

Docente: Marcelo Santos Linder.

Data: 14/05/2008

Disciplina: Programação para computação.

Durante nosso estudo sobre programação, vimos que os programas manipulam dados armazenados na memória do computador, representados através de bits. Estes dados podem ter diferentes significados, podendo, por exemplo, representar números inteiros ou reais, positivos ou negativos. No caso de considerarmos que estes dados representem números inteiros, a manipulação destes valores inclui operações como subtração, multiplicação e potenciação.

O objetivo deste trabalho é propiciar o estudo mais detalhado das formas de representação computacional de inteiros com sinal, bem como dos algoritmos utilizados para efetuar operações aritméticas sobre estes valores.

O trabalho em questão consiste em implementar um programa que manipule inteiros com sinal, representados com a notação em complemento de dois, através da efetuação sobre estes das operações de negação, soma, subtração, multiplicação e potenciação.

O programa deve receber na linha de comando na qual o mesmo é executado um inteiro maior que zero que corresponderá ao número de bytes utilizado para armazenar os valores em complemento de dois. Um menu deve ser apresentado ao usuário com as seguintes opções:

- Efetuar a negação de um número inteiro representado em complemento de dois;
- Efetuar a soma de dois números inteiros representados em complemento de dois;
- Efetuar a subtração de dois números inteiros representados em complemento de dois;
- Efetuar a multiplicação de dois números inteiros representados em complemento de dois;
- Efetuar a potenciação de um número elevado a outro número inteiro representado em complemento de dois;
- Finalizar programa.

Sempre após a efetuação de cada uma das operações, disponibilizadas no menu, além da impressão do resultado na saída padrão, deve ser perguntado ao usuário se o resultado da operação deve ser armazenado na “memória”, ou seja, se o resultado será guardado para uma utilização posterior. O usuário pode se utilizar da informação contida na “memória” como operando para qualquer uma das operações mencionadas, para tanto o mesmo digitará ‘m’ quando lhe for solicitado o valor do operando. Um detalhe relevante é que a informação armazenada na “memória” não é perdida quando o programa é finalizado, ou seja, em uma determinada execução do programa o usuário pode armazenar algo que será utilizado na próxima execução do mesmo. Como em cada execução do programa é informado quantos bytes serão utilizados para representar os valores manipuláveis, pode ocorrer uma incompatibilidade entre o tamanho do dado contido na memória, caso exista informação armazenada, e o tamanho do(s) operando(s) da atual instância do programa, sendo assim duas situações são possíveis:

1 – o número de bits necessários para o armazenamento do dado contido na memória é menor que o número de bits necessário para armazenar operandos. Neste caso, ocorrerá uma conversão implícita, ou seja, o dado da memória será convertido para a representação em complemento de dois com o número de bits utilizado para representar operandos, sem perda de informação;

2 – o número de bits necessários para o armazenamento do dado contido na memória é maior que o número de bits necessário para armazenar operandos. Neste caso, ocorrerá uma conversão explícita, ou seja, apenas parte do dado contido na memória será utilizada para constituir o operando em questão, a parte mais significativa da informação contida na memória será desprezada, há possibilidade de ocorrer perda de informação;

Após o usuário informar qual operação deve ser executada, o valor do(s) operando(s) deve(m) ser solicitado(s). As entradas devem ser validadas, ou seja, o valor dos operandos devem ser valores válidos. Uma informação relevante deve ser ratificada: como o tamanho de uma palavra manipulada por um computador tem tamanho limitado, os números que podem ser representados também têm tamanho limitado. Se o tamanho de uma palavra é igual a n bits, o maior número possível de ser representado é igual a 2^n . Se ao realizarmos operações sobre os números, elas gerarem resultados que não podem ser representados na quantidade de bits reservados ocorre o que denominados de *overflow* (números muito grandes). A ocorrência de *overflow* deve ser detectada.

O trabalho será efetuado em duplas, os componentes de cada dupla devem ser informados na aula do dia 16 de maio de 2008.

As equipes deverão apresentar um relatório sobre a forma como foi executado o trabalho e sobre o desempenho dos participantes na realização das tarefas. Esse relatório servirá como base para a avaliação dos trabalhos.

Alguns critérios de avaliação devem ser salientados:

1 - Cumprimento das tarefas;

2 - Elaboração do relatório de atividades;

3 - Se um determinado trabalho for plagiado por outra(s) equipe(s), ambas as equipes terão a nota final ZERADA.

OBS.: O programa deve ser escrito na linguagem C padrão ANSI. O programa não deve possuir variáveis globais.

O arquivo fonte do trabalho deve ser entregue em mídia e impresso, acompanhado do relatório do trabalho, no dia 11 de junho de 2008. Não serão tolerados atrasos na entrega.

A avaliação do trabalho se dará da seguinte forma:

1. Trabalho não compilou, **nota zero**;
2. Trabalho compilou e ao ser executado não apresentou saída alguma, **nota zero**;
3. Trabalho compilou, será submetido a testes e obterá nota igual a:

$$\sqrt[3]{\text{nota}_{\text{ testes}} \times \text{nota}_{\text{ arg üição}} \times \text{nota}_{\text{ qualidade}_{\text{ da}_{\text{ solução}}}}$$

A nota obtida nos testes varia de 0 (zero) a 20 (vinte).

A argüição será composta de questionamentos sobre a forma de implementação da solução, sua nota varia de 0 (zero) a 20 (vinte).

A qualidade da solução será avaliada levando-se em conta a quantidade de memória utilizada para solucionar o problema e ponderando-se a utilização, no desenvolvimento do programa, do potencial da linguagem. Esta nota variará de 0 (zero) a 20 (vinte).

Como pode ser observado a nota final do trabalho variará entre 0 (zero) e 20 (dez) pontos, esta nota será dividida entre os componentes da dupla, não necessariamente de forma igualitária, a divisão será arbitrada pelos componentes, em caso de não ocorrência de consenso, a nota de cada componente será igual a metade da nota obtida no trabalho.