visualization - [Stephan Diehl](#), 2007

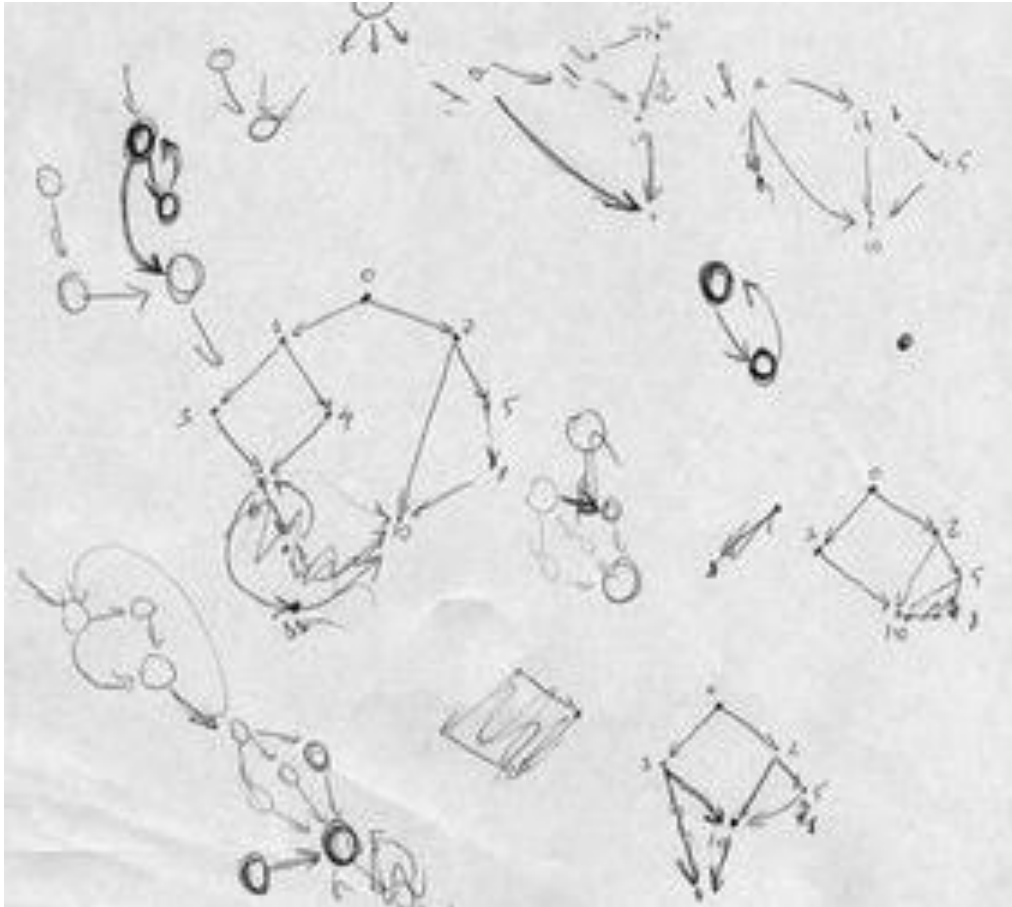
· pp.40 : 3.2.2 Control-Flow graph



## ◆ Algorithm : Create of CFG

### - Representation and Analysis of Software

· Gregg Rothermel, University of Nebraska – Lincoln, January 10, 2005



**Thank You!!**