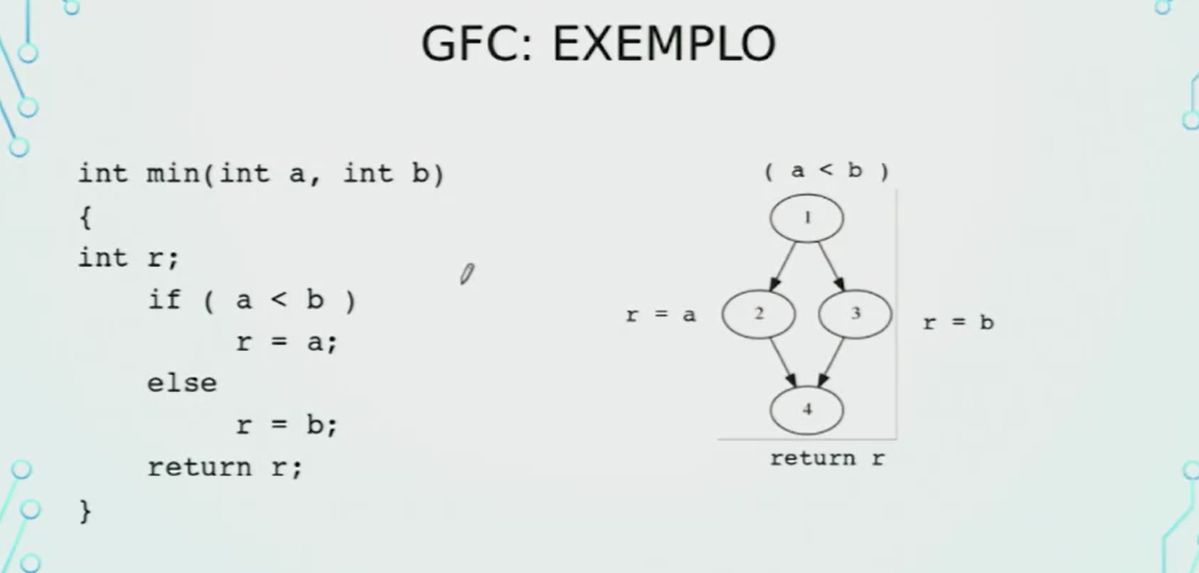
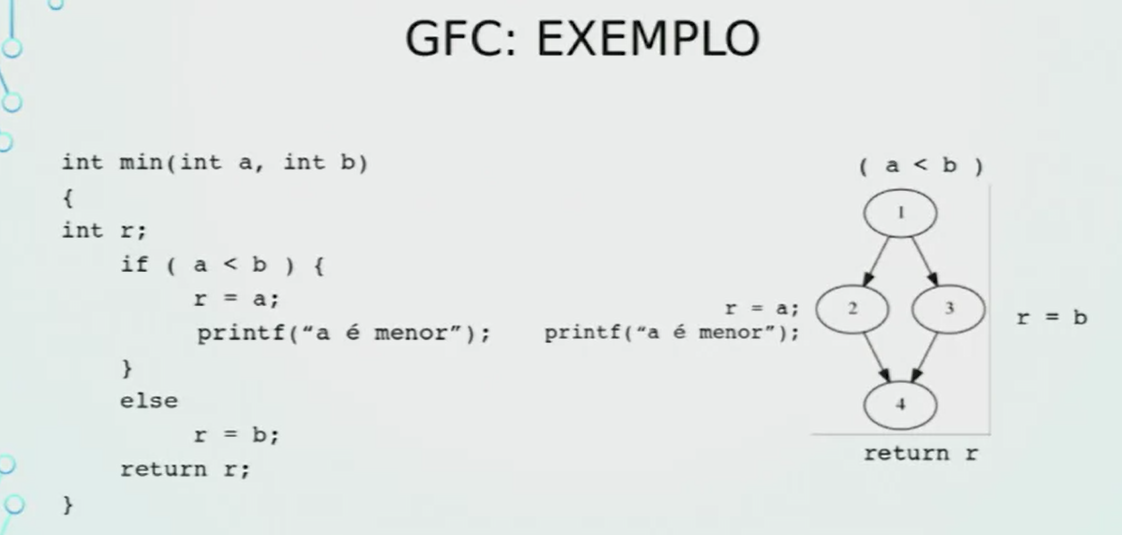
**QA – Teste Estrutural**

**Teste Estrutural** – Utilizado para **projetar testes baseando-se na estrutura interna (código) e na implementação de um programa. Tem por objetivo observar as estruturas que compõem o programa e garantir que todas essas estruturas sejam executadas, ao menos uma vez, durante os testes.**

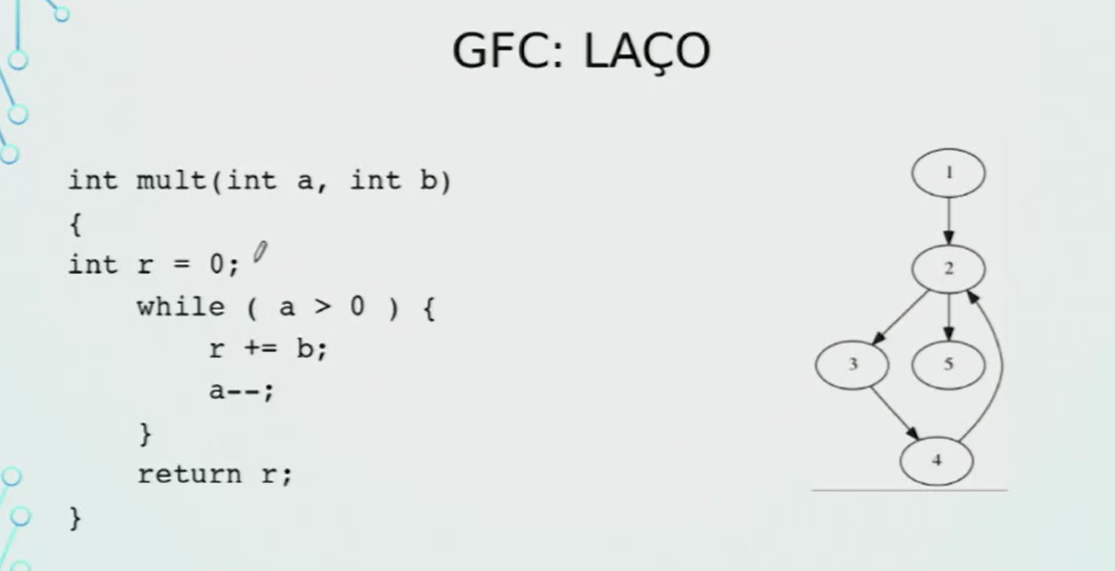
**Grafo de Fluxo de Controle (ou Grafo de Programa) – GFC**

**Exemplos:**

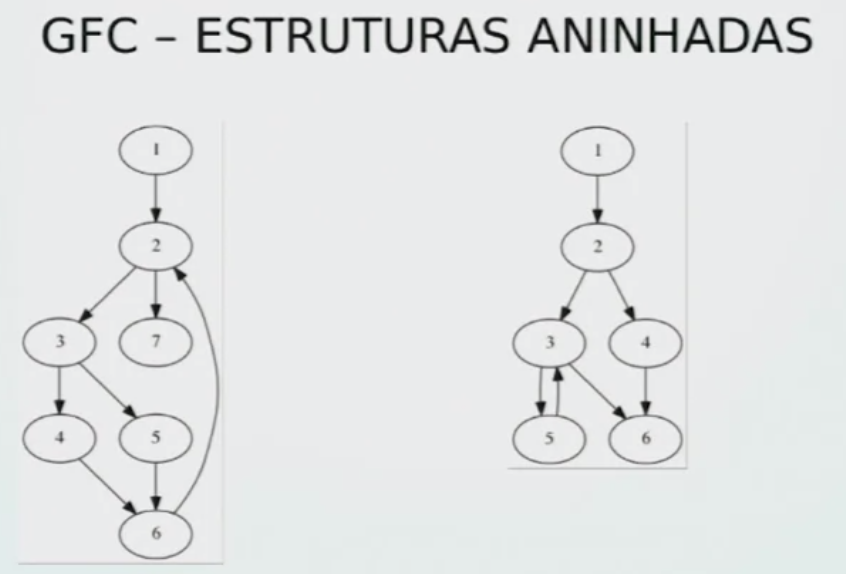
****

****

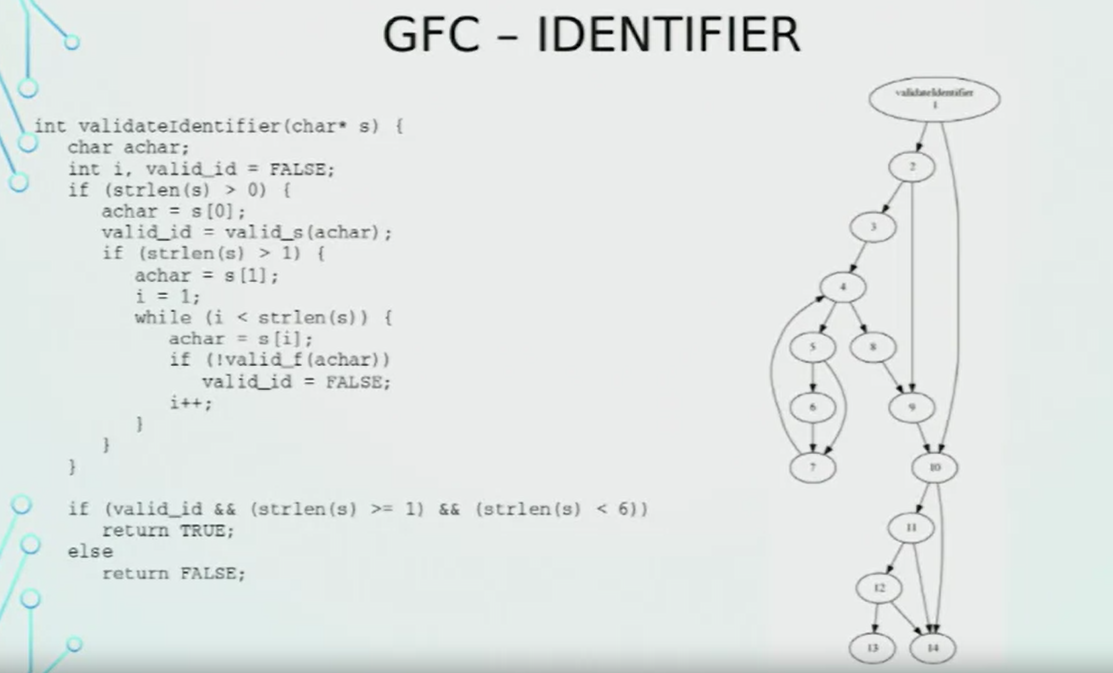
**GFC Laço:**

****

**GFC Estruturas Aninhadas:**

****

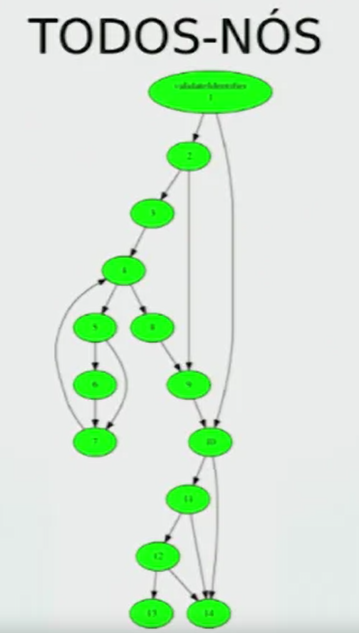
**GCF do Programa Identifier:**

****

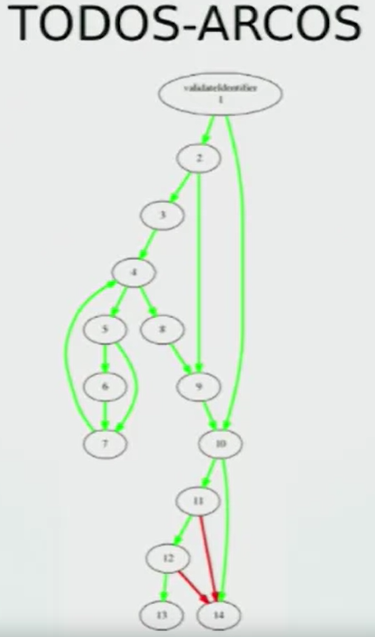
**Testes Baseados em Fluxo de Controles:** Encontrar casos de testes que exercitem os elementos do GFC.

**Critérios mais utilizados:**

**- Todos-Nós (ou Teste de Cobertura de Comandos)** – Casos de testes que executem todos os nós do fluxo de controle, pelo menos, uma vez cada:



**- Todos-Arcos (ou Teste de Cobertura de Ramos ou de Cobertura de Desvios)** – Executar todos os desvios que representam o fluxo de execução do programa:

****

**Código do identifier.c**

int validateIdentifier(char\* s) {

char achar;

int i, valid\_id = FALSE;

if (strlen(s) > 0) {

achar = s[0];

valid\_id = valid\_s(achar);

if (strlen(s) > 1) {

achar = s[1];

i = 1;

while (i < strlen(s)) {

achar = s[i];

if (!valid\_f(achar))

valid\_id = FALSE;

i++;

}

}

}

if (valid\_id && (strlen(s) >= 1) && (strlen(s) < 6))

return TRUE;

else

return FALSE;

}

**Ferramentas para Testes Estruturais** – Ferramentas utilizadas para identificar os testes executados, a partir dos casos de testes fornecidos. Essa ferramenta identifica os testes que já foram executados e os que ainda faltam ser efetuados.

* JaBUTi – <http://www2.ccsl.icmc.usp.br/pt-br/projects/jabuti>/ – LINK QUEBRADO;
* Eclemma – <http://www.eclemma.org>/;
* Cobertura – <http://cobertura.github.io/cobertura/>;
* GCC / GCOV / GCOVR – <http://gcovr.com/>.

**GCC – GNU Compiler Collection** –<http://gcovr.com/> - Permite a compilação e a criação de um código instrumentado, verificando quais trechos de testes foram executados e quais não foram executados, apresentando em uma visualização “amigável”.

É um conjunto de ferramentas, entre elas, um compilador de C / C++, que permite que o código seja compilado, afim de colher informações de cobertura. Permite a execução do código instrumentado, informando os trechos de código que foram executados.

**GCC – GNU COMPILER COLLECTION - Comandos para compilar:**

gcc -fprofile-arcs -ftest-coverage identifier.c -o identifier

identifier.c identifier.gcno

**GCOVR - Visualizar o resultado:**

gcovr -r . --branches --html --html-details -o identifier.html

**--> Lembrando que identifier é o nome do algoritmo testado**.

**Testes Baseados em Fluxo de Dados:**