



Centro Universitário

Centro de Educação Superior de Brasília
Centro Universitário Instituto de Educação Superior de Brasília

Curso: Engenharia de Computação

Campus: IESB SUL

Professor: CRISTIANO LEHRER

Data: 15/04/2024

Disciplina: Paradigmas de Linguagem de Programação

Turma: ENGCNM4A

Aluno: Mateus Bolzan Agnello

Matrícula: 2112082004

Assinatura: M.Bagnello

Nota: 4,0

Primeira Avaliação

INSTRUÇÕES

- Utilize a Folha de Respostas para responder todas as suas questões, tanto as objetivas de múltipla escolha quanto a(s) discursiva(s) (se disponível). As respostas às questões somente serão aceitas se registradas na Folha de Respostas.
- Utilize o espaço de resposta da(s) questão(ões) discursiva(s) dentro da prova (ou verso e/ou espaço ao final) para fazer seu rascunho. A(s) resposta(s) definitiva(s) deve(m) ser escrita(s) no(s) espaço(s) próprio(s) na Folha de Resposta(s) discursiva(s).
- Leia atentamente as questões antes de respondê-las.
- Releia atentamente as questões respondidas antes de entregar a sua avaliação.
- Use caneta esferográfica de tinta preta ou azul tanto para marcar as respostas das questões objetivas de múltipla escolha quanto para escrever a(s) resposta(s) da(s) questão(ões) discursiva(s).
- Observe com atenção a numeração das QUESTÕES OBJETIVAS DE MÚLTIPLA ESCOLHA da prova para marcar as respostas corretas das questões correspondentes localizada no gabarito.
- Para as QUESTÕES OBJETIVAS DE MÚLTIPLA ESCOLHA, preencha todo o campo conforme exemplo abaixo. Não serão consideradas e pontuadas questões rasuradas (respostas apagadas e repreenchidas e/ou respostas duplas).

| EXEMPLO | A | B | C | D | E |
|---------|-----------------------|----------------------------------|-----------------------|-----------------------|-----------------------|
| | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

- Para a(s) QUESTÃO(ÕES) DISCURSIVA(S), utilize apenas as linhas disponibilizadas. Aquilo que for escrito para além dessas linhas não será corrigido.
- Esta avaliação é composta de **5** QUESTÃO(ÕES) OBJETIVA(S) DE MÚLTIPLA ESCOLHA valendo **1,0** ponto(s) cada e **2** QUESTÃO(ÕES) DISCURSIVA(S) valendo **1,0** ponto(s) cada, totalizando **7,0** ponto(s).

INSTRUÇÕES ESPECÍFICAS:

A Questão Extra serve para complementar a nota dos trabalhos em até 1,0 ponto, respeitando o limite máximo estabelecido de 3,0 pontos. Exemplos:

- Caso a nota dos trabalhos seja 3,0 pontos, os pontos da questão extra serão descartados;
- Caso a nota dos trabalhos esteja entre 2,0 e 2,9 pontos, a questão extra poderá complementar a nota dos trabalhos até o limite máximo de 3,0 pontos, e os pontos sobressalentes serão descartados;
- Caso a nota dos trabalhos esteja abaixo de 2,0 pontos, a questão extra poderá aumentar a nota dos trabalhos em até 1,0 ponto.

Em hipótese alguma os pontos da Questão Extra serão somados com a nota da Primeira Avaliação.

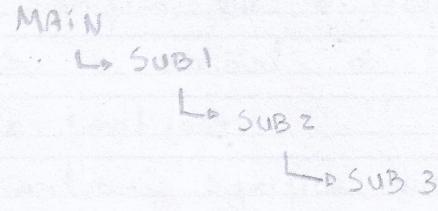
BOA PROVA!

| QUESTÃO | A | B | C | D | E |
|---------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|
| 1 | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> |
| 2 | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| 3 | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| 4 | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> |
| 5 | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

Questão 1 (vale 1,0 ponto)

Considere o seguinte programa escrito na sintaxe Pascal:

```
program MAIN;
var x: integer;
procedure SUB1;
var x: integer;
begin
  x := 4;
  SUB2(SUB3);
end;
procedure SUB2(procedure SUB)
var x: integer;
begin
  x := 6;
  SUB;
end;
procedure SUB3;
begin
  writeln('x = ', x);
end;
begin
  x := 2;
  SUB1;
end.
```



X := 6
X := 4

RASA = SUB 3
PROFUNDA = SUB 3, SUB 2, SUB 1, MAIN

Um aspecto interessante dos nomes de subprogramas passados como parâmetros é a questão referente ao ambiente de referenciamento correto para executar o subprograma passado. Com base no programa apresentado, avalie as asserções a seguir.

- I. Caso seja utilizado a vinculação rasa, o valor impresso será 6.
- II. Caso seja utilizado a vinculação profunda, o valor impresso será 4. ✓
- III. Caso seja utilizado a vinculação ad hoc, o valor impresso será 4.
- IV. Caso seja utilizado a vinculação rasa, o valor impresso será 2.

A respeito dessas asserções, assinale a alternativa correta.

- A) apenas as assertivas I e II.
- B) apenas as assertivas I e III. ✗
- C) apenas as assertivas II e III. ✗
- D) apenas as assertivas II e IV. ↙
- E) apenas as assertivas III e IV. ✗

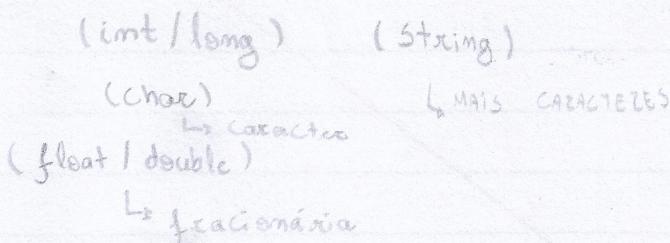
Questão 2 (vale 1,0 ponto)

(POSCOMP, 2022) Analise as seguintes assertivas sobre tipos de dados:

- I. Tipos reais (float/double) são utilizados para armazenar valores numéricos com parte fracionária. ✓
- II. Tipos caracteres (char) permitem armazenar um único caractere.
- III. Tipos inteiros (int/long) são utilizados para armazenar valores que pertencem ao conjunto dos números naturais (sem a parte fracionária).
- IV. Vetores, matrizes e ponteiros são exemplos de tipos de dados primitivos (básicos). ✗

Quais das assertivas apresentadas estão corretas?

- A. apenas as assertivas I e II. ↙
- B. apenas as assertivas I e III. ✗
- C. apenas as assertivas II e III. ✗
- D. apenas as assertivas II e IV. ✗
- E. apenas as assertivas III e IV. ✗



Questão 3 (vale 1,0 ponto)

(POSCOMP, 2022) Analise o código escrito na Linguagem C (Compilador Ansi C) abaixo e assinale a alternativa que corresponde a saída na tela.

```
int *p, **r, a = -1, c, b=10;  
p = &a; ~  
r = &p; ~  
c = **r + b--;  
printf("%d", c);
```

A. 7

B. 8

C. 9

D. 10

E. 11

Questão 4 (vale 1,0 ponto)

(POSCOMP, 2019) Considere as afirmações abaixo sobre comandos em linguagens de programação:

- I. Uma declaração de variável associa um nome a um valor que, geralmente, não pode ser alterado durante a execução do programa.
- II. Uma coerção é uma conversão de tipos generalizante, ou seja, aquela no qual um objeto é convertido para um tipo que pode incluir pelo menos aproximações para todos os valores do tipo original, como por exemplo, int para double.
- III. Expressões aritméticas são expressões cujos resultados são valores numéricos, inteiros ou fracionários.
- IV. Expressões lógicas são aquelas que têm como resultado um dos dois valores, verdadeiro ou falso.

Quais das assertivas apresentadas estão corretas?

- A. apenas as assertivas I e II.
B. apenas as assertivas I e III.
C. apenas as assertivas II e III.
D. apenas as assertivas II e IV.
E. apenas as assertivas III e IV.

Questão 5 (vale 1,0 ponto)

Considere o seguinte programa escrito na sintaxe C:

```
void sum(int i, int j, int t)  
{  
    for(; 0 < j; j--)  
    {  
        i = i + t;  
    }  
}  
  
void main()  
{  
    int k = 1;  
    sum(k, 5, k + k);  
    printf("%d", k);  
}
```

1 + 2 0<5
3 + 2 0<4
5 + 2 0<3
7 + 2 0<2
9 + 2 0<1
11

Com base no programa apresentado, avalie as asserções a seguir.

- I. Caso os parâmetros sejam passados por referência, o valor impresso será 11.
- II. Caso os parâmetros sejam passados por nome, o valor impresso será 243.
- III. Caso os parâmetros sejam passados por valor-resultado, o valor impresso será 13.
- IV. Caso os parâmetros sejam passados por valor, o valor impresso será 5.

A respeito dessas asserções, assinale a alternativa correta.

- A. apenas as assertivas I e II.
B. apenas as assertivas I e III.
C. apenas as assertivas II e III.
D. apenas as assertivas II e IV.
E. apenas as assertivas III e IV.

Questão 6 (vale 1,0 ponto)

A ideia existente por trás da semântica operacional é descrever o significado de um programa ao executar suas instruções em uma máquina, seja ela real ou simulada. As alterações que ocorrem no estado de uma máquina, quando ela executa determinada instrução, definem o significado desta. Usando as instruções da máquina virtual definida a seguir, apresente uma definição semântica operacional das instruções `break/continue` em C apresentada a seguir.

Pseudo

```
ident := var
ident := var op_bin var
ident := op_un var
goto label
if var relop var goto label
```

Onde,

ident é um identificador.
Var é um identificador ou uma constante.

op_bin pode ser um dos operadores aritméticos do conjunto { +, -, *, / }.

op_un pode ser um dos operadores unários do conjunto { +, - }.

relop pode ser um dos operadores relacionais do conjunto { =, <>, >, <, >=, <= }.

```
for (int a = 0; a < 10; a++)
{
    if (a == 4) continue;
    if (a == 6) break;
}
```

(Espaço para Rascunho – Transcreva para a Folha de Resposta)

- 01 A instrução "break" significa que o programa termina
- 02 naquele ponto, se a=6 o programa deve ser finalizado
- 03 Já a instrução "continue" significa que o programa deve
- 04 continuar rodando, se a=4 o programa simplesmente continua
- 05 o resto do programa
- 06
- 07
- 08
- 09
- 10

Questão 7 (vale 1,0 ponto)

Uma característica de projeto de expressões menos comumente discutida é a ordem de avaliação dos operandos. Por exemplo, no programa escrito na sintaxe C abaixo, o valor computado para a variável global a em fun2 dependerá da ordem de avaliação dos operandos na expressão a + fun1(). O valor da variável global a poderá ser 8 ou 20. Quais são as duas soluções para o problema da ordem de avaliação de operandos? Quais as desvantagens de cada solução?

```
int a = 5;
int fun1() {
    a = 17;
    return 3;
}
int fun2() {
    a = a + fun1();
}
int main() {
    fun2();
}
```

(Espaço para Rascunho – Transcreva para a Folha de Resposta)

01

02

03

04

05

06

07

08

09

10

Questão Extra (vale 1,0 ponto)

A vinculação do tipo subscrito a uma variável de array normalmente é estática, mas as faixas de valor de subscrito, às vezes, são vinculadas dinamicamente. Como acontece com as variáveis escalares, os arrays ocorrem em quatro categorias. Nesse caso, as definições de categoria baseiam-se na vinculação às faixas de valor de subscrito e nas vinculações ao armazenamento. Quais são essas quatro categorias e como elas relacionam as faixas de subscrito e o armazenamento dos elementos presentes no array;

(Espaço para Rascunho – Transcreva para a Folha de Resposta)

01

02

03

04

05

06

07

08

09

10