Progetto di Tecniche di Compilazione Avanzata
A.A. 1999/2000
Predizione Statica degli Accessi al Cache

Si consideri il problema della *predizione statica* del comportamento di un programma, relativamente ai suoi riferimenti di memoria. Lo scopo principale di questo tipo di analisi è quello di classificare *staticamente* i riferimenti in memoria di un programma in modo da individuare:

- quelli che provocheranno un *cache miss,*
- quelli a cui corrisponderà sicuramente un *cache hit* ,
- quelli per cui non si può dire nulla a tempo di compilazione.

Questa analisi richiede di associare staticamente ad ogni punto del programma l'insieme dei contenuti possibili del cache in quel punto, considerando tutte le esecuzioni possibili del programma. Questa informazione puo' essere utilizzata, ad esempio, per individure un limite superiore al numero di cache misses del programma e quindi al suo tempo di esecuzione.

Una analisi di questo tipo e' specificata nel *rapporto* allegato al progetto. Le analisi definite nel rapporto sono:

a) *must analysis*: per ogni punto p del programma, individua l'insieme di blocchi di memoria che *sicuramente saranno presenti* nel cache, quando l'esecuzione raggiunge p;

b) *may analysis*: per ogni punto p del programma, individua l'insieme di blocchi di memoria che *possono essere memorizzati nel cache* quando l'esecuzione raggiunge p. Questa analisi può essere utilizzata per individuare i cache miss.

Entrambe le analisi vengono proposte nell'ipotesi che il cache utilizzi un metodo di indirizzamento *associativo su insiemi* ed il metodo di rimpiazzamento dei

blocchi sia di tipo LRU. Inoltre, per ogni analisi vengono definite le seguenti versioni:

a) l'indirizzo logico di ogni riferimento in memoria può essere determinato staticamente (paragrafo 5)

b) non tutti gli indirizzi logici possono essere determinati staticamente, ad esempio nel caso di riferimenti ad un vettore (paragrafo 8)

Si richiede:

1) L' implementazione delle analisi specificate in a), b) mediante PAG.

2) L'analisi definita in b) può essere associata ad una analisi degli *intervalli*, che definisca, quando possibile, un intervallo di valori interi che comprende i valori che una variabile può assumere in un punto del programma. Dopo aver implementato tale analisi mediante PAG, discutere gli eventuali *benefici*, in termini di significatività dell'informazione restituita, prodotti dalla applicazione contemporanea delle delle due analisi.

Per ogni analisi è richiesto:

a) la specifica del dominio ed una descrizione ad alto livello degli operatori astratti implementati mediante le funzioni di trasferimento: per le analisi specificate nel rapporto, tale descrizione dovrà essere ad un livello più basso di quella del rapporto, ma più alto della specifica degli operatori in FULA;

b) l'implementazione dell'analisi mediante il tool *PAG*;

c) la verifica di ogni analisi mediante almeno *3 esempi* .

Presentare, per ogni analisi, una breve relazione che descriva le *scelte effettuate*, i *tipi di dato* ed il *codice FULA* che implementa l'analisi, gli esempi scelti e l'output generato dall'analizzatore per ogni esempio.