

SOLUÇÃO

MAC0113 - Introdução à Computação para Ciências Humanas - 2022S2

Avaliação Final

Utilize caneta azul ou preta e preencha completamente a quadrícula, como por exemplo: ■.

Não use símbolo X assim: ☒.

← Marque as quadrículas ao lado para formar o seu número USP e escreva seu nome completo em letra legível na linha pontilhada abaixo. **Se seu número possui menos que 8 dígitos complete com zeros à esquerda.**

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

Nome:

.....

O Departamento de Ciência da Computação considera qualquer forma de plágio e outros comportamentos antiéticos uma infração disciplinar inadmissível. Na ocorrência de tais casos, o Departamento recomenda que os estudantes envolvidos sejam reprovados na disciplina em questão, e que o ocorrido seja relatado à CG para as demais providências.

Estou ciente. Assinatura:

As provas são INDIVIDUALIZADAS.

Esta prova tem duração de 100 minutos. Não desmonte a prova.

SOLUÇÃO

QQuadrado Faça uma função `PertenceAoQuadrado` que receba como parâmetros dados dois números quaisquer, correspondentes às coordenadas de um ponto num plano cartesiano, verifique se esse ponto pertence ao quadrado de vértices localizados nas posições (-1,-1),(-1,1),(1,-1),(1,1). Caso o ponto pertença ao quadrado (inclusive nos limites), retorne verdadeiro (TRUE), ou retorne falso (FALSE), caso contrário.

```
PertenceAoQuadrado <- function(x,y) {
  L1
  L2
  L3
  L4
  L5
  L6
  L7
}
  return(res)
}
```

Rascunho

Para cada um dos itens a seguir, correspondendo às lacunas no código acima, assinale a única resposta que torna o programa acima correto. Cada linha do programa corresponde a uma questão abaixo. Não tente montar o programa testando as combinações possíveis pois não vai dar tempo. Escreva primeiro seu programa e procure analisar as opções abaixo depois. Isso é melhor do que tentar adivinhar o programa combinatoriamente olhando todas as alternativas. A cada opção errada que for selecionada, poderá ser descontada nota do exercício.

Questão 1:	L1: <input type="checkbox"/> <code>res <- FALSE</code> <input checked="" type="checkbox"/> <code>res <- FALSE</code> <input type="checkbox"/> <code>res <- TRUE</code> <input type="checkbox"/> <code>res <- 1</code>
Questão 2:	L2: <input checked="" type="checkbox"/> <code>if (</code> <input type="checkbox"/> <code>else if (</code> <input type="checkbox"/> <code>while (</code> <input type="checkbox"/> <code>for (</code>
Questão 3:	L3: <input type="checkbox"/> <code>(x >= -1) </code> <input checked="" type="checkbox"/> <code>(x >= -1) &&</code> <input type="checkbox"/> <code>(x > -1) &&</code> <input type="checkbox"/> <code>(x > -1) </code>
Questão 4:	L4: <input type="checkbox"/> <code>(x <= 1) </code> <input type="checkbox"/> <code>(x < 1) </code> <input type="checkbox"/> <code>(x < 1) &&</code> <input checked="" type="checkbox"/> <code>(x <= 1) &&</code>
Questão 5:	L5: <input type="checkbox"/> <code>(y > -1) </code> <input checked="" type="checkbox"/> <code>(y >= -1) &&</code> <input type="checkbox"/> <code>(y > -1) &&</code> <input type="checkbox"/> <code>(y >= -1) </code>
Questão 6:	L6: <input checked="" type="checkbox"/> <code>(y <= 1)) {</code> <input type="checkbox"/> <code>(y <= 1)) {</code> <input type="checkbox"/> <code>(y < 1)) {</code> <input type="checkbox"/> <code>(y == 1)) {</code>
Questão 7:	L7: <input type="checkbox"/> <code>res <- x</code> <input checked="" type="checkbox"/> <code>res <- TRUE</code> <input type="checkbox"/> <code>res <- FALSE</code> <input type="checkbox"/> <code>res <- FALSE</code>

SOLUÇÃO

QbubbleSort Um dos algoritmos redescobertos durante a aula de ordenação foi o algoritmo da bolha. Nele, valores vão sendo comparados dois a dois do início ao fim do vetor, várias vezes (o número de elementos do vetor, menos um). Escreva um programa em R que implementa o algoritmo da bolha. A função *Bolha* deve ser chamada sobre um vetor inicializado diretamente no código, não havendo função de leitura e o vetor impresso ao final da ordenação.

```
Bolha <- function(V) {
  L1: 
  L2: 
  L3: 
  L4: 
  L5: 
      tmp <- V[1]
      V[1] <- V[1+1]
      V[1+1] <- tmp
  }
  L10: 
}
L12: 
}
return(V)
}

W = c(4, 9, 1, 3, 7, -3, 4, 8)
W = Bolha(W)
for (valor in W) {
  print(valor)
}
```

Rascunho

Para cada um dos itens a seguir, correspondendo às lacunas no código acima, assinale a única resposta que torna o programa acima correto. Cada linha do programa corresponde a uma questão abaixo. Não tente montar o programa testando as combinações possíveis pois não vai dar tempo. Escreva primeiro seu programa e procure analisar as opções abaixo depois. Isso é melhor do que tentar adivinhar o programa combinatoriamente olhando todas as alternativas. A cada opção errada que for selecionada, poderá ser descontada nota do exercício.

Questão 8:	L1: <input type="checkbox"/> k <- dim(V)	<input type="checkbox"/> k <- length(V)-1 <input type="checkbox"/> k <- nchar(V) <input checked="" type="checkbox"/> k <- length(V)
Questão 9:	L2: <input type="checkbox"/> while (k > 0) {	<input type="checkbox"/> while (k > 1) { <input checked="" type="checkbox"/> while (k > 1) { <input type="checkbox"/> while (1 > k) {
Questão 10:	L3: <input checked="" type="checkbox"/> l <- 1	<input type="checkbox"/> l <- k <input type="checkbox"/> k <- 1 <input type="checkbox"/> l <- 2
Questão 11:	L4: <input type="checkbox"/> while (l <= k) {	<input checked="" type="checkbox"/> while (l < k) { <input type="checkbox"/> while (k < n) { <input type="checkbox"/> while (l < n) {
Questão 12:	L5: <input checked="" type="checkbox"/> if (V[1] > V[1+1]) {	<input type="checkbox"/> if (V[k] > V[k+1]) { <input type="checkbox"/> if (V[k] > V[1]) { <input type="checkbox"/> if (V[1] >= V[k]) {
Questão 13:	L10: <input checked="" type="checkbox"/> l <- l+1	<input type="checkbox"/> k <- k-1 <input type="checkbox"/> k <- k+1 <input type="checkbox"/> l <- l-1
Questão 14:	L12: <input type="checkbox"/> l <- l-1	<input type="checkbox"/> k <- k+1 <input type="checkbox"/> k <- 1 <input checked="" type="checkbox"/> k <- k-1