

COMPILADORES

Prova 2
01/09/2022
Prof. Marcus Ramos

Observações:

- A prova será feita no horário da aula (entre 10:00h e 12:00h);
- A prova é individual;
- Qualquer suspeita de cópia ou ajuda indevida implicará a atribuição da nota zero à prova;
- Ao término da prova (12:00h) as respostas deverão ser enviadas por email para `marcus.ramos@univasf.edu.br` com o título "C-2022-1-P2-<nome>", onde <nome> deverá ser substituído pelo seu nome; não usar aspas no título da mensagem; exemplo: C-2022-1-P2-Marcus Ramos
- Em caso de mais de um envio, apenas o primeiro será considerado;
- Os emails deverão ser recebidos até 12:30h; será emitida uma confirmação de recebimento para cada email enviado;
- Emails não recebidos até 12:30h não serão considerados.

1. A montagem da AST pelo analisador sintático é importante pois:
 - a) Não é possível determinar se a cadeia pertence à linguagem sem ela;
 - b) Não é possível executar as demais fases sem ela;
 - c) Permite que as fases posteriores da compilação sejam executadas mais facilmente;
 - d) Facilita a análise léxica;
2. De que maneira o padrão de projeto Visitor facilita a construção do compilador?
 - a) Através da separação entre estrutura de dados e operação que é executada sobre a mesma;
 - b) O Visitor não facilita em nada a construção do compilador;

- c) Através da combinação entre estrutura de dados e da operação que é executada sobre a mesma;
 - d) Na medida em que facilita a construção do analisador de contexto, apesar das dificuldades introduzidas para a construção do gerador de código;
3. Qual motivo justifica a existência de uma fase de análise de contexto em um compilador?
- a) A necessidade de compilar em um único passo;
 - b) A necessidade de eliminar cadeias de entrada que não obedecem às regras de contexto;
 - c) A necessidade de verificar se a gramática livre de contexto está sendo observada;
 - d) A necessidade de minimizar erros de sintaxe livre de contexto na cadeia de entrada;
4. O que acontece durante a subfase "identificação" da análise de contexto?
- a) Apenas a verificação se cada nome usado foi previamente declarado;
 - b) Apenas a verificação se cada nome é declarado uma única vez em cada bloco;
 - c) A verificação se cada nome é declarado uma única vez em cada bloco e também se cada nome usado foi previamente declarado;
 - d) A verificação do tipo das variáveis;
5. Qual é uma mensagem de erro típica emitida durante a subfase de "identificação"?
- a) Variável de tipo incompatível;
 - b) Símbolo desconhecido;
 - c) Símbolo inserido em local incorreto;
 - d) Identificador não declarado;
6. Qual é o papel desempenhado pela Tabela de Símbolos durante a subfase de "identificação"?
- a) Auxiliar na identificação, garantindo que a mesma possa acontecer em apenas dois passos;
 - b) Facilitar a verificação de tipos;

- c) Viabilizar a análise de contexto;
 - d) Auxiliar na identificação, garantindo que a mesma possa acontecer em um único passo;
7. O que acontece durante a subfase "verificação de tipos" da análise de contexto?
- a) Os tipos dos operandos são modificados para satisfazer as operações em que são usados;
 - b) O valor de todas as expressões é determinado;
 - c) Os tipos dos operandos são confrontados com as assinaturas das operações;
 - d) As operações são avaliadas conforme sua precedência e associatividade, porém de forma independente dos operandos;
8. Qual é uma mensagem de erro típica emitida durante a subfase de "verificação de tipos"?
- a) Tipos incompatíveis;
 - b) Variável já declarada;
 - c) Variável não declarada;
 - d) Símbolo desconhecido;
9. Qual é o papel desempenhado pela precedência e pela associatividade dos operadores na avaliação de uma expressão?
- a) Estabelecer uma ordem para a avaliação da expressão;
 - b) Garantir que todos os parênteses sejam avaliados por último;
 - c) Garantir que toda expressão tenha um tipo;
 - d) Garantir que toda expressão produza um valor;
10. O que é uma AST decorada?
- a) É uma árvore de sintaxe depois da análise léxica;
 - b) É a árvore de sintaxe depois da identificação e da verificação de tipos;
 - c) É uma árvore que representa a cadeia de entrada;
 - d) É uma árvore que representa as regras de contexto da linguagem-fonte;

11. Se um tipo de dados possui 900 valores diferentes, em quantos bits (no mínimo) esse tipo deve ser representado?
- a) 9;
 - b) 8;
 - c) 10;
 - d) 16;
12. Se um tipo de dados é implementado em 20 bits, quantos valores distintos ele é capaz de representar?
- a) $1.024 * 1.024$;
 - b) Um milhão;
 - c) 65.536;
 - d) $256 * 256 * 256$;
13. O que justifica certos tipos serem implementados de forma direta e outros de forma indireta?
- a) A forma direta garante flexibilidade, a forma indireta garante eficiência;
 - b) Aspectos sintáticos da linguagem-fonte;
 - c) A necessidade de se compilar em um único passo;
 - d) A forma direta garante eficiência, a forma indireta garante flexibilidade;
14. Por que os valores das expressões indexadoras devem ser verificados antes de se fazer a indexação propriamente dita?
- a) Para garantir que nenhuma área de memória seja acessada ilegalmente;
 - b) Para garantir que as regras de contexto sejam observadas;
 - c) Para garantir que os tipos das expressões indexadoras sejam corretos;
 - d) Para garantir que os mesmos estejam fora da faixa prevista;
15. Qual a fórmula que representa o endereço do elemento $m[i][j]$, supondo a declaração em C `"int m [10][20]"` e também que cada "int" ocupa 2 bytes na memória do computador?
- a) $\text{address } [m[i][j]] = \text{address } [m] + j * 40 + i * 2$

- b) $\text{address}[m[i][j]] = \text{address}[m] + i * 2 + j * 40$
 - c) $\text{address}[m[i][j]] = \text{address}[m] + i * 40 + j * 2$
 - d) $\text{address}[m[i][j]] = \text{address}[m] + j * 2 + * 40$
16. Qual a fórmula que representa o tamanho da matriz m , supondo a declaração em C `"int m [10][20]"` e também que cada "int" ocupa 2 bytes na memória do computador?
- a) 200
 - b) 400
 - c) 40
 - d) 20
17. Qual é a origem virtual do vetor v , supondo a declaração em Pascal `"var v: array [-3 .. 3] of integer"`, supondo que cada "integer" ocupa 2 bytes da memória do computador e que o endereço base de v é 1000?
- a) 1006
 - b) 994
 - c) 1000
 - d) -6
18. De que maneira se relacionam as fórmulas para calcular o endereço de um elemento num agregado heterogêneo e num agregado homogêneo?
- a) O primeiro caso é um caso particular do segundo pois os elementos podem possuir tamanhos diferentes.
 - b) O primeiro caso é um caso particular do segundo pois todos os elementos possuem o mesmo tamanho.
 - c) O segundo caso é um caso particular do primeiro pois todos os elementos possuem o mesmo tamanho.
 - d) O segundo caso é um caso particular do primeiro pois os elementos podem possuir tamanhos diferentes.
19. De que maneira se relacionam as fórmulas para calcular o tamanho de um agregado heterogêneo e de um agregado homogêneo?
- a) O segundo caso é um caso particular do primeiro pois os elementos podem possuir tamanhos diferentes.

- b) O primeiro caso é um caso particular do segundo pois os elementos podem possuir tamanhos diferentes.
 - c) O primeiro caso é um caso particular do segundo pois todos os elementos possuem o mesmo tamanho.
 - d) O segundo caso é um caso particular do primeiro pois todos os elementos possuem o mesmo tamanho.
20. Por que a verificação de índice na indexação de um vetor não pode ser feita, no caso geral, em tempo de compilação?
- a) Porque isso onera o tamanho do compilador;
 - b) Porque isso onera o tempo de execução do compilador;
 - c) Porque o tipo da expressão indexadora pode não ser conhecido;
 - d) Porque o valor da expressão indexadora pode não ser conhecido;