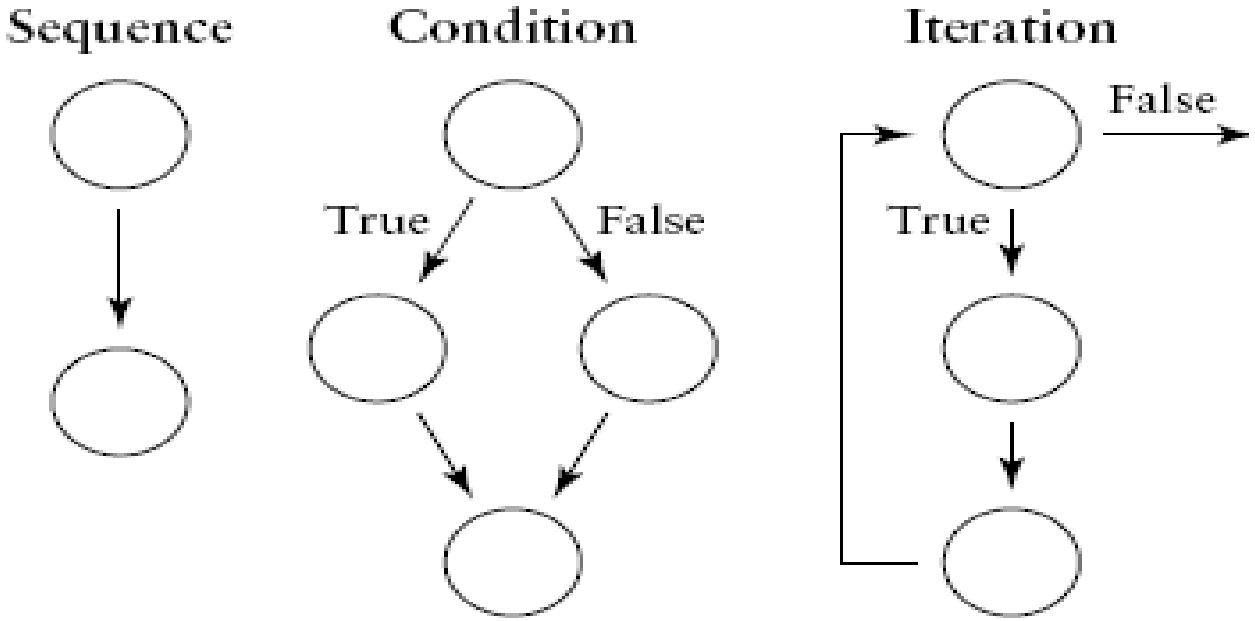
COVERAGE AND CONTROL FLOW GRAPHS

Introduction

- All structured programs can be built from three basic primes. They are
  - sequential (e.g., assignment statements),
  - decision (e.g., if/then/else statements),
  - iterative (e.g., while, for loops). Graphical representations for these three primes are shown in Figure.

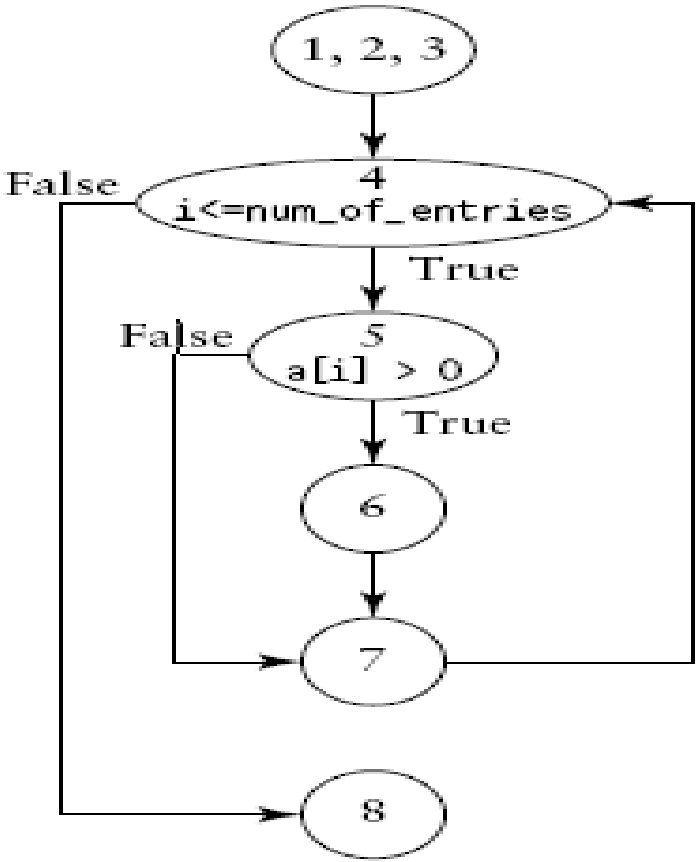
Representations



Example

- /* pos_sum finds the sum of all positive numbers (greater than zero) stored in an integer array a. Input parameters are num_of_entries, an integer, and a, an array of integers with num_of_entries elements. The output parameter is the integer sum*/
  - 1. pos_sum(a, num_of_entries, sum)
  - 2. sum_0
  - 3. inti_1
  - 4. while (i <_num_of_entries)
  - 5. if a[i] >0
  - 6. sum=sum+a[i]
  - endif
  - 7. i=i+1
  - end while
  - 8. end pos_sum

Flow graph for the example



COVERING CODE LOGIC

Decision or branch	Value of variable i	Value of predicate	Test case: Value of a, num_of_entries
			a = 1, −45,3 num_of_entries = 3
while	1	True	
	4	False	
if	1	True	
	2	False	

Example

- if(age <65 and married __ true)
- do X
- do Y ……..
- else
- do Z
- Condition 1: Age less than 65
- Condition 2: Married is true

Test cases for decision condition coverage

Value for age	Value for married	Condition 1 outcome	Condition 2 outcome	Decision outcome (compound predicate as a whole)	Test case ID
30	True	True	True	True	1
75	True	False	True	False	2
30	False	True	False	False	3

Thank You